
oci Documentation

Release 1.3.7

Oracle

Sep 11, 2017

1	Installation	3
1.1	Prerequisites	3
1.2	Downloading and Installing the SDK	3
1.3	Configuring the SDK	4
1.4	Troubleshooting	5
2	Configuration	7
3	Forward Compatibility	9
4	Backward Compatibility	11
5	Quickstart	13
5.1	CRUD operations and Pagination	13
5.2	Working with Bytes	15
5.3	Next Steps	16
6	Parallel Operations	17
7	Uploading Large Objects	19
8	Raw Requests	21
8.1	Creating a Signer	21
8.2	Using the Signer	21
9	API Reference	23
9.1	Core Services	23
9.2	Identity	124
9.3	Load Balancer	161
9.4	Object Storage	200
9.5	Base Client	217
9.6	Config	218
9.7	Exceptions	218
9.8	Signing	219
9.9	Utilities	219
9.10	Request	220
9.11	Response	220
10	Contributions	221

11 Notifications	223
12 License	225
12.1 UPL	225
12.2 Apache	226
13 Questions or Feedback	229
Python Module Index	231

This is the public Python SDK for Oracle Cloud Infrastructure. Python 2.7+ and 3.5+ are supported.

```
>>> import oci
>>> config = oci.config.from_file(
...     "~/oci/config",
...     "integ-beta-profile")
>>> identity = oci.identity.IdentityClient(config)
>>> user = identity.get_user(config["user"]).data
>>> print(user)
{
  "compartment_id": "ocidl.tenancy.oc1...",
  "description": "Integration testing user [BETA]",
  "id": "ocidl.user.oc1...",
  "inactive_status": null,
  "lifecycle_state": "ACTIVE",
  "name": "testing+integ-beta@corp.com",
  "time_created": "2016-08-30T23:46:44.680000+00:00"
}
```

To get started, head over to the [installation instructions](#) or see more examples in the [quickstart](#) section.

Note: The `oraclebmc` package is deprecated and will be removed in March 2018. Please check the [Backward Compatibility](#) section if you are using `oraclebmc`.

Installation

This topic describes how to install, configure, and use the Oracle Cloud Infrastructure Python SDK. The Python SDK supports operations for the following services:

- Identity and Access Management Service
- Core Services (Networking Service, Compute Service, and Block Volume Service)
- Object Storage Service

Prerequisites

- An Oracle Cloud Infrastructure account
- A user created in that account, in a group with a policy that grants the desired permissions. This can be a user for yourself, or another person/system that needs to call the API. For an example of how to set up a new user, group, compartment, and policy, see [Adding Users](#) in the Getting Started Guide. For a list of other typical Oracle Cloud Infrastructure policies, see [Common Policies](#) in the User Guide.
- Python version 2.7.5 or 3.5 or later, running on Mac, Windows, or Linux.
- The Python SDK uses the [cryptography.io](#) library, which has its own additional [build requirements](#).
- A keypair used for signing API requests, with the public key uploaded to Oracle. Only the user calling the API should be in possession of the private key. (For more information, see [Configuring the SDK](#).)

Downloading and Installing the SDK

You can install the Python SDK through the Python Package Index (PyPI), or alternatively through GitHub.

PyPi

To install from PyPI:

Use the following command:

```
pip install oci
```

GitHub

To install from GitHub:

1. Download the SDK from [GitHub](#). The download is a zip containing a whl file and documentation.

2. Extract the files from the zip.
3. Use the following command to install the SDK:

```
pip install oci-*-py2.py3-none-any.whl
```

Note: If you're unable to install the whl file, make sure pip is up to date. Use `pip install -U pip` and then try to install the whl file again.

Virtual environment (Optional)

Although optional, Oracle recommends that you run the SDK in a virtual environment with `virtualenv`.

With Linux, it's usually in a separate package from the main Python package. If you need to install `virtualenv`, use `pip install virtualenv`. To create and activate a virtual environment:

```
virtualenv <environment name>
. <environment name>/bin/activate
```

For example:

```
virtualenv oci_sdk_env
. oci_sdk_env/bin/activate
```

Configuring the SDK

Before using the SDK, you must set up your config file with the required credentials. For instructions, see [SDK and Tool Configuration](#) in the User Guide.

Verify OpenSSL Version

The supported version of OpenSSL for the Python SDK is version 1.0.1 or newer. Run the following command to find out the version of OpenSSL that you have:

```
python -c "import ssl; print(ssl.OPENSSL_VERSION)"
```

If the version is lower than 1.0.1, run the following command to bypass the version issue:

```
pip install requests[security]==2.11.1
```

This command instructs the `requests` library used by the Python SDK to use the version of OpenSSL that is bundled with the `cryptography` library used by the SDK.

Note: If you don't want to use `requests[security]` you can update OpenSSL as you normally would. For example, on OS X, use Homebrew to update OpenSSL using the following commands:

```
brew update
brew install openssl
brew install python
```


Troubleshooting

You might encounter issues when installing Python or the SDK, or using the SDK itself.

Service Errors

Any operation resulting in a service error will cause an exception of type `oci.exceptions.ServiceError` to be thrown by the SDK. For information about common service errors returned by OCI, see [API Errors](#).

SSL/TLS or Certificate Issues

When trying to use the SDK, if you get an exception related to SSL/TLS or certificates/certificate validation, see the command for installing `requests[security]` in *Verify OpenSSL Version*.

Configuration

oci uses a simple dict to build clients and other components. You can build these manually, or oci can parse and validate a config file.

Using the default configuration location `~/.oci/config` you can use `config.from_file()` to load any profile. By default, the `DEFAULT` profile is used:

```
>>> from oci.config import from_file
>>> config = from_file()

# Using a different profile from the default location
>>> config = from_file(profile_name="integ-beta")

# Using the default profile from a different file
>>> config = from_file(file_location="~/.oci/config.prod")
```

Since `config` is a dict, you can also build it manually and check it with `config.validate_config()`:

```
import os
from myproject import testrunner
user_ocid = os.environ["USER_OCID"]
key_file = key_for(user_ocid)

config = {
    "user": user_ocid,
    "key_file": key_file,
    "fingerprint": calc_fingerprint(key_file),
    "tenancy": testrunner.tenancy,
    "region": testrunner.region
}

from oci.config import validate_config
validate_config(config)
```

See also:

The [SDK and Tool Configuration](#) page has a full description of the required and supported options. These are supported across the SDKs, so if you've already set this file up for the Ruby or Java SDKs, you're all set.

Forward Compatibility

Some response fields are enum-typed. In the future, individual services may return values not covered by existing enums for that field. To address this possibility, every enum-type response field has an additional value named “UNKNOWN_ENUM_VALUE”. If a service returns a value that is not recognized by your version of the SDK, then the response field will be set to this value. Please ensure that your code handles the “UNKNOWN_ENUM_VALUE” case if you have conditional logic based on an enum-typed field.

Backward Compatibility

The top level namespace / package name for the Python SDK has been changed from `oraclebmc` to `oci`, so all of the documentation now references `oci`. If you are using the `oraclebmc` package you should continue to reference `oraclebmc` in your code and when interpreting the documentation you should replace `oci` with `oraclebmc` (i.e. if there is a class defined in the docs as `oci.base_client.BaseClient` in the `oraclebmc` package this class will be called `oraclebmc.base_client.BaseClient`).

Note: The `oraclebmc` package is deprecated and will be removed in March 2018. Please upgrade to the `oci` package to avoid interruption at that time.

Quickstart

Clients only require a valid config object:

```
>>> from oci.identity import IdentityClient
>>> identity = IdentityClient(config)
```

CRUD operations and Pagination

Creating entities

Let's create a new user and group, and add the user to the group. Then we'll list all users in the tenancy, and finally clean up the user and group we created.

First, we'll need to create a valid config object and service client. If you haven't set up a config file, head over to the *Configuration* section to create one. We'll use the default location `~/.oci/config` and default profile name `DEFAULT` to create an Identity client. Since we'll be using the root compartment (or tenancy) for most operations, let's also extract that from the config object:

```
>>> import oci
>>> config = oci.config.from_file()
>>> identity = oci.identity.IdentityClient(config)
>>> compartment_id = config["tenancy"]
```

Next we'll need to populate an instance of the `CreateGroupDetails` model with our request, and then send it:

```
>>> from oci.identity.models import CreateGroupDetails
>>> request = CreateGroupDetails()
>>> request.compartment_id = compartment_id
>>> request.name = "my-test-group"
>>> request.description = "Created with the Python SDK"

>>> group = identity.create_group(request)
>>> print(group.data.id)
"Id": "ocid1.group.oc1..aaaaaaaakib..."
```

Creating a user is very similar:

```
>>> from oci.identity.models import CreateUserDetails
>>> request = CreateUserDetails()
>>> request.compartment_id = compartment_id
```

```
>>> request.name = "my-test-user"
>>> request.description = "Created with the Python SDK"
>>> user = identity.create_user(request)
>>> print(user.data.id)
"ocidl.user.oc1..aaaaaaaamkym..."
```

Using the ids from the group and user above, we can add the user to the group:

```
>>> from oci.identity.models import AddUserToGroupDetails
>>> request = AddUserToGroupDetails()
>>> request.group_id = group.data.id
>>> request.user_id = user.data.id
>>> response = identity.add_user_to_group(request)
>>> print(response.status)
200
```

Listing with Pagination

List operations use pagination to limit the size of each response. The Python SDK exposes the pagination values through the `has_next_page` and `next_page` attributes on each response. For example, listing users in the root compartment:

```
>>> first_page = identity.list_users(compartment_id=compartment_id)
>>> len(first_page.data)
100
>>> first_page.has_next_page
True
>>> first_page.next_page
'AAAAAAAAAAHNo_rjHo6xZPxHLZZ020jMio...'
```

Even though a response includes a next page, there may not be more results. The last page will return an empty list, and will not have a `next_page` token.

Here's a very simple way to paginate a call:

```
def paginate(operation, *args, **kwargs):
    while True:
        response = operation(*args, **kwargs)
        for value in response.data:
            yield value
        kwargs["page"] = response.next_page
        if not response.has_next_page:
            break
```

To iterate over all users, the call is now:

```
>>> for user in paginate(
...     identity.list_users,
...     compartment_id=compartment_id):
...     print(user)
```

This `paginate` function will work for any list call, but will not include the response metadata, such as headers, HTTP status code, or request id.

Deleting entities

Now to clean up the entities we created. Users can't be deleted if they're still part of a group, and groups can't be deleted if they still have users. So we need to use `identity.remove_user_from_group`, which takes a `user_group_membership_id`. Because users and groups can have any number of relationships, we'll use `list_user_group_memberships` and provide **both** optional parameters `user_id` and `group_id` to constrain the result set:

```
>>> memberships = identity.list_user_group_memberships(
...     compartment_id=compartment_id,
...     user_id=user.data.id,
...     group_id=group.data.id)
# There can never be more than one membership for a unique user/group combination
>>> assert len(memberships.data) == 1
>>> membership_id = memberships.data[0].id
```

Finally, we can remove the user from the group, and delete both resources. Here we're using `response.status` to make sure the delete responded with 204:

```
>>> identity.remove_user_from_group(
...     user_group_membership_id=membership_id).status
204
>>> identity.delete_user(user_id=user.data.id).status
204
>>> identity.delete_group(group_id=group.data.id).status
204
```

Working with Bytes

When using object storage, you'll need to provide a namespace, in addition to your compartment id:

```
>>> object_storage = oci.object_storage.ObjectStorageClient(config)
>>> namespace = object_storage.get_namespace().data
```

To upload an object, we'll create a bucket:

```
>>> from oci.object_storage.models import CreateBucketDetails
>>> request = CreateBucketDetails()
>>> request.compartment_id = compartment_id
>>> request.name = "MyTestBucket"
>>> bucket = object_storage.create_bucket(namespace, request)
>>> bucket.data.etag
'5281759f-60bb-4b93-8676-f8d141b5f211'
```

Now we can upload arbitrary bytes:

```
>>> my_data = b"Hello, World!"
>>> obj = object_storage.put_object(
...     namespace,
...     bucket.data.name,
...     "my-object-name",
...     my_data)
```

And to get it back:

```
>>> same_obj = object_storage.get_object(
...     namespace,
...     bucket.data.name,
...     "my-object-name")
... same_obj.data
<Response [200]>
... same_obj.data.content
b'Hello, World!'
```

Next Steps

Next, head to the [User Guides](#) or jump right into the [API Reference](#) to explore the available operations for each service, and their parameters. Additional Python examples can be found on [GitHub](#).

Note: The Python SDK uses lowercase_with_underscores for operations and parameters. For example, the [ListApiKeys](#) operation is called with `IdentityClient.list_api_keys` and its parameter `userId` is translated to `user_id`.

Parallel Operations

The Python SDK supports parallel requests to Oracle Cloud Infrastructure. For example, the [object storage upload example](#) shows how multiple processes can be used to upload files to object storage.

Uploading Large Objects

The Object Storage service supports multipart uploads to make large object uploads easier by splitting the large object into parts. The Python SDK supports raw multipart upload operations for advanced use cases, as well as a higher-level upload class that uses the multipart upload APIs. [Managing Multipart Uploads](#) provides links to the APIs used for raw multipart upload operations. Higher-level uploads can be performed using the UploadManager. The UploadManger will: split a large object into parts for you, upload the parts in parallel, and then recombine and commit the parts as a single object in Object Storage.

The [UploadObject](#) example shows how UploadManager can be used to upload files to object storage.

Raw Requests

The Python SDK exposes a custom `requests.auth.AuthBase` which you can use to sign non-standard calls. This can be helpful if you need to make a OCI- authenticated request to an alternate endpoint or to a OCI API not yet supported in the SDK.

Creating a Signer

Constructing a Signer instance requires a few pieces of information. By default, the SDK uses the values in the config file at `~/.oci/config`. You can manually specify the required fields, or use a config loader to pull in the values from a file:

```
from oci.signer import Signer
auth = Signer(
    tenancy='ocidl.tenancy.oc1..aaaaaaaa[...]',
    user='ocidl.user.oc1..aaaaaaaa[...]',
    fingerprint='20:3b:97:13:55:1c:[...]',
    private_key_file_location='~/.oci/oci_api_key.pem',
    pass_phrase='hunter2' # optional
)

# Or load directly from a file
from oci.config import from_file
config = from_file('~/.oci/config')
auth = Signer(
    tenancy=config['tenancy'],
    user=config['user'],
    fingerprint=config['fingerprint'],
    private_key_file_location=config['key_file'],
    pass_phrase=config['pass_phrase']
)
```

Using the Signer

Once you have an instance of the auth handler, simply include it as the `auth=` param when using Requests.

```
import requests
```

```
url = 'https://iaas.us-phoenix-1.oraclecloud.com/20160918/instances[...]'
response = requests.get(url, auth=auth)
```

Remember that the result will come back in its raw form and is not unpacked into a model instance. You will need to handle the (de)serialization yourself.

The following creates a new user by talking to the identity endpoint:

```
endpoint = 'https://identity.us-phoenix-1.oraclecloud.com/20160918/users/'
body = {
    'compartmentId': config['tenancy'], # root compartment
    'name': 'TestUser',
    'description': 'Created with a raw request'
}

response = requests.post(endpoint, json=body, auth=auth)

response.raise_for_status()
print(response.json()['id'])
```

API Reference

Core Services

Clients

Block Storage

class `oci.core.blockstorage_client.BlockstorageClient` (*config*)

create_volume (*create_volume_details*, ***kwargs*)

CreateVolume Creates a new volume in the specified compartment. Volumes can be created in sizes ranging from 50 GB (51200 MB) to 2 TB (2097152 MB), in 1 GB (1024 MB) increments. By default, volumes are 1 TB (1048576 MB). For general information about block volumes, see [Overview of Block Volume Service](#).

A volume and instance can be in separate compartments but must be in the same Availability Domain. For information about access control and compartments, see [Overview of the IAM Service](#). For information about Availability Domains, see [Regions and Availability Domains](#). To get a list of Availability Domains, use the *ListAvailabilityDomains* operation in the Identity and Access Management Service API.

You may optionally specify a *display name* for the volume, which is simply a friendly name or description. It does not have to be unique, and you can change it. Avoid entering confidential information.

Parameters

- **create_volume_details** (`CreateVolumeDetails`) – (required) Request to create a new volume.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *Volume*

Return type *Response*

create_volume_backup (*create_volume_backup_details*, ***kwargs*)

CreateVolumeBackup Creates a new backup of the specified volume. For general information about volume backups, see [Overview of Block Volume Service Backups](#)

When the request is received, the backup object is in a `REQUEST_RECEIVED` state. When the data is imaged, it goes into a `CREATING` state. After the backup is fully uploaded to the cloud, it goes into an `AVAILABLE` state.

Parameters

- **create_volume_backup_details** (`CreateVolumeBackupDetails`) – (required) Request to create a new backup of given volume.
- **opc_retry_token** (`str`) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *VolumeBackup*

Return type *Response*

delete_volume (`volume_id`, `**kwargs`)

DeleteVolume Deletes the specified volume. The volume cannot have an active connection to an instance. To disconnect the volume from a connected instance, see [Disconnecting From a Volume](#). **Warning:** All data on the volume will be permanently lost when the volume is deleted.

Parameters

- **volume_id** (`str`) – (required) The OCID of the volume.
- **if_match** (`str`) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type `None`

Return type *Response*

delete_volume_backup (`volume_backup_id`, `**kwargs`)

DeleteVolumeBackup Deletes a volume backup.

Parameters

- **volume_backup_id** (`str`) – (required) The OCID of the volume backup.
- **if_match** (`str`) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type `None`

Return type *Response*

get_volume (`volume_id`, `**kwargs`)

GetVolume Gets information for the specified volume.

Parameters **volume_id** (`str`) – (required) The OCID of the volume.

Returns A *Response* object with data of type *Volume*

Return type *Response*

get_volume_backup (`volume_backup_id`, `**kwargs`)

GetVolumeBackup Gets information for the specified volume backup.

Parameters `volume_backup_id` (*str*) – (required) The OCID of the volume backup.

Returns A *Response* object with data of type *VolumeBackup*

Return type *Response*

list_volume_backups (*compartment_id*, ***kwargs*)

ListVolumeBackups Lists the volume backups in the specified compartment. You can filter the results by volume.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **volume_id** (*str*) – (optional) The OCID of the volume.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.
Example: 500
- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Returns A *Response* object with data of type list of *VolumeBackup*

Return type *Response*

list_volumes (*compartment_id*, ***kwargs*)

ListVolumes Lists the volumes in the specified compartment and Availability Domain.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **availability_domain** (*str*) – (optional) The name of the Availability Domain.
Example: *Uocm:PHX-AD-1*
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.
Example: 500
- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Returns A *Response* object with data of type list of *Volume*

Return type *Response*

update_volume (*volume_id*, *update_volume_details*, ***kwargs*)

UpdateVolume Updates the specified volume’s display name. Avoid entering confidential information.

Parameters

- **volume_id** (*str*) – (required) The OCID of the volume.
- **update_volume_details** (*UpdateVolumeDetails*) – (required) Update volume’s display name. Avoid entering confidential information.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *Volume*

Return type *Response*

update_volume_backup (*volume_backup_id*, *update_volume_backup_details*, ***kwargs*)

UpdateVolumeBackup Updates the display name for the specified volume backup. Avoid entering confidential information.

Parameters

- **volume_backup_id** (*str*) – (required) The OCID of the volume backup.
- **update_volume_backup_details** (*UpdateVolumeBackupDetails*) – (required) Update volume backup fields
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *VolumeBackup*

Return type *Response*

Compute

`class oci.core.compute_client.ComputeClient` (*config*)

attach_vnic (*attach_vnic_details*, ***kwargs*)

AttachVnic Creates a secondary VNIC and attaches it to the specified instance. For more information about secondary VNICs, see [Virtual Network Interface Cards \(VNICs\)](#).

Parameters

- **attach_vnic_details** (*AttachVnicDetails*) – (required) Attach VNIC details.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *VnicAttachment*

Return type *Response*

attach_volume (*attach_volume_details*, ***kwargs*)

AttachVolume Attaches the specified storage volume to the specified instance.

Parameters

- **attach_volume_details** (*AttachVolumeDetails*) – (required) Attach volume request
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *VolumeAttachment*

Return type *Response*

capture_console_history (*capture_console_history_details*, ***kwargs*)

CaptureConsoleHistory Captures the most recent serial console data (up to a megabyte) for the specified instance.

The *CaptureConsoleHistory* operation works with the other console history operations as described below.

1. Use *CaptureConsoleHistory* to request the capture of up to a megabyte of the most recent console history. This call returns a *ConsoleHistory* object. The object will have a state of REQUESTED. 2. Wait for the capture operation to succeed by polling *GetConsoleHistory* with the identifier of the console history metadata. The state of the *ConsoleHistory* object will go from REQUESTED to GETTING-HISTORY and then SUCCEEDED (or FAILED). 3. Use *GetConsoleHistoryContent* to get the actual console history data (not the metadata). 4. Optionally, use *DeleteConsoleHistory* to delete the console history metadata and the console history data.

Parameters

- **capture_console_history_details** (*CaptureConsoleHistoryDetails*) – (required) Console history details
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *ConsoleHistory*

Return type *Response*

create_image (*create_image_details*, ***kwargs*)

CreateImage Creates a boot disk image for the specified instance or imports an exported image from the Oracle Bare Metal Cloud Object Storage Service.

When creating a new image, you must provide the OCID of the instance you want to use as the basis for the image, and the OCID of the compartment containing that instance. For more information about images, see [Managing Custom Images](#).

When importing an exported image from the Object Storage Service, you specify the source information in *image_source_details()*.

When importing an image based on the namespace, bucket name, and object name, use *image_source_via_object_storage_tuple_details()*.

When importing an image based on the Object Storage Service URL, use *image_source_via_object_storage_uri_details()*. See [Object Storage URLs and pre-authenticated requests](#) for constructing URLs for image import/export.

For more information about importing exported images, see [Image Import/Export](#).

You may optionally specify a *display name* for the image, which is simply a friendly name or description. It does not have to be unique, and you can change it. See *update_image()*. Avoid entering confidential information.

Parameters

- **create_image_details** (*CreateImageDetails*) – (required) Image creation details
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due

to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *Image*

Return type *Response*

create_instance_console_connection (*create_instance_console_connection_details*,
***kwargs*)

CreateInstanceConsoleConnection Create a console connection for an instance.

Parameters

- **create_instance_console_connection_details** (*CreateInstanceConsoleConnectionDetails*) – (required) Request object for creating an InstanceConsoleConnection
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *InstanceConsoleConnection*

Return type *Response*

delete_console_history (*instance_console_history_id*, ***kwargs*)

DeleteConsoleHistory Deletes the specified console history metadata and the console history data.

Parameters

- **instance_console_history_id** (*str*) – (required) The OCID of the console history.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type None

Return type *Response*

delete_image (*image_id*, ***kwargs*)

DeleteImage Deletes an image.

Parameters

- **image_id** (*str*) – (required) The OCID of the image.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type None

Return type *Response*

delete_instance_console_connection (*instance_console_connection_id*, ***kwargs*)

DeleteInstanceConsoleConnection Delete the console connection for an instance

Parameters

- **instance_console_connection_id** (*str*) – (required) The OCID of the instance console connection
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type None

Return type *Response*

detach_vnic (*vnic_attachment_id*, ***kwargs*)

DetachVnic Detaches and deletes the specified secondary VNIC. This operation cannot be used on the instance’s primary VNIC. When you terminate an instance, all attached VNICs (primary and secondary) are automatically detached and deleted.

Parameters

- **vnic_attachment_id** (*str*) – (required) The OCID of the VNIC attachment.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type None

Return type *Response*

detach_volume (*volume_attachment_id*, ***kwargs*)

DetachVolume Detaches a storage volume from an instance. You must specify the OCID of the volume attachment.

This is an asynchronous operation. The attachment’s *lifecycleState* will change to DETACHING temporarily until the attachment is completely removed.

Parameters

- **volume_attachment_id** (*str*) – (required) The OCID of the volume attachment.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type None

Return type *Response*

export_image (*image_id*, *export_image_details*, ***kwargs*)

ExportImage Exports the specified image to the Oracle Bare Metal Cloud Object Storage Service. You can use the Object Storage Service URL, or the namespace, bucket name, and object name when specifying the location to export to.

For more information about exporting images, see [Image Import/Export](#).

To perform an image export, you need write access to the Object Storage Service bucket for the image, see [Let Users Write Objects to Object Storage Buckets](#).

See [Object Storage URLs and pre-authenticated requests](#) for constructing URLs for image import/export.

Parameters

- **image_id** (*str*) – (required) The OCID of the image.

- **export_image_details** (`ExportImageDetails`) – (required) Details for the image export.
- **opc_retry_token** (`str`) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).
- **if_match** (`str`) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *Image*

Return type *Response*

get_console_history (`instance_console_history_id`, `**kwargs`)

`GetConsoleHistory` Shows the metadata for the specified console history. See `capture_console_history()` for details about using the console history operations.

Parameters **instance_console_history_id** (`str`) – (required) The OCID of the console history.

Returns A *Response* object with data of type *ConsoleHistory*

Return type *Response*

get_console_history_content (`instance_console_history_id`, `**kwargs`)

`GetConsoleHistoryContent` Gets the actual console history data (not the metadata). See `capture_console_history()` for details about using the console history operations.

Parameters

- **instance_console_history_id** (`str`) – (required) The OCID of the console history.
- **offset** (`int`) – (optional) Offset of the snapshot data to retrieve.
- **length** (`int`) – (optional) Length of the snapshot data to retrieve.

Returns A *Response* object with data of type bytes

Return type *Response*

get_image (`image_id`, `**kwargs`)

`GetImage` Gets the specified image.

Parameters **image_id** (`str`) – (required) The OCID of the image.

Returns A *Response* object with data of type *Image*

Return type *Response*

get_instance (`instance_id`, `**kwargs`)

`GetInstance` Gets information about the specified instance.

Parameters **instance_id** (`str`) – (required) The OCID of the instance.

Returns A *Response* object with data of type *Instance*

Return type *Response*

get_instance_console_connection (`instance_console_connection_id`, `**kwargs`)

`GetInstanceConsoleConnection` Get the details of an instance console connection

Parameters `instance_console_connection_id` (*str*) – (required) The OCID of the instance console connection

Returns A *Response* object with data of type *InstanceConsoleConnection*

Return type *Response*

get_vnic_attachment (*vnic_attachment_id*, ***kwargs*)

GetVnicAttachment Gets the information for the specified VNIC attachment.

Parameters `vnic_attachment_id` (*str*) – (required) The OCID of the VNIC attachment.

Returns A *Response* object with data of type *VnicAttachment*

Return type *Response*

get_volume_attachment (*volume_attachment_id*, ***kwargs*)

GetVolumeAttachment Gets information about the specified volume attachment.

Parameters `volume_attachment_id` (*str*) – (required) The OCID of the volume attachment.

Returns A *Response* object with data of type *VolumeAttachment*

Return type *Response*

get_windows_instance_initial_credentials (*instance_id*, ***kwargs*)

GetWindowsInstanceInitialCredentials Gets the generated credentials for the instance. Only works for Windows instances. The returned credentials are only valid for the initial login.

Parameters `instance_id` (*str*) – (required) The OCID of the instance.

Returns A *Response* object with data of type *InstanceCredentials*

Return type *Response*

instance_action (*instance_id*, *action*, ***kwargs*)

InstanceAction Performs one of the power actions (start, stop, softreset, or reset) on the specified instance.

start - power on

stop - power off

softreset - ACPI shutdown and power on

reset - power off and power on

Note that the **stop** state has no effect on the resources you consume. Billing continues for instances that you stop, and related resources continue to apply against any relevant quotas. You must terminate an instance (`terminate_instance()`) to remove its resources from billing and quotas.

Parameters

- **instance_id** (*str*) – (required) The OCID of the instance.
- **action** (*str*) – (required) The action to perform on the instance.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a

previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type *Instance*

Return type *Response*

launch_instance (*launch_instance_details*, ***kwargs*)

LaunchInstance Creates a new instance in the specified compartment and the specified Availability Domain. For general information about instances, see [Overview of the Compute Service](#).

For information about access control and compartments, see [Overview of the IAM Service](#).

For information about Availability Domains, see [Regions and Availability Domains](#). To get a list of Availability Domains, use the *ListAvailabilityDomains* operation in the Identity and Access Management Service API.

All Oracle Bare Metal Cloud Services resources, including instances, get an Oracle-assigned, unique ID called an Oracle Cloud Identifier (OCID). When you create a resource, you can find its OCID in the response. You can also retrieve a resource's OCID by using a List API operation on that resource type, or by viewing the resource in the Console.

When you launch an instance, it is automatically attached to a virtual network interface card (VNIC), called the *primary VNIC*. The VNIC has a private IP address from the subnet's CIDR. You can either assign a private IP address of your choice or let Oracle automatically assign one. You can choose whether the instance has a public IP address. To retrieve the addresses, use the `list_vnic_attachments()` operation to get the VNIC ID for the instance, and then call `get_vnic()` with the VNIC ID.

You can later add secondary VNICs to an instance. For more information, see [Virtual Network Interface Cards \(VNICs\)](#).

Parameters

- **launch_instance_details** (*LaunchInstanceDetails*) – (required) Instance details
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *Instance*

Return type *Response*

list_console_histories (*compartment_id*, ***kwargs*)

ListConsoleHistories Lists the console history metadata for the specified compartment or instance.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **availability_domain** (*str*) – (optional) The name of the Availability Domain.
Example: *Ucm:PHX-AD-1*
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.
Example: *500*
- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

- **instance_id** (*str*) – (optional) The OCID of the instance.

Returns A *Response* object with data of type list of *ConsoleHistory*

Return type *Response*

list_images (*compartment_id*, ***kwargs*)

ListImages Lists the available images in the specified compartment. For more information about images, see [Managing Custom Images](#).

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **display_name** (*str*) – (optional) A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Example: *My new resource*

- **operating_system** (*str*) – (optional) The image's operating system.

Example: *Oracle Linux*

- **operating_system_version** (*str*) – (optional) The image's operating system version.

Example: *7.2*

- **limit** (*int*) – (optional) The maximum number of items to return in a paginated "List" call.

Example: *500*

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous "List" call.

Returns A *Response* object with data of type list of *Image*

Return type *Response*

list_instance_console_connections (*compartment_id*, ***kwargs*)

ListInstanceConsoleConnections Lists the console connections for the specified compartment or instance that have not been deleted.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **instance_id** (*str*) – (optional) The OCID of the instance.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated "List" call.

Example: *500*

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous "List" call.

Returns A *Response* object with data of type list of *InstanceConsoleConnection*

Return type *Response*

list_instances (*compartment_id*, ***kwargs*)

ListInstances Lists the instances in the specified compartment and the specified Availability Domain. You can filter the results by specifying an instance name (the list will include all the identically-named instances in the compartment).

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **availability_domain** (*str*) – (optional) The name of the Availability Domain.
Example: *Uocm:PHX-AD-1*
- **display_name** (*str*) – (optional) A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.
Example: *My new resource*
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.
Example: *500*
- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Returns A *Response* object with data of type list of *Instance*

Return type *Response*

list_shapes (*compartment_id*, ***kwargs*)

ListShapes Lists the shapes that can be used to launch an instance within the specified compartment. You can filter the list by compatibility with a specific image.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **availability_domain** (*str*) – (optional) The name of the Availability Domain.
Example: *Uocm:PHX-AD-1*
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.
Example: *500*
- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.
- **image_id** (*str*) – (optional) The OCID of an image.

Returns A *Response* object with data of type list of *Shape*

Return type *Response*

list_vnic_attachments (*compartment_id*, ***kwargs*)

ListVnicAttachments Lists the VNIC attachments in the specified compartment. A VNIC attachment resides in the same compartment as the attached instance. The list can be filtered by instance, VNIC, or Availability Domain.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **availability_domain** (*str*) – (optional) The name of the Availability Domain.
Example: *Uocm:PHX-AD-1*
- **instance_id** (*str*) – (optional) The OCID of the instance.

- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: *500*

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.
- **vnic_id** (*str*) – (optional) The OCID of the VNIC.

Returns A *Response* object with data of type list of *VnicAttachment*

Return type *Response*

list_volume_attachments (*compartment_id*, ***kwargs*)

ListVolumeAttachments Lists the volume attachments in the specified compartment. You can filter the list by specifying an instance OCID, volume OCID, or both.

Currently, the only supported volume attachment type is *IScsiVolumeAttachment*.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **availability_domain** (*str*) – (optional) The name of the Availability Domain.
Example: *Uocm:PHX-AD-1*
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.
Example: *500*
- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.
- **instance_id** (*str*) – (optional) The OCID of the instance.
- **volume_id** (*str*) – (optional) The OCID of the volume.

Returns A *Response* object with data of type list of *VolumeAttachment*

Return type *Response*

terminate_instance (*instance_id*, ***kwargs*)

TerminateInstance Terminates the specified instance. Any attached VNICs and volumes are automatically detached when the instance terminates.

This is an asynchronous operation. The instance’s *lifecycleState* will change to *TERMINATING* temporarily until the instance is completely removed.

Parameters

- **instance_id** (*str*) – (required) The OCID of the instance.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *None*

Return type *Response*

update_image (*image_id*, *update_image_details*, ***kwargs*)

UpdateImage Updates the display name of the image. Avoid entering confidential information.

Parameters

- **image_id** (*str*) – (required) The OCID of the image.
- **update_image_details** (*UpdateImageDetails*) – (required) Updates the image display name field. Avoid entering confidential information.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *Image*

Return type *Response*

update_instance (*instance_id*, *update_instance_details*, ***kwargs*)

UpdateInstance Updates the display name of the specified instance. Avoid entering confidential information. The OCID of the instance remains the same.

Parameters

- **instance_id** (*str*) – (required) The OCID of the instance.
- **update_instance_details** (*UpdateInstanceDetails*) – (required) Update instance fields
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *Instance*

Return type *Response*

Virtual Network

class `oci.core.virtual_network_client.VirtualNetworkClient` (*config*)

create_cpe (*create_cpe_details*, ***kwargs*)

CreateCpe Creates a new virtual Customer-Premises Equipment (CPE) object in the specified compartment. For more information, see [IPSec VPNs](#).

For the purposes of access control, you must provide the OCID of the compartment where you want the CPE to reside. Notice that the CPE doesn’t have to be in the same compartment as the IPSec connection or other Networking Service components. If you’re not sure which compartment to use, put the CPE in

the same compartment as the DRG. For more information about compartments and access control, see [Overview of the IAM Service](#). For information about OCIDs, see [Resource Identifiers](#).

You must provide the public IP address of your on-premises router. See [Configuring Your On-Premises Router for an IPSec VPN](#).

You may optionally specify a *display name* for the CPE, otherwise a default is provided. It does not have to be unique, and you can change it. Avoid entering confidential information.

Parameters

- **create_cpe_details** (*CreateCpeDetails*) – (required) Details for creating a CPE.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *Cpe*

Return type *Response*

create_cross_connect (*create_cross_connect_details*, ***kwargs*)

CreateCrossConnect Creates a new cross-connect. Oracle recommends you create each cross-connect in a *CrossConnectGroup* so you can use link aggregation with the connection.

After creating the *CrossConnect* object, you need to go the *FastConnect* location and request to have the physical cable installed. For more information, see [FastConnect Overview](#).

For the purposes of access control, you must provide the OCID of the compartment where you want the cross-connect to reside. If you're not sure which compartment to use, put the cross-connect in the same compartment with your VCN. For more information about compartments and access control, see [Overview of the IAM Service](#). For information about OCIDs, see [Resource Identifiers](#).

You may optionally specify a *display name* for the cross-connect. It does not have to be unique, and you can change it. Avoid entering confidential information.

Parameters

- **create_cross_connect_details** (*CreateCrossConnectDetails*) – (required) Details to create a *CrossConnect*
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *CrossConnect*

Return type *Response*

create_cross_connect_group (*create_cross_connect_group_details*, ***kwargs*)

CreateCrossConnectGroup Creates a new cross-connect group to use with Oracle Bare Metal Cloud Services *FastConnect*. For more information, see [FastConnect Overview](#).

For the purposes of access control, you must provide the OCID of the compartment where you want the cross-connect group to reside. If you're not sure which compartment to use, put the cross-connect group in the same compartment with your VCN. For more information about compartments and access control, see [Overview of the IAM Service](#). For information about OCIDs, see [Resource Identifiers](#).

You may optionally specify a *display name* for the cross-connect group. It does not have to be unique, and you can change it. Avoid entering confidential information.

Parameters

- **create_cross_connect_group_details** (`CreateCrossConnectGroupDetails`) – (required) Details to create a `CrossConnectGroup`
- **opc_retry_token** (`str`) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *CrossConnectGroup*

Return type *Response*

create_dhcp_options (*create_dhcp_details*, ***kwargs*)

`CreateDhcpOptions` Creates a new set of DHCP options for the specified VCN. For more information, see `DhcpOptions`.

For the purposes of access control, you must provide the OCID of the compartment where you want the set of DHCP options to reside. Notice that the set of options doesn't have to be in the same compartment as the VCN, subnets, or other Networking Service components. If you're not sure which compartment to use, put the set of DHCP options in the same compartment as the VCN. For more information about compartments and access control, see [Overview of the IAM Service](#). For information about OCIDs, see [Resource Identifiers](#).

You may optionally specify a *display name* for the set of DHCP options, otherwise a default is provided. It does not have to be unique, and you can change it. Avoid entering confidential information.

Parameters

- **create_dhcp_details** (`CreateDhcpDetails`) – (required) Request object for creating a new set of DHCP options.
- **opc_retry_token** (`str`) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *DhcpOptions*

Return type *Response*

create_drg (*create_drg_details*, ***kwargs*)

`CreateDrg` Creates a new Dynamic Routing Gateway (DRG) in the specified compartment. For more information, see [Dynamic Routing Gateways \(DRGs\)](#).

For the purposes of access control, you must provide the OCID of the compartment where you want the DRG to reside. Notice that the DRG doesn't have to be in the same compartment as the VCN, the DRG attachment, or other Networking Service components. If you're not sure which compartment to use, put the DRG in the same compartment as the VCN. For more information about compartments and access control, see [Overview of the IAM Service](#). For information about OCIDs, see [Resource Identifiers](#).

You may optionally specify a *display name* for the DRG, otherwise a default is provided. It does not have to be unique, and you can change it. Avoid entering confidential information.

Parameters

- **create_drg_details** (*CreateDrgDetails*) – (required) Details for creating a DRG.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *Drg*

Return type *Response*

create_drg_attachment (*create_drg_attachment_details*, ***kwargs*)

CreateDrgAttachment Attaches the specified DRG to the specified VCN. A VCN can be attached to only one DRG at a time, and vice versa. The response includes a *DrgAttachment* object with its own OCID. For more information about DRGs, see [Dynamic Routing Gateways \(DRGs\)](#).

You may optionally specify a *display name* for the attachment, otherwise a default is provided. It does not have to be unique, and you can change it. Avoid entering confidential information.

For the purposes of access control, the DRG attachment is automatically placed into the same compartment as the VCN. For more information about compartments and access control, see [Overview of the IAM Service](#).

Parameters

- **create_drg_attachment_details** (*CreateDrgAttachmentDetails*) – (required) Details for creating a *DrgAttachment*.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *DrgAttachment*

Return type *Response*

create_internet_gateway (*create_internet_gateway_details*, ***kwargs*)

CreateInternetGateway Creates a new Internet Gateway for the specified VCN. For more information, see [Connectivity to the Internet](#).

For the purposes of access control, you must provide the OCID of the compartment where you want the Internet Gateway to reside. Notice that the Internet Gateway doesn't have to be in the same compartment as the VCN or other Networking Service components. If you're not sure which compartment to use, put the Internet Gateway in the same compartment with the VCN. For more information about compartments and access control, see [Overview of the IAM Service](#). For information about OCIDs, see [Resource Identifiers](#).

You may optionally specify a *display name* for the Internet Gateway, otherwise a default is provided. It does not have to be unique, and you can change it. Avoid entering confidential information.

For traffic to flow between a subnet and an Internet Gateway, you must create a route rule accordingly in the subnet's route table (for example, 0.0.0.0/0 > Internet Gateway). See `update_route_table()`.

You must specify whether the Internet Gateway is enabled when you create it. If it's disabled, that means no traffic will flow to/from the internet even if there's a route rule that enables that traffic. You can later use `update_internet_gateway()` to easily disable/enable the gateway without changing the route rule.

Parameters

- **create_internet_gateway_details** (`CreateInternetGatewayDetails`) – (required) Details for creating a new Internet Gateway.
- **opc_retry_token** (`str`) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *InternetGateway*

Return type *Response*

create_ip_sec_connection (`create_ip_sec_connection_details, **kwargs`)

CreateIPSecConnection Creates a new IPSec connection between the specified DRG and CPE. For more information, see [IPSec VPNs](#).

In the request, you must include at least one static route to the CPE object (you’re allowed a maximum of 10). For example: 10.0.8.0/16.

For the purposes of access control, you must provide the OCID of the compartment where you want the IPSec connection to reside. Notice that the IPSec connection doesn’t have to be in the same compartment as the DRG, CPE, or other Networking Service components. If you’re not sure which compartment to use, put the IPSec connection in the same compartment as the DRG. For more information about compartments and access control, see [Overview of the IAM Service](#). For information about OCIDs, see [Resource Identifiers](#).

You may optionally specify a *display name* for the IPSec connection, otherwise a default is provided. It does not have to be unique, and you can change it. Avoid entering confidential information.

After creating the IPSec connection, you need to configure your on-premises router with tunnel-specific information returned by `get_ip_sec_connection_device_config()`. For each tunnel, that operation gives you the IP address of Oracle’s VPN headend and the shared secret (that is, the pre-shared key). For more information, see [Configuring Your On-Premises Router for an IPSec VPN](#).

To get the status of the tunnels (whether they’re up or down), use `get_ip_sec_connection_device_status()`.

Parameters

- **create_ip_sec_connection_details** (`CreateIPSecConnectionDetails`) – (required) Details for creating an *IPSecConnection*.
- **opc_retry_token** (`str`) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *IPSecConnection*

Return type *Response*

create_private_ip (`create_private_ip_details, **kwargs`)

CreatePrivateIp Creates a secondary private IP for the specified VNIC. For more information about secondary private IPs, see [IP Addresses](#).

Parameters

- **create_private_ip_details** (`CreatePrivateIpDetails`) – (required) Create private IP details.
- **opc_retry_token** (`str`) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same

action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *PrivateIp*

Return type *Response*

create_route_table (*create_route_table_details*, ***kwargs*)

CreateRouteTable Creates a new route table for the specified VCN. In the request you must also include at least one route rule for the new route table. For information on the number of rules you can have in a route table, see [Service Limits](#). For general information about route tables in your VCN, see [Route Tables](#).

For the purposes of access control, you must provide the OCID of the compartment where you want the route table to reside. Notice that the route table doesn't have to be in the same compartment as the VCN, subnets, or other Networking Service components. If you're not sure which compartment to use, put the route table in the same compartment as the VCN. For more information about compartments and access control, see [Overview of the IAM Service](#). For information about OCIDs, see [Resource Identifiers](#).

You may optionally specify a *display name* for the route table, otherwise a default is provided. It does not have to be unique, and you can change it. Avoid entering confidential information.

Parameters

- **create_route_table_details** (*CreateRouteTableDetails*) – (required) Details for creating a new route table.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *RouteTable*

Return type *Response*

create_security_list (*create_security_list_details*, ***kwargs*)

CreateSecurityList Creates a new security list for the specified VCN. For more information about security lists, see [Security Lists](#). For information on the number of rules you can have in a security list, see [Service Limits](#).

For the purposes of access control, you must provide the OCID of the compartment where you want the security list to reside. Notice that the security list doesn't have to be in the same compartment as the VCN, subnets, or other Networking Service components. If you're not sure which compartment to use, put the security list in the same compartment as the VCN. For more information about compartments and access control, see [Overview of the IAM Service](#). For information about OCIDs, see [Resource Identifiers](#).

You may optionally specify a *display name* for the security list, otherwise a default is provided. It does not have to be unique, and you can change it. Avoid entering confidential information.

Parameters

- **create_security_list_details** (*CreateSecurityListDetails*) – (required) Details regarding the security list to create.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *SecurityList*

Return type *Response*

create_subnet (*create_subnet_details*, ***kwargs*)

CreateSubnet Creates a new subnet in the specified VCN. You can't change the size of the subnet after creation, so it's important to think about the size of subnets you need before creating them. For more information, see [VCNs and Subnets](#). For information on the number of subnets you can have in a VCN, see [Service Limits](#).

For the purposes of access control, you must provide the OCID of the compartment where you want the subnet to reside. Notice that the subnet doesn't have to be in the same compartment as the VCN, route tables, or other Networking Service components. If you're not sure which compartment to use, put the subnet in the same compartment as the VCN. For more information about compartments and access control, see [Overview of the IAM Service](#). For information about OCIDs, see [Resource Identifiers](#).

You may optionally associate a route table with the subnet. If you don't, the subnet will use the VCN's default route table. For more information about route tables, see [Route Tables](#).

You may optionally associate a security list with the subnet. If you don't, the subnet will use the VCN's default security list. For more information about security lists, see [Security Lists](#).

You may optionally associate a set of DHCP options with the subnet. If you don't, the subnet will use the VCN's default set. For more information about DHCP options, see [DHCP Options](#).

You may optionally specify a *display name* for the subnet, otherwise a default is provided. It does not have to be unique, and you can change it. Avoid entering confidential information.

You can also add a DNS label for the subnet, which is required if you want the Internet and VCN Resolver to resolve hostnames for instances in the subnet. For more information, see [DNS in Your Virtual Cloud Network](#).

Parameters

- **create_subnet_details** (*CreateSubnetDetails*) – (required) Details for creating a subnet.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *Subnet*

Return type *Response*

create_vcn (*create_vcn_details*, ***kwargs*)

CreateVcn Creates a new Virtual Cloud Network (VCN). For more information, see [VCNs and Subnets](#).

For the VCN you must specify a single, contiguous IPv4 CIDR block. Oracle recommends using one of the private IP address ranges specified in [RFC 1918](#) (10.0.0.0/8, 172.16/12, and 192.168/16). Example: 172.16.0.0/16. The CIDR block can range from /16 to /30, and it must not overlap with your on-premises network. You can't change the size of the VCN after creation.

For the purposes of access control, you must provide the OCID of the compartment where you want the VCN to reside. Consult an Oracle Bare Metal Cloud Services administrator in your organization if you're not sure which compartment to use. Notice that the VCN doesn't have to be in the same compartment as the subnets or other Networking Service components. For more information about compartments and access control, see [Overview of the IAM Service](#). For information about OCIDs, see [Resource Identifiers](#).

You may optionally specify a *display name* for the VCN, otherwise a default is provided. It does not have to be unique, and you can change it. Avoid entering confidential information.

You can also add a DNS label for the VCN, which is required if you want the instances to use the Interent and VCN Resolver option for DNS in the VCN. For more information, see [DNS in Your Virtual Cloud Network](#).

The VCN automatically comes with a default route table, default security list, and default set of DHCP options. The OCID for each is returned in the response. You can't delete these default objects, but you can change their contents (that is, change the route rules, security list rules, and so on).

The VCN and subnets you create are not accessible until you attach an Internet Gateway or set up an IPsec VPN or FastConnect. For more information, see [Overview of the Networking Service](#).

Parameters

- **create_vcn_details** (*CreateVcnDetails*) – (required) Details for creating a new VCN.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *Vcn*

Return type *Response*

create_virtual_circuit (*create_virtual_circuit_details*, ***kwargs*)

CreateVirtualCircuit Creates a new virtual circuit to use with Oracle Bare Metal Cloud Services FastConnect. For more information, see [FastConnect Overview](#).

For the purposes of access control, you must provide the OCID of the compartment where you want the virtual circuit to reside. If you're not sure which compartment to use, put the virtual circuit in the same compartment with the DRG it's using. For more information about compartments and access control, see [Overview of the IAM Service](#). For information about OCIDs, see [Resource Identifiers](#).

You may optionally specify a *display name* for the virtual circuit. It does not have to be unique, and you can change it. Avoid entering confidential information.

Important: When creating a virtual circuit, you specify a DRG for the traffic to flow through. Make sure you attach the DRG to your VCN and confirm the VCN's routing sends traffic to the DRG. Otherwise traffic will not flow. For more information, see [Route Tables](#).

Parameters

- **create_virtual_circuit_details** (*CreateVirtualCircuitDetails*) – (required) Details to create a VirtualCircuit.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *VirtualCircuit*

Return type *Response*

delete_cpe (*cpe_id*, ***kwargs*)

DeleteCpe Deletes the specified CPE object. The CPE must not be connected to a DRG. This is an

asynchronous operation. The CPE's *lifecycleState* will change to `TERMINATING` temporarily until the CPE is completely removed.

Parameters

- **cpe_id** (*str*) – (required) The OCID of the CPE.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type `None`

Return type *Response*

delete_cross_connect (*cross_connect_id*, ****kwargs**)

DeleteCrossConnect Deletes the specified cross-connect. It must not be mapped to a `VirtualCircuit`.

Parameters

- **cross_connect_id** (*str*) – (required) The OCID of the cross-connect.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type `None`

Return type *Response*

delete_cross_connect_group (*cross_connect_group_id*, ****kwargs**)

DeleteCrossConnectGroup Deletes the specified cross-connect group. It must not contain any cross-connects, and it cannot be mapped to a `VirtualCircuit`.

Parameters

- **cross_connect_group_id** (*str*) – (required) The OCID of the cross-connect group.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type `None`

Return type *Response*

delete_dhcp_options (*dhcp_id*, ****kwargs**)

DeleteDhcpOptions Deletes the specified set of DHCP options, but only if it's not associated with a subnet. You can't delete a VCN's default set of DHCP options.

This is an asynchronous operation. The state of the set of options will switch to `TERMINATING` temporarily until the set is completely removed.

Parameters

- **dhcp_id** (*str*) – (required) The OCID for the set of DHCP options.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type None

Return type *Response*

delete_drg (*drg_id*, ***kwargs*)

DeleteDrg Deletes the specified DRG. The DRG must not be attached to a VCN or be connected to your on-premise network. Also, there must not be a route table that lists the DRG as a target. This is an asynchronous operation. The DRG's *lifecycleState* will change to TERMINATING temporarily until the DRG is completely removed.

Parameters

- **drg_id** (*str*) – (required) The OCID of the DRG.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type None

Return type *Response*

delete_drg_attachment (*drg_attachment_id*, ***kwargs*)

DeleteDrgAttachment Detaches a DRG from a VCN by deleting the corresponding *DrgAttachment*. This is an asynchronous operation. The attachment's *lifecycleState* will change to DETACHING temporarily until the attachment is completely removed.

Parameters

- **drg_attachment_id** (*str*) – (required) The OCID of the DRG attachment.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type None

Return type *Response*

delete_internet_gateway (*ig_id*, ***kwargs*)

DeleteInternetGateway Deletes the specified Internet Gateway. The Internet Gateway does not have to be disabled, but there must not be a route table that lists it as a target.

This is an asynchronous operation. The gateway's *lifecycleState* will change to TERMINATING temporarily until the gateway is completely removed.

Parameters

- **ig_id** (*str*) – (required) The OCID of the Internet Gateway.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type None

Return type *Response*

delete_ip_sec_connection (*ipsc_id*, ***kwargs*)

DeleteIPSecConnection Deletes the specified IPSec connection. If your goal is to disable the IPSec VPN between your VCN and on-premises network, it's easiest to simply detach the DRG but keep all the IPSec

VPN components intact. If you were to delete all the components and then later need to create an IPsec VPN again, you would need to configure your on-premises router again with the new information returned from `create_ip_sec_connection()`.

This is an asynchronous operation. The connection's *lifecycleState* will change to `TERMINATING` temporarily until the connection is completely removed.

Parameters

- **ipsec_id** (*str*) – (required) The OCID of the IPsec connection.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type `None`

Return type *Response*

delete_private_ip (*private_ip_id*, ***kwargs*)

DeletePrivateIp Unassigns and deletes the specified private IP. You must specify the object's OCID. The private IP address is returned to the subnet's pool of available addresses.

This operation cannot be used with primary private IPs, which are automatically unassigned and deleted when the VNIC is terminated.

Parameters

- **private_ip_id** (*str*) – (required) The private IP's OCID.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type `None`

Return type *Response*

delete_route_table (*rt_id*, ***kwargs*)

DeleteRouteTable Deletes the specified route table, but only if it's not associated with a subnet. You can't delete a VCN's default route table.

This is an asynchronous operation. The route table's *lifecycleState* will change to `TERMINATING` temporarily until the route table is completely removed.

Parameters

- **rt_id** (*str*) – (required) The OCID of the route table.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type `None`

Return type *Response*

delete_security_list (*security_list_id*, ***kwargs*)

DeleteSecurityList Deletes the specified security list, but only if it's not associated with a subnet. You can't delete a VCN's default security list.

This is an asynchronous operation. The security list's *lifecycleState* will change to TERMINATING temporarily until the security list is completely removed.

Parameters

- **security_list_id** (*str*) – (required) The OCID of the security list.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type None

Return type *Response*

delete_subnet (*subnet_id*, ***kwargs*)

DeleteSubnet Deletes the specified subnet, but only if there are no instances in the subnet. This is an asynchronous operation. The subnet's *lifecycleState* will change to TERMINATING temporarily. If there are any instances in the subnet, the state will instead change back to AVAILABLE.

Parameters

- **subnet_id** (*str*) – (required) The OCID of the subnet.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type None

Return type *Response*

delete_vcn (*vcn_id*, ***kwargs*)

DeleteVcn Deletes the specified VCN. The VCN must be empty and have no attached gateways. This is an asynchronous operation. The VCN's *lifecycleState* will change to TERMINATING temporarily until the VCN is completely removed.

Parameters

- **vcn_id** (*str*) – (required) The OCID of the VCN.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type None

Return type *Response*

delete_virtual_circuit (*virtual_circuit_id*, ***kwargs*)

DeleteVirtualCircuit Deletes the specified virtual circuit.

Important: If you're using FastConnect via a provider, make sure to also terminate the connection with the provider, or else the provider may continue to bill you.

Parameters

- **virtual_circuit_id** (*str*) – (required) The OCID of the virtual circuit.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a

previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type *None*

Return type *Response*

get_cpe (*cpe_id*, ****kwargs**)

GetCpe Gets the specified CPE's information.

Parameters **cpe_id** (*str*) – (required) The OCID of the CPE.

Returns A *Response* object with data of type *Cpe*

Return type *Response*

get_cross_connect (*cross_connect_id*, ****kwargs**)

GetCrossConnect Gets the specified cross-connect's information.

Parameters **cross_connect_id** (*str*) – (required) The OCID of the cross-connect.

Returns A *Response* object with data of type *CrossConnect*

Return type *Response*

get_cross_connect_group (*cross_connect_group_id*, ****kwargs**)

GetCrossConnectGroups Gets the specified cross-connect group's information.

Parameters **cross_connect_group_id** (*str*) – (required) The OCID of the cross-connect group.

Returns A *Response* object with data of type *CrossConnectGroup*

Return type *Response*

get_cross_connect_letter_of_authority (*cross_connect_id*, ****kwargs**)

GetCrossConnectLetterOfAuthority Gets the Letter of Authority for the specified cross-connect.

Parameters **cross_connect_id** (*str*) – (required) The OCID of the cross-connect.

Returns A *Response* object with data of type *LetterOfAuthority*

Return type *Response*

get_cross_connect_status (*cross_connect_id*, ****kwargs**)

GetCrossConnectStatus Gets the status of the specified cross-connect.

Parameters **cross_connect_id** (*str*) – (required) The OCID of the cross-connect.

Returns A *Response* object with data of type *CrossConnectStatus*

Return type *Response*

get_dhcp_options (*dhcp_id*, ****kwargs**)

GetDhcpOptions Gets the specified set of DHCP options.

Parameters **dhcp_id** (*str*) – (required) The OCID for the set of DHCP options.

Returns A *Response* object with data of type *DhcpOptions*

Return type *Response*

get_drg (*drg_id*, ****kwargs**)

GetDrg Gets the specified DRG's information.

Parameters **drg_id** (*str*) – (required) The OCID of the DRG.

Returns A *Response* object with data of type *Drg*

Return type *Response*

get_drg_attachment (*drg_attachment_id*, ***kwargs*)

GetDrgAttachment Gets the information for the specified *DrgAttachment*.

Parameters **drg_attachment_id** (*str*) – (required) The OCID of the DRG attachment.

Returns A *Response* object with data of type *DrgAttachment*

Return type *Response*

get_internet_gateway (*ig_id*, ***kwargs*)

GetInternetGateway Gets the specified Internet Gateway’s information.

Parameters **ig_id** (*str*) – (required) The OCID of the Internet Gateway.

Returns A *Response* object with data of type *InternetGateway*

Return type *Response*

get_ip_sec_connection (*ipsc_id*, ***kwargs*)

GetIPSecConnection Gets the specified IPSec connection’s basic information, including the static routes for the on-premises router. If you want the status of the connection (whether it’s up or down), use `get_ip_sec_connection_device_status()`.

Parameters **ipsc_id** (*str*) – (required) The OCID of the IPSec connection.

Returns A *Response* object with data of type *IPSecConnection*

Return type *Response*

get_ip_sec_connection_device_config (*ipsc_id*, ***kwargs*)

GetIPSecConnectionDeviceConfig Gets the configuration information for the specified IPSec connection. For each tunnel, the response includes the IP address of Oracle’s VPN headend and the shared secret.

Parameters **ipsc_id** (*str*) – (required) The OCID of the IPSec connection.

Returns A *Response* object with data of type *IPSecConnectionDeviceConfig*

Return type *Response*

get_ip_sec_connection_device_status (*ipsc_id*, ***kwargs*)

GetIPSecConnectionDeviceStatus Gets the status of the specified IPSec connection (whether it’s up or down).

Parameters **ipsc_id** (*str*) – (required) The OCID of the IPSec connection.

Returns A *Response* object with data of type *IPSecConnectionDeviceStatus*

Return type *Response*

get_private_ip (*private_ip_id*, ***kwargs*)

GetPrivateIp Gets the specified private IP. You must specify the object’s OCID. Alternatively, you can get the object by using `list_private_ips()` with the private IP address (for example, 10.0.3.3) and subnet OCID.

Parameters **private_ip_id** (*str*) – (required) The private IP’s OCID.

Returns A *Response* object with data of type *PrivateIp*

Return type *Response*

get_route_table (*rt_id*, ***kwargs*)

GetRouteTable Gets the specified route table’s information.

Parameters **rt_id** (*str*) – (required) The OCID of the route table.

Returns A *Response* object with data of type *RouteTable*

Return type *Response*

get_security_list (*security_list_id*, ***kwargs*)

GetSecurityList Gets the specified security list's information.

Parameters **security_list_id** (*str*) – (required) The OCID of the security list.

Returns A *Response* object with data of type *SecurityList*

Return type *Response*

get_subnet (*subnet_id*, ***kwargs*)

GetSubnet Gets the specified subnet's information.

Parameters **subnet_id** (*str*) – (required) The OCID of the subnet.

Returns A *Response* object with data of type *Subnet*

Return type *Response*

get_vcn (*vcn_id*, ***kwargs*)

GetVcn Gets the specified VCN's information.

Parameters **vcn_id** (*str*) – (required) The OCID of the VCN.

Returns A *Response* object with data of type *Vcn*

Return type *Response*

get_virtual_circuit (*virtual_circuit_id*, ***kwargs*)

GetVirtualCircuit Gets the specified virtual circuit's information.

Parameters **virtual_circuit_id** (*str*) – (required) The OCID of the virtual circuit.

Returns A *Response* object with data of type *VirtualCircuit*

Return type *Response*

get_vnic (*vnic_id*, ***kwargs*)

GetVnic Gets the information for the specified virtual network interface card (VNIC). You can get the VNIC OCID from the `list_vnic_attachments()` operation.

Parameters **vnic_id** (*str*) – (required) The OCID of the VNIC.

Returns A *Response* object with data of type *Vnic*

Return type *Response*

list_cpes (*compartment_id*, ***kwargs*)

ListCpes Lists the Customer-Premises Equipment objects (CPEs) in the specified compartment.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: `500`

- **page** (*str*) – (optional) The value of the `opc-next-page` response header from the previous “List” call.

Returns A *Response* object with data of type list of *Cpe*

Return type *Response*

list_cross_connect_groups (*compartment_id*, ***kwargs*)

ListCrossConnectGroups Lists the cross-connect groups in the specified compartment.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: 500

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Returns A *Response* object with data of type list of *CrossConnectGroup*

Return type *Response*

list_cross_connect_locations (*compartment_id*, ***kwargs*)

ListCrossConnectLocations Lists the available FastConnect locations for cross-connect installation. You need this information so you can specify your desired location when you create a cross-connect.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: 500

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Returns A *Response* object with data of type list of *CrossConnectLocation*

Return type *Response*

list_cross_connects (*compartment_id*, ***kwargs*)

ListCrossConnects Lists the cross-connects in the specified compartment. You can filter the list by specifying the OCID of a cross-connect group.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **cross_connect_group_id** (*str*) – (optional) The OCID of the cross-connect group.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: 500

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Returns A *Response* object with data of type list of *CrossConnect*

Return type *Response*

list_crossconnect_port_speed_shapes (*compartment_id*, ***kwargs*)

ListCrossConnectPortSpeedShapes Lists the available port speeds for cross-connects. You need this information so you can specify your desired port speed (that is, shape) when you create a cross-connect.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: 500

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Returns A *Response* object with data of type list of *CrossConnectPortSpeedShape*

Return type *Response*

list_dhcp_options (*compartment_id*, *vcn_id*, ***kwargs*)

ListDhcpOptions Lists the sets of DHCP options in the specified VCN and specified compartment. The response includes the default set of options that automatically comes with each VCN, plus any other sets you’ve created.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **vcn_id** (*str*) – (required) The OCID of the VCN.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: 500

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Returns A *Response* object with data of type list of *DhcpOptions*

Return type *Response*

list_drg_attachments (*compartment_id*, ***kwargs*)

ListDrgAttachments Lists the *DrgAttachment* objects for the specified compartment. You can filter the results by VCN or DRG.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **vcn_id** (*str*) – (optional) The OCID of the VCN.
- **drg_id** (*str*) – (optional) The OCID of the DRG.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: 500

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Returns A *Response* object with data of type list of *DrgAttachment*

Return type *Response*

list_drgs (*compartment_id*, ***kwargs*)

ListDrgs Lists the DRGs in the specified compartment.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.

- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: *500*

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Returns A *Response* object with data of type list of *Drg*

Return type *Response*

list_fast_connect_provider_services (*compartment_id*, ***kwargs*)

ListFastConnectProviderServices Lists the service offerings from supported providers. You need this information so you can specify your desired provider and service offering when you create a virtual circuit.

For the compartment ID, provide the OCID of your tenancy (the root compartment).

For more information, see [FastConnect Overview](#).

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: *500*

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Returns A *Response* object with data of type list of *FastConnectProviderService*

Return type *Response*

list_internet_gateways (*compartment_id*, *vcn_id*, ***kwargs*)

ListInternetGateways Lists the Internet Gateways in the specified VCN and the specified compartment.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **vcn_id** (*str*) – (required) The OCID of the VCN.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: *500*

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Returns A *Response* object with data of type list of *InternetGateway*

Return type *Response*

list_ip_sec_connections (*compartment_id*, ***kwargs*)

ListIPSecConnections Lists the IPSec connections for the specified compartment. You can filter the results by DRG or CPE.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **drg_id** (*str*) – (optional) The OCID of the DRG.
- **cpe_id** (*str*) – (optional) The OCID of the CPE.

- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: 500

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Returns A *Response* object with data of type list of *IPSecConnection*

Return type *Response*

list_private_ips (***kwargs*)

ListPrivateIps Lists the PrivateIp objects based on one of these filters:

- Subnet OCID.
- VNIC OCID.
- Both private IP address and subnet OCID: This lets

you get a *privateIP* object based on its private IP address (for example, 10.0.3.3) and not its OCID. For comparison, `get_private_ip()` requires the OCID.

If you’re listing all the private IPs associated with a given subnet or VNIC, the response includes both primary and secondary private IPs.

Parameters

- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: 500

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

- **ip_address** (*str*) – (optional) The private IP address of the *privateIp* object.

Example: 10.0.3.3

- **subnet_id** (*str*) – (optional) The OCID of the subnet.

- **vnic_id** (*str*) – (optional) The OCID of the VNIC.

Returns A *Response* object with data of type list of *PrivateIp*

Return type *Response*

list_route_tables (*compartment_id*, *vcn_id*, ***kwargs*)

ListRouteTables Lists the route tables in the specified VCN and specified compartment. The response includes the default route table that automatically comes with each VCN, plus any route tables you’ve created.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **vcn_id** (*str*) – (required) The OCID of the VCN.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: 500

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Returns A *Response* object with data of type list of *RouteTable*

Return type *Response*

list_security_lists (*compartment_id*, *vcn_id*, ***kwargs*)

ListSecurityLists Lists the security lists in the specified VCN and compartment.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **vcn_id** (*str*) – (required) The OCID of the VCN.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: 500

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Returns A *Response* object with data of type list of *SecurityList*

Return type *Response*

list_subnets (*compartment_id*, *vcn_id*, ***kwargs*)

ListSubnets Lists the subnets in the specified VCN and the specified compartment.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **vcn_id** (*str*) – (required) The OCID of the VCN.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: 500

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Returns A *Response* object with data of type list of *Subnet*

Return type *Response*

list_vcns (*compartment_id*, ***kwargs*)

ListVcns Lists the Virtual Cloud Networks (VCNs) in the specified compartment.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: 500

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Returns A *Response* object with data of type list of *Vcn*

Return type *Response*

list_virtual_circuit_bandwidth_shapes (*compartment_id*, ***kwargs*)

ListVirtualCircuitBandwidthShapes Lists the available bandwidth levels for virtual circuits. You need this

information so you can specify your desired bandwidth level (that is, shape) when you create a virtual circuit.

For the compartment ID, provide the OCID of your tenancy (the root compartment).

For more information about virtual circuits, see [FastConnect Overview](#).

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: 500

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Returns A *Response* object with data of type list of *VirtualCircuitBandwidthShape*

Return type *Response*

list_virtual_circuits (*compartment_id*, ***kwargs*)

ListVirtualCircuits Lists the virtual circuits in the specified compartment.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: 500

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Returns A *Response* object with data of type list of *VirtualCircuit*

Return type *Response*

update_cpe (*cpe_id*, *update_cpe_details*, ***kwargs*)

UpdateCpe Updates the specified CPE’s display name. Avoid entering confidential information.

Parameters

- **cpe_id** (*str*) – (required) The OCID of the CPE.
- **update_cpe_details** (*UpdateCpeDetails*) – (required) Details object for updating a CPE.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *Cpe*

Return type *Response*

update_cross_connect (*cross_connect_id*, *update_cross_connect_details*, ***kwargs*)

UpdateCrossConnect Updates the specified cross-connect.

Parameters

- **cross_connect_id** (*str*) – (required) The OCID of the cross-connect.

- **update_cross_connect_details** (`UpdateCrossConnectDetails`) – (required) Update `CrossConnect` fields.
- **if_match** (`str`) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *CrossConnect*

Return type *Response*

update_cross_connect_group (`cross_connect_group_id`, `update_cross_connect_group_details`, `**kwargs`)

`UpdateCrossConnectGroup` Updates the specified cross-connect group’s display name. Avoid entering confidential information.

Parameters

- **cross_connect_group_id** (`str`) – (required) The OCID of the cross-connect group.
- **update_cross_connect_group_details** (`UpdateCrossConnectGroupDetails`) – (required) Update `CrossConnectGroup` fields
- **if_match** (`str`) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *CrossConnectGroup*

Return type *Response*

update_dhcp_options (`dhcp_id`, `update_dhcp_details`, `**kwargs`)

`UpdateDhcpOptions` Updates the specified set of DHCP options. You can update the display name or the options themselves. Avoid entering confidential information.

Note that the *options* object you provide replaces the entire existing set of options.

Parameters

- **dhcp_id** (`str`) – (required) The OCID for the set of DHCP options.
- **update_dhcp_details** (`UpdateDhcpDetails`) – (required) Request object for updating a set of DHCP options.
- **if_match** (`str`) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *DhcpOptions*

Return type *Response*

update_drg (`drg_id`, `update_drg_details`, `**kwargs`)

`UpdateDrg` Updates the specified DRG’s display name. Avoid entering confidential information.

Parameters

- **drg_id** (`str`) – (required) The OCID of the DRG.
- **update_drg_details** (`UpdateDrgDetails`) – (required) Details object for updating a DRG.

- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *Drg*

Return type *Response*

update_drg_attachment (*drg_attachment_id*, *update_drg_attachment_details*, ****kwargs**)

UpdateDrgAttachment Updates the display name for the specified *DrgAttachment*. Avoid entering confidential information.

Parameters

- **drg_attachment_id** (*str*) – (required) The OCID of the DRG attachment.
- **update_drg_attachment_details** (*UpdateDrgAttachmentDetails*) – (required) Details object for updating a *DrgAttachment*.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *DrgAttachment*

Return type *Response*

update_internet_gateway (*ig_id*, *update_internet_gateway_details*, ****kwargs**)

UpdateInternetGateway Updates the specified Internet Gateway. You can disable/enable it, or change its display name. Avoid entering confidential information.

If the gateway is disabled, that means no traffic will flow to/from the internet even if there’s a route rule that enables that traffic.

Parameters

- **ig_id** (*str*) – (required) The OCID of the Internet Gateway.
- **update_internet_gateway_details** (*UpdateInternetGatewayDetails*) – (required) Details for updating the Internet Gateway.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *InternetGateway*

Return type *Response*

update_ip_sec_connection (*ipsec_id*, *update_ip_sec_connection_details*, ****kwargs**)

UpdateIPSecConnection Updates the display name for the specified IPSec connection. Avoid entering confidential information.

Parameters

- **ipsec_id** (*str*) – (required) The OCID of the IPSec connection.
- **update_ip_sec_connection_details** (*UpdateIPSecConnectionDetails*) – (required) Details object for updating a IPSec connection.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a

previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type *IPSecConnection*

Return type *Response*

update_private_ip (*private_ip_id*, *update_private_ip_details*, ***kwargs*)

UpdatePrivateIp Updates the specified private IP. You must specify the object's OCID. Use this operation if you want to:

- Move a secondary private IP to a different VNIC in the same subnet.
- Change the display name for a secondary private IP.
- Change the hostname for a secondary private IP.

This operation cannot be used with primary private IPs. To update the hostname for the primary IP on a VNIC, use `update_vnic()`.

Parameters

- **private_ip_id** (*str*) – (required) The private IP's OCID.
- **update_private_ip_details** (*UpdatePrivateIpDetails*) – (required) Private IP details.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type *PrivateIp*

Return type *Response*

update_route_table (*rt_id*, *update_route_table_details*, ***kwargs*)

UpdateRouteTable Updates the specified route table's display name or route rules. Avoid entering confidential information.

Note that the *routeRules* object you provide replaces the entire existing set of rules.

Parameters

- **rt_id** (*str*) – (required) The OCID of the route table.
- **update_route_table_details** (*UpdateRouteTableDetails*) – (required) Details object for updating a route table.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type *RouteTable*

Return type *Response*

update_security_list (*security_list_id*, *update_security_list_details*, ***kwargs*)

UpdateSecurityList Updates the specified security list's display name or rules. Avoid entering confidential information.

Note that the *egressSecurityRules* or *ingressSecurityRules* objects you provide replace the entire existing objects.

Parameters

- **security_list_id** (*str*) – (required) The OCID of the security list.
- **update_security_list_details** (`UpdateSecurityListDetails`) – (required) Updated details for the security list.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *SecurityList*

Return type *Response*

update_subnet (*subnet_id*, *update_subnet_details*, ***kwargs*)

UpdateSubnet Updates the specified subnet’s display name. Avoid entering confidential information.

Parameters

- **subnet_id** (*str*) – (required) The OCID of the subnet.
- **update_subnet_details** (`UpdateSubnetDetails`) – (required) Details object for updating a subnet.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *Subnet*

Return type *Response*

update_vcn (*vcn_id*, *update_vcn_details*, ***kwargs*)

UpdateVcn Updates the specified VCN’s display name. Avoid entering confidential information.

Parameters

- **vcn_id** (*str*) – (required) The OCID of the VCN.
- **update_vcn_details** (`UpdateVcnDetails`) – (required) Details object for updating a VCN.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *Vcn*

Return type *Response*

update_virtual_circuit (*virtual_circuit_id*, *update_virtual_circuit_details*, ***kwargs*)

UpdateVirtualCircuit Updates the specified virtual circuit. This can be called by either the customer who owns the virtual circuit, or the provider (when provisioning or de-provisioning the virtual circuit from their end). The documentation for `update_virtual_circuit_details()` indicates who can update each property of the virtual circuit.

Important: If the virtual circuit is working and in the PROVISIONED state, updating any of the network-related properties (such as the DRG being used, the BGP ASN, and so on) will cause the virtual circuit’s state to switch to PROVISIONING and the related BGP session to go down. After Oracle re-provisions the virtual circuit, its state will return to PROVISIONED. Make sure you confirm that the associated BGP session is back up. For more information about the various states and how to test connectivity, see [FastConnect Overview](#).

Parameters

- **virtual_circuit_id** (*str*) – (required) The OCID of the virtual circuit.
- **update_virtual_circuit_details** (*UpdateVirtualCircuitDetails*) – (required) Update VirtualCircuit fields.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *VirtualCircuit*

Return type *Response*

update_vnic (*vnic_id*, *update_vnic_details*, ***kwargs*)

UpdateVnic Updates the specified VNIC.

Parameters

- **vnic_id** (*str*) – (required) The OCID of the VNIC.
- **update_vnic_details** (*UpdateVnicDetails*) – (required) Details object for updating a VNIC.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *Vnic*

Return type *Response*

Models

class `oci.core.models.AttachIScsiVolumeDetails`

use_chap

Gets the use_chap of this AttachIScsiVolumeDetails. Whether to use CHAP authentication for the volume attachment. Defaults to false.

Returns The use_chap of this AttachIScsiVolumeDetails.

Return type bool

class `oci.core.models.AttachVnicDetails`

create_vnic_details

Gets the create_vnic_details of this AttachVnicDetails. Details for creating a new VNIC.

Returns The create_vnic_details of this AttachVnicDetails.

Return type *CreateVnicDetails*

display_name

Gets the display_name of this AttachVnicDetails. A user-friendly name for the attachment. Does not have to be unique, and it cannot be changed.

Returns The display_name of this AttachVnicDetails.

Return type str

instance_id

Gets the instance_id of this AttachVnicDetails. The OCID of the instance.

Returns The instance_id of this AttachVnicDetails.

Return type str

class `oci.core.models.AttachVolumeDetails`

display_name

Gets the display_name of this AttachVolumeDetails. A user-friendly name. Does not have to be unique, and it cannot be changed. Avoid entering confidential information.

Returns The display_name of this AttachVolumeDetails.

Return type str

static get_subtype (*object_dictionary*)

Given the hash representation of a subtype of this class, use the info in the hash to return the class of the subtype.

instance_id

Gets the instance_id of this AttachVolumeDetails. The OCID of the instance.

Returns The instance_id of this AttachVolumeDetails.

Return type str

type

Gets the type of this AttachVolumeDetails. The type of volume. The only supported value is "iscsi".

Returns The type of this AttachVolumeDetails.

Return type str

volume_id

Gets the volume_id of this AttachVolumeDetails. The OCID of the volume.

Returns The volume_id of this AttachVolumeDetails.

Return type str

class `oci.core.models.CaptureConsoleHistoryDetails`

instance_id

Gets the instance_id of this CaptureConsoleHistoryDetails. The OCID of the instance to get the console history from.

Returns The instance_id of this CaptureConsoleHistoryDetails.

Return type str

class `oci.core.models.ConsoleHistory`

availability_domain

Gets the availability_domain of this ConsoleHistory. The Availability Domain of an instance.

Example: *Uocm:PHX-AD-1*

Returns The availability_domain of this ConsoleHistory.

Return type str

compartment_id

Gets the compartment_id of this ConsoleHistory. The OCID of the compartment.

Returns The compartment_id of this ConsoleHistory.

Return type str

display_name

Gets the display_name of this ConsoleHistory. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Example: *My console history metadata*

Returns The display_name of this ConsoleHistory.

Return type str

id

Gets the id of this ConsoleHistory. The OCID of the console history metadata object.

Returns The id of this ConsoleHistory.

Return type str

instance_id

Gets the instance_id of this ConsoleHistory. The OCID of the instance this console history was fetched from.

Returns The instance_id of this ConsoleHistory.

Return type str

lifecycle_state

Gets the lifecycle_state of this ConsoleHistory. The current state of the console history.

Allowed values for this property are: "REQUESTED", "GETTING-HISTORY", "SUCCEEDED", "FAILED", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The lifecycle_state of this ConsoleHistory.

Return type str

time_created

Gets the time_created of this ConsoleHistory. The date and time the history was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The time_created of this ConsoleHistory.

Return type datetime

class `oci.core.models.Cpe`

compartment_id

Gets the compartment_id of this Cpe. The OCID of the compartment containing the CPE.

Returns The compartment_id of this Cpe.

Return type str

display_name

Gets the display_name of this Cpe. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The display_name of this Cpe.

Return type str

id

Gets the id of this Cpe. The CPE's Oracle ID (OCID).

Returns The id of this Cpe.

Return type str

ip_address

Gets the ip_address of this Cpe. The public IP address of the on-premises router.

Returns The ip_address of this Cpe.

Return type str

time_created

Gets the time_created of this Cpe. The date and time the CPE was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The time_created of this Cpe.

Return type datetime

class oci.core.models.**CreateCpeDetails**

compartment_id

Gets the compartment_id of this CreateCpeDetails. The OCID of the compartment to contain the CPE.

Returns The compartment_id of this CreateCpeDetails.

Return type str

display_name

Gets the display_name of this CreateCpeDetails. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The display_name of this CreateCpeDetails.

Return type str

ip_address

Gets the ip_address of this CreateCpeDetails. The public IP address of the on-premises router.

Example: *143.19.23.16*

Returns The ip_address of this CreateCpeDetails.

Return type str

class oci.core.models.**CreateCrossConnectDetails**

compartment_id

Gets the compartment_id of this CreateCrossConnectDetails. The OCID of the compartment to contain the cross-connect.

Returns The compartment_id of this CreateCrossConnectDetails.

Return type str

cross_connect_group_id

Gets the `cross_connect_group_id` of this `CreateCrossConnectDetails`. The OCID of the cross-connect group to put this cross-connect in.

Returns The `cross_connect_group_id` of this `CreateCrossConnectDetails`.

Return type str

display_name

Gets the `display_name` of this `CreateCrossConnectDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `CreateCrossConnectDetails`.

Return type str

far_cross_connect_or_cross_connect_group_id

Gets the `far_cross_connect_or_cross_connect_group_id` of this `CreateCrossConnectDetails`. If you already have an existing cross-connect or cross-connect group at this `FastConnect` location, and you want this new cross-connect to be on a different router (for the purposes of redundancy), provide the OCID of that existing cross-connect or cross-connect group.

Returns The `far_cross_connect_or_cross_connect_group_id` of this `CreateCrossConnectDetails`.

Return type str

location_name

Gets the `location_name` of this `CreateCrossConnectDetails`. The name of the `FastConnect` location where this cross-connect will be installed. To get a list of the available locations, see `list_cross_connect_locations()`.

Example: *CyrusOne, Chandler, AZ*

Returns The `location_name` of this `CreateCrossConnectDetails`.

Return type str

near_cross_connect_or_cross_connect_group_id

Gets the `near_cross_connect_or_cross_connect_group_id` of this `CreateCrossConnectDetails`. If you already have an existing cross-connect or cross-connect group at this `FastConnect` location, and you want this new cross-connect to be on the same router, provide the OCID of that existing cross-connect or cross-connect group.

Returns The `near_cross_connect_or_cross_connect_group_id` of this `CreateCrossConnectDetails`.

Return type str

port_speed_shape_name

Gets the `port_speed_shape_name` of this `CreateCrossConnectDetails`. The port speed for this cross-connect. To get a list of the available port speeds, see `list_crossconnect_port_speed_shapes()`.

Example: *10 Gbps*

Returns The `port_speed_shape_name` of this `CreateCrossConnectDetails`.

Return type str

class `oci.core.models.CreateCrossConnectGroupDetails`

compartment_id

Gets the compartment_id of this CreateCrossConnectGroupDetails. The OCID of the compartment to contain the cross-connect group.

Returns The compartment_id of this CreateCrossConnectGroupDetails.

Return type str

display_name

Gets the display_name of this CreateCrossConnectGroupDetails. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The display_name of this CreateCrossConnectGroupDetails.

Return type str

class oci.core.models.**CreateDhcpDetails**

compartment_id

Gets the compartment_id of this CreateDhcpDetails. The OCID of the compartment to contain the set of DHCP options.

Returns The compartment_id of this CreateDhcpDetails.

Return type str

display_name

Gets the display_name of this CreateDhcpDetails. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The display_name of this CreateDhcpDetails.

Return type str

options

Gets the options of this CreateDhcpDetails. A set of DHCP options.

Returns The options of this CreateDhcpDetails.

Return type list[DhcpOption]

vcn_id

Gets the vcn_id of this CreateDhcpDetails. The OCID of the VCN the set of DHCP options belongs to.

Returns The vcn_id of this CreateDhcpDetails.

Return type str

class oci.core.models.**CreateDrgAttachmentDetails**

display_name

Gets the display_name of this CreateDrgAttachmentDetails. A user-friendly name. Does not have to be unique. Avoid entering confidential information.

Returns The display_name of this CreateDrgAttachmentDetails.

Return type str

drg_id

Gets the drg_id of this CreateDrgAttachmentDetails. The OCID of the DRG.

Returns The drg_id of this CreateDrgAttachmentDetails.

Return type str

vcn_id

Gets the vcn_id of this CreateDrgAttachmentDetails. The OCID of the VCN.

Returns The vcn_id of this CreateDrgAttachmentDetails.

Return type str

class `oci.core.models.CreateDrgDetails`

compartment_id

Gets the compartment_id of this CreateDrgDetails. The OCID of the compartment to contain the DRG.

Returns The compartment_id of this CreateDrgDetails.

Return type str

display_name

Gets the display_name of this CreateDrgDetails. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The display_name of this CreateDrgDetails.

Return type str

class `oci.core.models.CreateIPSecConnectionDetails`

compartment_id

Gets the compartment_id of this CreateIPSecConnectionDetails. The OCID of the compartment to contain the IPSec connection.

Returns The compartment_id of this CreateIPSecConnectionDetails.

Return type str

cpe_id

Gets the cpe_id of this CreateIPSecConnectionDetails. The OCID of the CPE.

Returns The cpe_id of this CreateIPSecConnectionDetails.

Return type str

display_name

Gets the display_name of this CreateIPSecConnectionDetails. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The display_name of this CreateIPSecConnectionDetails.

Return type str

drg_id

Gets the drg_id of this CreateIPSecConnectionDetails. The OCID of the DRG.

Returns The drg_id of this CreateIPSecConnectionDetails.

Return type str

static_routes

Gets the static_routes of this CreateIPSecConnectionDetails. Static routes to the CPE. At least one route must be included. The CIDR must not be a multicast address or class E address.

Example: *10.0.1.0/24*

Returns The static_routes of this CreateIPSecConnectionDetails.

Return type list[str]

class `oci.core.models.CreateImageDetails`

compartment_id

Gets the `compartment_id` of this `CreateImageDetails`. The OCID of the compartment containing the instance you want to use as the basis for the image.

Returns The `compartment_id` of this `CreateImageDetails`.

Return type `str`

display_name

Gets the `display_name` of this `CreateImageDetails`. A user-friendly name for the image. It does not have to be unique, and it's changeable. Avoid entering confidential information.

You cannot use an Oracle-provided image name as a custom image name.

Example: *My Oracle Linux image*

Returns The `display_name` of this `CreateImageDetails`.

Return type `str`

image_source_details

Gets the `image_source_details` of this `CreateImageDetails`. Details for creating an image through import

Returns The `image_source_details` of this `CreateImageDetails`.

Return type *ImageSourceDetails*

instance_id

Gets the `instance_id` of this `CreateImageDetails`. The OCID of the instance you want to use as the basis for the image.

Returns The `instance_id` of this `CreateImageDetails`.

Return type `str`

class `oci.core.models.CreateInstanceConsoleConnectionDetails`

instance_id

Gets the `instance_id` of this `CreateInstanceConsoleConnectionDetails`. The host instance OCID

Returns The `instance_id` of this `CreateInstanceConsoleConnectionDetails`.

Return type `str`

public_key

Gets the `public_key` of this `CreateInstanceConsoleConnectionDetails`. An ssh public key that will be used to authenticate the console connection.

Returns The `public_key` of this `CreateInstanceConsoleConnectionDetails`.

Return type `str`

class `oci.core.models.CreateInternetGatewayDetails`

compartment_id

Gets the `compartment_id` of this `CreateInternetGatewayDetails`. The OCID of the compartment to contain the Internet Gateway.

Returns The `compartment_id` of this `CreateInternetGatewayDetails`.

Return type `str`

display_name

Gets the `display_name` of this `CreateInternetGatewayDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `CreateInternetGatewayDetails`.

Return type str

is_enabled

Gets the `is_enabled` of this `CreateInternetGatewayDetails`. Whether the gateway is enabled upon creation.

Returns The `is_enabled` of this `CreateInternetGatewayDetails`.

Return type bool

vcn_id

Gets the `vcn_id` of this `CreateInternetGatewayDetails`. The OCID of the VCN the Internet Gateway is attached to.

Returns The `vcn_id` of this `CreateInternetGatewayDetails`.

Return type str

class `oci.core.models.CreatePrivateIpDetails`

display_name

Gets the `display_name` of this `CreatePrivateIpDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `CreatePrivateIpDetails`.

Return type str

hostname_label

Gets the `hostname_label` of this `CreatePrivateIpDetails`. The hostname for the private IP. Used for DNS. The value is the hostname portion of the private IP's fully qualified domain name (FQDN) (for example, *binstance-1* in FQDN *binstance-1.subnet123.vcn1.oraclevcn.com*). Must be unique across all VNICs in the subnet and comply with [RFC 952](#) and [RFC 1123](#).

For more information, see [DNS in Your Virtual Cloud Network](#).

Example: *binstance-1*

Returns The `hostname_label` of this `CreatePrivateIpDetails`.

Return type str

ip_address

Gets the `ip_address` of this `CreatePrivateIpDetails`. A private IP address of your choice. Must be an available IP address within the subnet's CIDR. If you don't specify a value, Oracle automatically assigns a private IP address from the subnet.

Example: *10.0.3.3*

Returns The `ip_address` of this `CreatePrivateIpDetails`.

Return type str

vnic_id

Gets the `vnic_id` of this `CreatePrivateIpDetails`. The OCID of the VNIC to assign the private IP to. The VNIC and private IP must be in the same subnet.

Returns The `vnic_id` of this `CreatePrivateIpDetails`.

Return type str

class `oci.core.models.CreateRouteTableDetails`

compartment_id

Gets the `compartment_id` of this `CreateRouteTableDetails`. The OCID of the compartment to contain the route table.

Returns The `compartment_id` of this `CreateRouteTableDetails`.

Return type `str`

display_name

Gets the `display_name` of this `CreateRouteTableDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `CreateRouteTableDetails`.

Return type `str`

route_rules

Gets the `route_rules` of this `CreateRouteTableDetails`. The collection of rules used for routing destination IPs to network devices.

Returns The `route_rules` of this `CreateRouteTableDetails`.

Return type `list[RouteRule]`

vcn_id

Gets the `vcn_id` of this `CreateRouteTableDetails`. The OCID of the VCN the route table belongs to.

Returns The `vcn_id` of this `CreateRouteTableDetails`.

Return type `str`

class `oci.core.models.CreateSecurityListDetails`

compartment_id

Gets the `compartment_id` of this `CreateSecurityListDetails`. The OCID of the compartment to contain the security list.

Returns The `compartment_id` of this `CreateSecurityListDetails`.

Return type `str`

display_name

Gets the `display_name` of this `CreateSecurityListDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `CreateSecurityListDetails`.

Return type `str`

egress_security_rules

Gets the `egress_security_rules` of this `CreateSecurityListDetails`. Rules for allowing egress IP packets.

Returns The `egress_security_rules` of this `CreateSecurityListDetails`.

Return type `list[EgressSecurityRule]`

ingress_security_rules

Gets the `ingress_security_rules` of this `CreateSecurityListDetails`. Rules for allowing ingress IP packets.

Returns The `ingress_security_rules` of this `CreateSecurityListDetails`.

Return type `list[IngressSecurityRule]`

vcn_id

Gets the vcn_id of this CreateSecurityListDetails. The OCID of the VCN the security list belongs to.

Returns The vcn_id of this CreateSecurityListDetails.

Return type str

class oci.core.models.CreateSubnetDetails**availability_domain**

Gets the availability_domain of this CreateSubnetDetails. The Availability Domain to contain the subnet.

Example: *Uocm:PHX-AD-1*

Returns The availability_domain of this CreateSubnetDetails.

Return type str

cidr_block

Gets the cidr_block of this CreateSubnetDetails. The CIDR IP address range of the subnet.

Example: *172.16.1.0/24*

Returns The cidr_block of this CreateSubnetDetails.

Return type str

compartment_id

Gets the compartment_id of this CreateSubnetDetails. The OCID of the compartment to contain the subnet.

Returns The compartment_id of this CreateSubnetDetails.

Return type str

dhcp_options_id

Gets the dhcp_options_id of this CreateSubnetDetails. The OCID of the set of DHCP options the subnet will use. If you don't provide a value, the subnet will use the VCN's default set of DHCP options.

Returns The dhcp_options_id of this CreateSubnetDetails.

Return type str

display_name

Gets the display_name of this CreateSubnetDetails. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The display_name of this CreateSubnetDetails.

Return type str

dns_label

Gets the dns_label of this CreateSubnetDetails. A DNS label for the subnet, used in conjunction with the VNIC's hostname and VCN's DNS label to form a fully qualified domain name (FQDN) for each VNIC within this subnet (for example, *bminstance-1.subnet123.vcn1.oraclevcn.com*). Must be an alphanumeric string that begins with a letter and is unique within the VCN. The value cannot be changed.

This value must be set if you want to use the Internet and VCN Resolver to resolve the hostnames of instances in the subnet. It can only be set if the VCN itself was created with a DNS label.

For more information, see [DNS in Your Virtual Cloud Network](#).

Example: *subnet123*

Returns The dns_label of this CreateSubnetDetails.

Return type str

prohibit_public_ip_on_vnic

Gets the `prohibit_public_ip_on_vnic` of this `CreateSubnetDetails`. Whether VNICs within this subnet can have public IP addresses. Defaults to `false`, which means VNICs created in this subnet will automatically be assigned public IP addresses unless specified otherwise during instance launch or VNIC creation (with the `assignPublicIp` flag in `CreateVnicDetails`). If `prohibitPublicIpOnVnic` is set to `true`, VNICs created in this subnet cannot have public IP addresses (that is, it's a private subnet).

Example: `true`

Returns The `prohibit_public_ip_on_vnic` of this `CreateSubnetDetails`.

Return type `bool`

route_table_id

Gets the `route_table_id` of this `CreateSubnetDetails`. The OCID of the route table the subnet will use. If you don't provide a value, the subnet will use the VCN's default route table.

Returns The `route_table_id` of this `CreateSubnetDetails`.

Return type `str`

security_list_ids

Gets the `security_list_ids` of this `CreateSubnetDetails`. OCIDs for the security lists to associate with the subnet. If you don't provide a value, the VCN's default security list will be associated with the subnet. Remember that security lists are associated at the subnet level, but the rules are applied to the individual VNICs in the subnet.

Returns The `security_list_ids` of this `CreateSubnetDetails`.

Return type `list[str]`

vcn_id

Gets the `vcn_id` of this `CreateSubnetDetails`. The OCID of the VCN to contain the subnet.

Returns The `vcn_id` of this `CreateSubnetDetails`.

Return type `str`

class `oci.core.models.CreateVcnDetails`

cidr_block

Gets the `cidr_block` of this `CreateVcnDetails`. The CIDR IP address block of the VCN.

Example: `172.16.0.0/16`

Returns The `cidr_block` of this `CreateVcnDetails`.

Return type `str`

compartment_id

Gets the `compartment_id` of this `CreateVcnDetails`. The OCID of the compartment to contain the VCN.

Returns The `compartment_id` of this `CreateVcnDetails`.

Return type `str`

display_name

Gets the `display_name` of this `CreateVcnDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `CreateVcnDetails`.

Return type `str`

dns_label

Gets the `dns_label` of this `CreateVcnDetails`. A DNS label for the VCN, used in conjunction with the VNIC's hostname and subnet's DNS label to form a fully qualified domain name (FQDN) for each VNIC within this subnet (for example, *binstance-1.subnet123.vcn1.oraclevcn.com*). Not required to be unique, but it's a best practice to set unique DNS labels for VCNs in your tenancy. Must be an alphanumeric string that begins with a letter. The value cannot be changed.

You must set this value if you want instances to be able to use hostnames to resolve other instances in the VCN. Otherwise the Internet and VCN Resolver will not work.

For more information, see [DNS in Your Virtual Cloud Network](#).

Example: *vcn1*

Returns The `dns_label` of this `CreateVcnDetails`.

Return type str

class `oci.core.models.CreateVirtualCircuitDetails`

bandwidth_shape_name

Gets the `bandwidth_shape_name` of this `CreateVirtualCircuitDetails`. The provisioned data rate of the connection. To get a list of the available bandwidth levels (that is, shapes), see `list_virtual_circuit_bandwidth_shapes()`.

Example: *10 Gbps*

Returns The `bandwidth_shape_name` of this `CreateVirtualCircuitDetails`.

Return type str

compartment_id

Gets the `compartment_id` of this `CreateVirtualCircuitDetails`. The OCID of the compartment to contain the virtual circuit.

Returns The `compartment_id` of this `CreateVirtualCircuitDetails`.

Return type str

cross_connect_mappings

Gets the `cross_connect_mappings` of this `CreateVirtualCircuitDetails`. Create a *CrossConnectMapping* for each cross-connect or cross-connect group this virtual circuit will run on.

Returns The `cross_connect_mappings` of this `CreateVirtualCircuitDetails`.

Return type list[CrossConnectMapping]

customer_bgp_asn

Gets the `customer_bgp_asn` of this `CreateVirtualCircuitDetails`. Your BGP ASN (either public or private). Provide this value only if there's a BGP session that goes from your edge router to Oracle. Otherwise, leave this empty or null.

Returns The `customer_bgp_asn` of this `CreateVirtualCircuitDetails`.

Return type int

display_name

Gets the `display_name` of this `CreateVirtualCircuitDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `CreateVirtualCircuitDetails`.

Return type str

gateway_id

Gets the `gateway_id` of this `CreateVirtualCircuitDetails`. The OCID of the *Drg* that this virtual circuit uses.

Returns The `gateway_id` of this `CreateVirtualCircuitDetails`.

Return type str

provider_name

Gets the `provider_name` of this `CreateVirtualCircuitDetails`. The name of the provider (if you're connecting via a provider). To get a list of the provider names, see `list_fast_connect_provider_services()`.

Returns The `provider_name` of this `CreateVirtualCircuitDetails`.

Return type str

provider_service_name

Gets the `provider_service_name` of this `CreateVirtualCircuitDetails`. The name of the service offered by the provider (if you're connecting via a provider). To get a list of the available service offerings, see `list_fast_connect_provider_services()`.

Returns The `provider_service_name` of this `CreateVirtualCircuitDetails`.

Return type str

region

Gets the region of this `CreateVirtualCircuitDetails`. The Oracle Bare Metal Cloud Services region where this virtual circuit is located.

Example: *phx*

Returns The region of this `CreateVirtualCircuitDetails`.

Return type str

type

Gets the type of this `CreateVirtualCircuitDetails`. The type of IP addresses used in this virtual circuit. PRIVATE means [RFC 1918](#) addresses (10.0.0.0/8, 172.16/12, and 192.168/16). Only PRIVATE is supported.

Allowed values for this property are: "PUBLIC", "PRIVATE"

Returns The type of this `CreateVirtualCircuitDetails`.

Return type str

class `oci.core.models.CreateVnicDetails`**assign_public_ip**

Gets the `assign_public_ip` of this `CreateVnicDetails`. Whether the VNIC should be assigned a public IP address. Defaults to whether the subnet is public or private. If not set and the VNIC is being created in a private subnet (that is, where `prohibitPublicIpOnVnic = true` in the *Subnet*), then no public IP address is assigned. If not set and the subnet is public (`prohibitPublicIpOnVnic = false`), then a public IP address is assigned. If set to true and `prohibitPublicIpOnVnic = true`, an error is returned.

Note: This public IP address is associated with the primary private IP on the VNIC. Secondary private IPs cannot have public IP addresses associated with them. For more information, see [IP Addresses](#).

Example: *false*

Returns The `assign_public_ip` of this `CreateVnicDetails`.

Return type bool

display_name

Gets the `display_name` of this `CreateVnicDetails`. A user-friendly name for the VNIC. Does not have to be unique. Avoid entering confidential information.

Returns The `display_name` of this `CreateVnicDetails`.

Return type str

hostname_label

Gets the `hostname_label` of this `CreateVnicDetails`. The hostname for the VNIC's primary private IP. Used for DNS. The value is the hostname portion of the primary private IP's fully qualified domain name (FQDN) (for example, `binstance-1` in FQDN `binstance-1.subnet123.vcn1.oraclevcn.com`). Must be unique across all VNICs in the subnet and comply with [RFC 952](#) and [RFC 1123](#). The value appears in the `Vnic` object and also the `PrivateIp` object returned by `list_private_ips()` and `get_private_ip()`.

For more information, see [DNS in Your Virtual Cloud Network](#).

When launching an instance, use this `hostnameLabel` instead of the deprecated `hostnameLabel` in `launch_instance_details()`. If you provide both, the values must match.

Example: `binstance-1`

Returns The `hostname_label` of this `CreateVnicDetails`.

Return type str

private_ip

Gets the `private_ip` of this `CreateVnicDetails`. A private IP address of your choice to assign to the VNIC. Must be an available IP address within the subnet's CIDR. If you don't specify a value, Oracle automatically assigns a private IP address from the subnet. This is the VNIC's *primary* private IP address. The value appears in the `Vnic` object and also the `PrivateIp` object returned by `list_private_ips()` and `get_private_ip()`.

Example: `10.0.3.3`

Returns The `private_ip` of this `CreateVnicDetails`.

Return type str

subnet_id

Gets the `subnet_id` of this `CreateVnicDetails`. The OCID of the subnet to create the VNIC in. When launching an instance, use this `subnetId` instead of the deprecated `subnetId` in `launch_instance_details()`. At least one of them is required; if you provide both, the values must match.

Returns The `subnet_id` of this `CreateVnicDetails`.

Return type str

class `oci.core.models.CreateVolumeBackupDetails`

display_name

Gets the `display_name` of this `CreateVolumeBackupDetails`. A user-friendly name for the volume backup. Does not have to be unique and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `CreateVolumeBackupDetails`.

Return type str

volume_id

Gets the `volume_id` of this `CreateVolumeBackupDetails`. The OCID of the volume that needs to be backed up.

Returns The volume_id of this CreateVolumeBackupDetails.

Return type str

class oci.core.models.**CreateVolumeDetails**

availability_domain

Gets the availability_domain of this CreateVolumeDetails. The Availability Domain of the volume.

Example: *Uocm:PHX-AD-1*

Returns The availability_domain of this CreateVolumeDetails.

Return type str

compartment_id

Gets the compartment_id of this CreateVolumeDetails. The OCID of the compartment that contains the volume.

Returns The compartment_id of this CreateVolumeDetails.

Return type str

display_name

Gets the display_name of this CreateVolumeDetails. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The display_name of this CreateVolumeDetails.

Return type str

size_in_mbs

Gets the size_in_mbs of this CreateVolumeDetails. The size of the volume in MBs. The value must be a multiple of 1024.

Returns The size_in_mbs of this CreateVolumeDetails.

Return type int

volume_backup_id

Gets the volume_backup_id of this CreateVolumeDetails. The OCID of the volume backup from which the data should be restored on the newly created volume.

Returns The volume_backup_id of this CreateVolumeDetails.

Return type str

class oci.core.models.**CrossConnect**

compartment_id

Gets the compartment_id of this CrossConnect. The OCID of the compartment containing the cross-connect group.

Returns The compartment_id of this CrossConnect.

Return type str

cross_connect_group_id

Gets the cross_connect_group_id of this CrossConnect. The OCID of the cross-connect group this cross-connect belongs to (if any).

Returns The cross_connect_group_id of this CrossConnect.

Return type str

display_name

Gets the `display_name` of this `CrossConnect`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `CrossConnect`.

Return type str

id

Gets the `id` of this `CrossConnect`. The cross-connect's Oracle ID (OCID).

Returns The `id` of this `CrossConnect`.

Return type str

lifecycle_state

Gets the `lifecycle_state` of this `CrossConnect`. The cross-connect's current state.

Allowed values for this property are: "PENDING_CUSTOMER", "PROVISIONING", "PROVISIONED", "INACTIVE", "TERMINATING", "TERMINATED", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The `lifecycle_state` of this `CrossConnect`.

Return type str

location_name

Gets the `location_name` of this `CrossConnect`. The name of the `FastConnect` location where this cross-connect is installed.

Returns The `location_name` of this `CrossConnect`.

Return type str

port_name

Gets the `port_name` of this `CrossConnect`. A string identifying the meet-me room port for this cross-connect.

Returns The `port_name` of this `CrossConnect`.

Return type str

port_speed_shape_name

Gets the `port_speed_shape_name` of this `CrossConnect`. The port speed for this cross-connect.

Example: *10 Gbps*

Returns The `port_speed_shape_name` of this `CrossConnect`.

Return type str

time_created

Gets the `time_created` of this `CrossConnect`. The date and time the cross-connect was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The `time_created` of this `CrossConnect`.

Return type datetime

`class oci.core.models.CrossConnectGroup`

compartment_id

Gets the compartment_id of this CrossConnectGroup. The OCID of the compartment containing the cross-connect group.

Returns The compartment_id of this CrossConnectGroup.

Return type str

display_name

Gets the display_name of this CrossConnectGroup. The display name of A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The display_name of this CrossConnectGroup.

Return type str

id

Gets the id of this CrossConnectGroup. The cross-connect group's Oracle ID (OCID).

Returns The id of this CrossConnectGroup.

Return type str

lifecycle_state

Gets the lifecycle_state of this CrossConnectGroup. The cross-connect group's current state.

Allowed values for this property are: "PROVISIONING", "PROVISIONED", "INACTIVE", "TERMINATING", "TERMINATED", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The lifecycle_state of this CrossConnectGroup.

Return type str

time_created

Gets the time_created of this CrossConnectGroup. The date and time the cross-connect group was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The time_created of this CrossConnectGroup.

Return type datetime

class oci.core.models.**CrossConnectLocation**

description

Gets the description of this CrossConnectLocation. A description of the location.

Returns The description of this CrossConnectLocation.

Return type str

name

Gets the name of this CrossConnectLocation. The name of the location.

Example: *CyrusOne, Chandler, AZ*

Returns The name of this CrossConnectLocation.

Return type str

class oci.core.models.**CrossConnectMapping**

bgp_md5_auth_key

Gets the `bgp_md5_auth_key` of this `CrossConnectMapping`. The key for BGP MD5 authentication. Only applicable if your system requires MD5 authentication. If empty or not set (null), that means you don't use BGP MD5 authentication.

Returns The `bgp_md5_auth_key` of this `CrossConnectMapping`.

Return type str

cross_connect_or_cross_connect_group_id

Gets the `cross_connect_or_cross_connect_group_id` of this `CrossConnectMapping`. The OCID of the cross-connect or cross-connect group for this mapping. Specified by the owner of the cross-connect or cross-connect group (the customer if the customer is colocated with Oracle, or the provider if the customer is connecting via provider).

Returns The `cross_connect_or_cross_connect_group_id` of this `CrossConnectMapping`.

Return type str

customer_bgp_peering_ip

Gets the `customer_bgp_peering_ip` of this `CrossConnectMapping`. The BGP IP address for the router on the other end of the BGP session from Oracle. Specified by the owner of that router. If the session goes from Oracle to a customer, this is the BGP IP address of the customer's edge router. If the session goes from Oracle to a provider, this is the BGP IP address of the provider's edge router. Must use a /30 or /31 subnet mask.

Example: *10.0.0.18/31*

Returns The `customer_bgp_peering_ip` of this `CrossConnectMapping`.

Return type str

oracle_bgp_peering_ip

Gets the `oracle_bgp_peering_ip` of this `CrossConnectMapping`. The IP address for Oracle's end of the BGP session. Must use a /30 or /31 subnet mask. If the session goes from Oracle to a customer's edge router, the customer specifies this information. If the session goes from Oracle to a provider's edge router, the provider specifies this.

Example: *10.0.0.19/31*

Returns The `oracle_bgp_peering_ip` of this `CrossConnectMapping`.

Return type str

vlan

Gets the `vlan` of this `CrossConnectMapping`. The number of the specific VLAN (on the cross-connect or cross-connect group) that is assigned to this virtual circuit. Specified by the owner of the cross-connect or cross-connect group (the customer if the customer is colocated with Oracle, or the provider if the customer is connecting via provider).

Example: *200*

Returns The `vlan` of this `CrossConnectMapping`.

Return type int

class `oci.core.models.CrossConnectPortSpeedShape`

name

Gets the `name` of this `CrossConnectPortSpeedShape`. The name of the port speed shape.

Example: *10 Gbps*

Returns The name of this CrossConnectPortSpeedShape.

Return type str

port_speed_in_gbps

Gets the port_speed_in_gbps of this CrossConnectPortSpeedShape. The port speed in Gbps.

Example: *10*

Returns The port_speed_in_gbps of this CrossConnectPortSpeedShape.

Return type int

class oci.core.models.**CrossConnectStatus**

cross_connect_id

Gets the cross_connect_id of this CrossConnectStatus. The OCID of the cross-connect.

Returns The cross_connect_id of this CrossConnectStatus.

Return type str

interface_state

Gets the interface_state of this CrossConnectStatus. Whether Oracle's side of the interface is up or down.

Allowed values for this property are: "UP", "DOWN", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The interface_state of this CrossConnectStatus.

Return type str

light_level_ind_bm

Gets the light_level_ind_bm of this CrossConnectStatus. The light level of the cross-connect (in dBm).

Example: *14.0*

Returns The light_level_ind_bm of this CrossConnectStatus.

Return type float

light_level_indicator

Gets the light_level_indicator of this CrossConnectStatus. Status indicator corresponding to the light level.

- NO_LIGHT:** No measurable light
- LOW_WARN:** There's measurable light but it's too low
- HIGH_WARN:** Light level is too high
- BAD:** There's measurable light but the signal-to-noise ratio is bad
- GOOD:** Good light level

Allowed values for this property are: "NO_LIGHT", "LOW_WARN", "HIGH_WARN", "BAD", "GOOD", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The light_level_indicator of this CrossConnectStatus.

Return type str

class oci.core.models.**DhcpDnsOption**

custom_dns_servers

Gets the `custom_dns_servers` of this `DhcpDnsOption`. If you set `serverType` to `CustomDnsServer`, specify the IP address of at least one DNS server of your choice (three maximum).

Returns The `custom_dns_servers` of this `DhcpDnsOption`.

Return type list[str]

server_type

Gets the `server_type` of this `DhcpDnsOption`. - **VcnLocal**: Reserved for future use.

•**VcnLocalPlusInternet**: Also referred to as “Internet and VCN Resolver”.

Instances can resolve internet hostnames (no Internet Gateway is required), and can resolve hostnames of instances in the VCN. This is the default value in the default set of DHCP options in the VCN. For the Internet and VCN Resolver to work across the VCN, there must also be a DNS label set for the VCN, a DNS label set for each subnet, and a hostname for each instance. The Internet and VCN Resolver also enables reverse DNS lookup, which lets you determine the hostname corresponding to the private IP address. For more information, see [DNS in Your Virtual Cloud Network](#).

•**CustomDnsServer**: Instances use a DNS server of your choice (three maximum).

Allowed values for this property are: “VcnLocal”, “VcnLocalPlusInternet”, “CustomDnsServer”, ‘UNKNOWN_ENUM_VALUE’. Any unrecognized values returned by a service will be mapped to ‘UNKNOWN_ENUM_VALUE’.

Returns The `server_type` of this `DhcpDnsOption`.

Return type str

class `oci.core.models.DhcpOption`

static get_subtype (*object_dictionary*)

Given the hash representation of a subtype of this class, use the info in the hash to return the class of the subtype.

type

Gets the type of this `DhcpOption`. The specific DHCP option. Either `DomainNameServer` (for `DhcpDnsOption`) or `SearchDomain` (for `DhcpSearchDomainOption`).

Returns The type of this `DhcpOption`.

Return type str

class `oci.core.models.DhcpOptions`

compartment_id

Gets the `compartment_id` of this `DhcpOptions`. The OCID of the compartment containing the set of DHCP options.

Returns The `compartment_id` of this `DhcpOptions`.

Return type str

display_name

Gets the `display_name` of this `DhcpOptions`. A user-friendly name. Does not have to be unique, and it’s changeable. Avoid entering confidential information.

Returns The `display_name` of this `DhcpOptions`.

Return type str

id

Gets the id of this DhcpOptions. Oracle ID (OCID) for the set of DHCP options.

Returns The id of this DhcpOptions.

Return type str

lifecycle_state

Gets the lifecycle_state of this DhcpOptions. The current state of the set of DHCP options.

Allowed values for this property are: “PROVISIONING”, “AVAILABLE”, “TERMINATING”, “TERMINATED”, ‘UNKNOWN_ENUM_VALUE’. Any unrecognized values returned by a service will be mapped to ‘UNKNOWN_ENUM_VALUE’.

Returns The lifecycle_state of this DhcpOptions.

Return type str

options

Gets the options of this DhcpOptions. The collection of individual DHCP options.

Returns The options of this DhcpOptions.

Return type list[DhcpOption]

time_created

Gets the time_created of this DhcpOptions. Date and time the set of DHCP options was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The time_created of this DhcpOptions.

Return type datetime

vcn_id

Gets the vcn_id of this DhcpOptions. The OCID of the VCN the set of DHCP options belongs to.

Returns The vcn_id of this DhcpOptions.

Return type str

class `oci.core.models.DhcpSearchDomainOption`

search_domain_names

Gets the search_domain_names of this DhcpSearchDomainOption. A single search domain name according to [RFC 952](#) and [RFC 1123](#). During a DNS query, the OS will append this search domain name to the value being queried.

If you set *DhcpDnsOption* to *VcnLocalPlusInternet*, and you assign a DNS label to the VCN during creation, the search domain name in the VCN’s default set of DHCP options is automatically set to the VCN domain (for example, *vcn1.oraclevcn.com*).

If you don’t want to use a search domain name, omit this option from the set of DHCP options. Do not include this option with an empty list of search domain names, or with an empty string as the value for any search domain name.

Returns The search_domain_names of this DhcpSearchDomainOption.

Return type list[str]

class `oci.core.models.Drg`

compartment_id

Gets the `compartment_id` of this Drg. The OCID of the compartment containing the DRG.

Returns The `compartment_id` of this Drg.

Return type str

display_name

Gets the `display_name` of this Drg. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this Drg.

Return type str

id

Gets the `id` of this Drg. The DRG's Oracle ID (OCID).

Returns The `id` of this Drg.

Return type str

lifecycle_state

Gets the `lifecycle_state` of this Drg. The DRG's current state.

Allowed values for this property are: "PROVISIONING", "AVAILABLE", "TERMINATING", "TERMINATED", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The `lifecycle_state` of this Drg.

Return type str

time_created

Gets the `time_created` of this Drg. The date and time the DRG was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The `time_created` of this Drg.

Return type datetime

class oci.core.models.DrgAttachment**compartment_id**

Gets the `compartment_id` of this DrgAttachment. The OCID of the compartment containing the DRG attachment.

Returns The `compartment_id` of this DrgAttachment.

Return type str

display_name

Gets the `display_name` of this DrgAttachment. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this DrgAttachment.

Return type str

drg_id

Gets the `drg_id` of this DrgAttachment. The OCID of the DRG.

Returns The `drg_id` of this DrgAttachment.

Return type str

id

Gets the id of this DrgAttachment. The DRG attachment's Oracle ID (OCID).

Returns The id of this DrgAttachment.

Return type str

lifecycle_state

Gets the lifecycle_state of this DrgAttachment. The DRG attachment's current state.

Allowed values for this property are: "ATTACHING", "ATTACHED", "DETACHING", "DETACHED", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The lifecycle_state of this DrgAttachment.

Return type str

time_created

Gets the time_created of this DrgAttachment. The date and time the DRG attachment was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The time_created of this DrgAttachment.

Return type datetime

vcn_id

Gets the vcn_id of this DrgAttachment. The OCID of the VCN.

Returns The vcn_id of this DrgAttachment.

Return type str

class `oci.core.models.EgressSecurityRule`

destination

Gets the destination of this EgressSecurityRule. The destination CIDR block for the egress rule. This is the range of IP addresses that a packet originating from the instance can go to.

Returns The destination of this EgressSecurityRule.

Return type str

icmp_options

Gets the icmp_options of this EgressSecurityRule. Optional and valid only for ICMP. Use to specify a particular ICMP type and code as defined in [ICMP Parameters](#). If you specify ICMP as the protocol but omit this object, then all ICMP types and codes are allowed. If you do provide this object, the type is required and the code is optional. To enable MTU negotiation for ingress internet traffic, make sure to allow type 3 ("Destination Unreachable") code 4 ("Fragmentation Needed and Don't Fragment was Set"). If you need to specify multiple codes for a single type, create a separate security list rule for each.

Returns The icmp_options of this EgressSecurityRule.

Return type *IcmpOptions*

is_stateless

Gets the is_stateless of this EgressSecurityRule. A stateless rule allows traffic in one direction. Remember to add a corresponding stateless rule in the other direction if you need to support bidirectional traffic. For example, if egress traffic allows TCP destination port 80, there should be an ingress rule to allow TCP

source port 80. Defaults to false, which means the rule is stateful and a corresponding rule is not necessary for bidirectional traffic.

Returns The `is_stateless` of this `EgressSecurityRule`.

Return type `bool`

protocol

Gets the protocol of this `EgressSecurityRule`. The transport protocol. Specify either *all* or an IPv4 protocol number as defined in [Protocol Numbers](#). Options are supported only for ICMP (“1”), TCP (“6”), and UDP (“17”).

Returns The protocol of this `EgressSecurityRule`.

Return type `str`

tcp_options

Gets the `tcp_options` of this `EgressSecurityRule`. Optional and valid only for TCP. Use to specify particular destination ports for TCP rules. If you specify TCP as the protocol but omit this object, then all destination ports are allowed.

Returns The `tcp_options` of this `EgressSecurityRule`.

Return type *TcpOptions*

udp_options

Gets the `udp_options` of this `EgressSecurityRule`. Optional and valid only for UDP. Use to specify particular destination ports for UDP rules. If you specify UDP as the protocol but omit this object, then all destination ports are allowed.

Returns The `udp_options` of this `EgressSecurityRule`.

Return type *UdpOptions*

class `oci.core.models.ExportImageDetails`

destination_type

Gets the `destination_type` of this `ExportImageDetails`. The destination type. Use *objectStorageTuple* when specifying the namespace, bucket name, and object name. Use *objectStorageUri* when specifying the Object Storage Service URL.

Returns The `destination_type` of this `ExportImageDetails`.

Return type `str`

static get_subtype (*object_dictionary*)

Given the hash representation of a subtype of this class, use the info in the hash to return the class of the subtype.

class `oci.core.models.ExportImageViaObjectStorageTupleDetails`

bucket_name

Gets the `bucket_name` of this `ExportImageViaObjectStorageTupleDetails`. The Object Storage Service bucket to export the image to.

Returns The `bucket_name` of this `ExportImageViaObjectStorageTupleDetails`.

Return type `str`

namespace_name

Gets the `namespace_name` of this `ExportImageViaObjectStorageTupleDetails`. The Object Storage Service namespace to export the image to.

Returns The namespace_name of this ExportImageViaObjectStorageTupleDetails.

Return type str

object_name

Gets the object_name of this ExportImageViaObjectStorageTupleDetails. The Object Storage Service object name for the exported image.

Returns The object_name of this ExportImageViaObjectStorageTupleDetails.

Return type str

class `oci.core.models.ExportImageViaObjectStorageUriDetails`

destination_uri

Gets the destination_uri of this ExportImageViaObjectStorageUriDetails. The Object Storage Service URL to export the image to. See [Object Storage URLs](#) and [pre-authenticated requests](#) for constructing URLs for image import/export.

Returns The destination_uri of this ExportImageViaObjectStorageUriDetails.

Return type str

class `oci.core.models.FastConnectProviderService`

description

Gets the description of this FastConnectProviderService. A description of the service offered by the provider.

Returns The description of this FastConnectProviderService.

Return type str

provider_name

Gets the provider_name of this FastConnectProviderService. The name of the provider.

Returns The provider_name of this FastConnectProviderService.

Return type str

provider_service_name

Gets the provider_service_name of this FastConnectProviderService. The name of the service offered by the provider.

Returns The provider_service_name of this FastConnectProviderService.

Return type str

class `oci.core.models.IPSecConnection`

compartment_id

Gets the compartment_id of this IPSecConnection. The OCID of the compartment containing the IPSec connection.

Returns The compartment_id of this IPSecConnection.

Return type str

cpe_id

Gets the cpe_id of this IPSecConnection. The OCID of the CPE.

Returns The cpe_id of this IPSecConnection.

Return type str

display_name

Gets the `display_name` of this `IPSecConnection`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `IPSecConnection`.

Return type str

drg_id

Gets the `drg_id` of this `IPSecConnection`. The OCID of the DRG.

Returns The `drg_id` of this `IPSecConnection`.

Return type str

id

Gets the `id` of this `IPSecConnection`. The IPsec connection's Oracle ID (OCID).

Returns The `id` of this `IPSecConnection`.

Return type str

lifecycle_state

Gets the `lifecycle_state` of this `IPSecConnection`. The IPsec connection's current state.

Allowed values for this property are: "PROVISIONING", "AVAILABLE", "TERMINATING", "TERMINATED", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The `lifecycle_state` of this `IPSecConnection`.

Return type str

static_routes

Gets the `static_routes` of this `IPSecConnection`. Static routes to the CPE. At least one route must be included. The CIDR must not be a multicast address or class E address.

Example: *10.0.1.0/24*

Returns The `static_routes` of this `IPSecConnection`.

Return type list[str]

time_created

Gets the `time_created` of this `IPSecConnection`. The date and time the IPsec connection was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The `time_created` of this `IPSecConnection`.

Return type datetime

class `oci.core.models.IPSecConnectionDeviceConfig`

compartment_id

Gets the `compartment_id` of this `IPSecConnectionDeviceConfig`. The OCID of the compartment containing the IPsec connection.

Returns The `compartment_id` of this `IPSecConnectionDeviceConfig`.

Return type str

id

Gets the id of this IPSecConnectionDeviceConfig. The IPSec connection's Oracle ID (OCID).

Returns The id of this IPSecConnectionDeviceConfig.

Return type str

time_created

Gets the time_created of this IPSecConnectionDeviceConfig. The date and time the IPSec connection was created.

Returns The time_created of this IPSecConnectionDeviceConfig.

Return type datetime

tunnels

Gets the tunnels of this IPSecConnectionDeviceConfig. Two *TunnelConfig* objects.

Returns The tunnels of this IPSecConnectionDeviceConfig.

Return type list[TunnelConfig]

class oci.core.models.**IPSecConnectionDeviceStatus****compartment_id**

Gets the compartment_id of this IPSecConnectionDeviceStatus. The OCID of the compartment containing the IPSec connection.

Returns The compartment_id of this IPSecConnectionDeviceStatus.

Return type str

id

Gets the id of this IPSecConnectionDeviceStatus. The IPSec connection's Oracle ID (OCID).

Returns The id of this IPSecConnectionDeviceStatus.

Return type str

time_created

Gets the time_created of this IPSecConnectionDeviceStatus. The date and time the IPSec connection was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The time_created of this IPSecConnectionDeviceStatus.

Return type datetime

tunnels

Gets the tunnels of this IPSecConnectionDeviceStatus. Two *TunnelStatus* objects.

Returns The tunnels of this IPSecConnectionDeviceStatus.

Return type list[TunnelStatus]

class oci.core.models.**IScsiVolumeAttachment****chap_secret**

Gets the chap_secret of this IScsiVolumeAttachment. The Challenge-Handshake-Authentication-Protocol (CHAP) secret valid for the associated CHAP user name. (Also called the "CHAP password".)

Example: *d6866c0d-298b-48ba-95af-309b4faux45e*

Returns The chap_secret of this IScsiVolumeAttachment.

Return type str

chap_username

Gets the chap_username of this IScsiVolumeAttachment. The volume's system-generated Challenge-Handshake-Authentication-Protocol (CHAP) user name.

Example: *ocidl.volume.oc1.phx.abyhqljrgvttnlx73nmrwfaux7kcvzfs3s66izvxf2h4lgvyndsdsnoiwr5q*

Returns The chap_username of this IScsiVolumeAttachment.

Return type str

ipv4

Gets the ipv4 of this IScsiVolumeAttachment. The volume's iSCSI IP address.

Example: *169.254.0.2*

Returns The ipv4 of this IScsiVolumeAttachment.

Return type str

iqn

Gets the iqn of this IScsiVolumeAttachment. The target volume's iSCSI Qualified Name in the format defined by RFC 3720.

Example: *iqn.2015-12.us.oracle.com:456b0391-17b8-4122-bbf1-f85fc0bb97d9*

Returns The iqn of this IScsiVolumeAttachment.

Return type str

port

Gets the port of this IScsiVolumeAttachment. The volume's iSCSI port.

Example: *3260*

Returns The port of this IScsiVolumeAttachment.

Return type int

class `oci.core.models.IcmpOptions`

code

Gets the code of this IcmpOptions. The ICMP code (optional).

Returns The code of this IcmpOptions.

Return type int

type

Gets the type of this IcmpOptions. The ICMP type.

Returns The type of this IcmpOptions.

Return type int

class `oci.core.models.Image`

base_image_id

Gets the base_image_id of this Image. The OCID of the image originally used to launch the instance.

Returns The base_image_id of this Image.

Return type str

compartment_id

Gets the `compartment_id` of this Image. The OCID of the compartment containing the instance you want to use as the basis for the image.

Returns The `compartment_id` of this Image.

Return type str

create_image_allowed

Gets the `create_image_allowed` of this Image. Whether instances launched with this image can be used to create new images. For example, you cannot create an image of an Oracle Database instance.

Example: *true*

Returns The `create_image_allowed` of this Image.

Return type bool

display_name

Gets the `display_name` of this Image. A user-friendly name for the image. It does not have to be unique, and it's changeable. Avoid entering confidential information. You cannot use an Oracle-provided image name as a custom image name.

Example: *My custom Oracle Linux image*

Returns The `display_name` of this Image.

Return type str

id

Gets the `id` of this Image. The OCID of the image.

Returns The `id` of this Image.

Return type str

lifecycle_state

Gets the `lifecycle_state` of this Image. Allowed values for this property are: "PROVISIONING", "IMPORTING", "AVAILABLE", "EXPORTING", "DISABLED", "DELETED", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The `lifecycle_state` of this Image.

Return type str

operating_system

Gets the `operating_system` of this Image. The image's operating system.

Example: *Oracle Linux*

Returns The `operating_system` of this Image.

Return type str

operating_system_version

Gets the `operating_system_version` of this Image. The image's operating system version.

Example: *7.2*

Returns The `operating_system_version` of this Image.

Return type str

time_created

Gets the `time_created` of this Image. The date and time the image was created, in the format defined by RFC3339.

Example: `2016-08-25T21:10:29.600Z`

Returns The `time_created` of this Image.

Return type `datetime`

class `oci.core.models.ImageSourceDetails`

static get_subtype (*object_dictionary*)

Given the hash representation of a subtype of this class, use the info in the hash to return the class of the subtype.

source_type

Gets the `source_type` of this `ImageSourceDetails`. The source type for the image. Use *objectStorageTuple* when specifying the namespace, bucket name, and object name. Use *objectStorageUri* when specifying the Object Storage Service URL.

Returns The `source_type` of this `ImageSourceDetails`.

Return type `str`

class `oci.core.models.ImageSourceViaObjectStorageTupleDetails`

bucket_name

Gets the `bucket_name` of this `ImageSourceViaObjectStorageTupleDetails`. The Object Storage Service bucket for the image.

Returns The `bucket_name` of this `ImageSourceViaObjectStorageTupleDetails`.

Return type `str`

namespace_name

Gets the `namespace_name` of this `ImageSourceViaObjectStorageTupleDetails`. The Object Storage Service namespace for the image.

Returns The `namespace_name` of this `ImageSourceViaObjectStorageTupleDetails`.

Return type `str`

object_name

Gets the `object_name` of this `ImageSourceViaObjectStorageTupleDetails`. The Object Storage Service name for the image.

Returns The `object_name` of this `ImageSourceViaObjectStorageTupleDetails`.

Return type `str`

class `oci.core.models.ImageSourceViaObjectStorageUriDetails`

source_uri

Gets the `source_uri` of this `ImageSourceViaObjectStorageUriDetails`. The Object Storage Service URL for the image.

Returns The `source_uri` of this `ImageSourceViaObjectStorageUriDetails`.

Return type `str`

`class oci.core.models.IngressSecurityRule`

icmp_options

Gets the `icmp_options` of this `IngressSecurityRule`. Optional and valid only for ICMP. Use to specify a particular ICMP type and code as defined in [ICMP Parameters](#). If you specify ICMP as the protocol but omit this object, then all ICMP types and codes are allowed. If you do provide this object, the type is required and the code is optional. To enable MTU negotiation for ingress internet traffic, make sure to allow type 3 (“Destination Unreachable”) code 4 (“Fragmentation Needed and Don’t Fragment was Set”). If you need to specify multiple codes for a single type, create a separate security list rule for each.

Returns The `icmp_options` of this `IngressSecurityRule`.

Return type *IcmpOptions*

is_stateless

Gets the `is_stateless` of this `IngressSecurityRule`. A stateless rule allows traffic in one direction. Remember to add a corresponding stateless rule in the other direction if you need to support bidirectional traffic. For example, if ingress traffic allows TCP destination port 80, there should be an egress rule to allow TCP source port 80. Defaults to false, which means the rule is stateful and a corresponding rule is not necessary for bidirectional traffic.

Returns The `is_stateless` of this `IngressSecurityRule`.

Return type bool

protocol

Gets the protocol of this `IngressSecurityRule`. The transport protocol. Specify either *all* or an IPv4 protocol number as defined in [Protocol Numbers](#). Options are supported only for ICMP (“1”), TCP (“6”), and UDP (“17”).

Returns The protocol of this `IngressSecurityRule`.

Return type str

source

Gets the source of this `IngressSecurityRule`. The source CIDR block for the ingress rule. This is the range of IP addresses that a packet coming into the instance can come from.

Returns The source of this `IngressSecurityRule`.

Return type str

tcp_options

Gets the `tcp_options` of this `IngressSecurityRule`. Optional and valid only for TCP. Use to specify particular destination ports for TCP rules. If you specify TCP as the protocol but omit this object, then all destination ports are allowed.

Returns The `tcp_options` of this `IngressSecurityRule`.

Return type *TcpOptions*

udp_options

Gets the `udp_options` of this `IngressSecurityRule`. Optional and valid only for UDP. Use to specify particular destination ports for UDP rules. If you specify UDP as the protocol but omit this object, then all destination ports are allowed.

Returns The `udp_options` of this `IngressSecurityRule`.

Return type *UdpOptions*

`class oci.core.models.Instance`

availability_domain

Gets the `availability_domain` of this Instance. The Availability Domain the instance is running in.

Example: *Uocm:PHX-AD-1*

Returns The `availability_domain` of this Instance.

Return type str

compartment_id

Gets the `compartment_id` of this Instance. The OCID of the compartment that contains the instance.

Returns The `compartment_id` of this Instance.

Return type str

display_name

Gets the `display_name` of this Instance. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Example: *My bare metal instance*

Returns The `display_name` of this Instance.

Return type str

extended_metadata

Gets the `extended_metadata` of this Instance. Additional metadata key/value pairs that you provide. They serve a similar purpose and functionality from fields in the 'metadata' object.

They are distinguished from 'metadata' fields in that these can be nested JSON objects (whereas 'metadata' fields are string/string maps only).

If you don't need nested metadata values, it is strongly advised to avoid using this object and use the Metadata object instead.

Returns The `extended_metadata` of this Instance.

Return type dict(str, object)

id

Gets the `id` of this Instance. The OCID of the instance.

Returns The `id` of this Instance.

Return type str

image_id

Gets the `image_id` of this Instance. The image used to boot the instance. You can enumerate all available images by calling `list_images()`.

Returns The `image_id` of this Instance.

Return type str

ipxe_script

Gets the `ipxe_script` of this Instance. When an Oracle Bare Metal Cloud Services or virtual machine instance boots, the iPXE firmware that runs on the instance is configured to run an iPXE script to continue the boot process.

If you want more control over the boot process, you can provide your own custom iPXE script that will run when the instance boots; however, you should be aware that the same iPXE script will run every time an instance boots; not only after the initial `LaunchInstance` call.

The default iPXE script connects to the instance's local boot volume over iSCSI and performs a network boot. If you use a custom iPXE script and want to network-boot from the instance's local boot volume

over iSCSI the same way as the default iPXE script, you should use the following iSCSI IP address: 169.254.0.2, and boot volume IQN: iqn.2015-02.oracle.boot.

For more information about the Bring Your Own Image feature of Oracle Bare Metal Cloud Services, see [Bring Your Own Image](#).

For more information about iPXE, see <http://ipxe.org>.

Returns The ipxe_script of this Instance.

Return type str

lifecycle_state

Gets the lifecycle_state of this Instance. The current state of the instance.

Allowed values for this property are: “PROVISIONING”, “RUNNING”, “STARTING”, “STOPPING”, “STOPPED”, “CREATING_IMAGE”, “TERMINATING”, “TERMINATED”, ‘UNKNOWN_ENUM_VALUE’. Any unrecognized values returned by a service will be mapped to ‘UNKNOWN_ENUM_VALUE’.

Returns The lifecycle_state of this Instance.

Return type str

metadata

Gets the metadata of this Instance. Custom metadata that you provide.

Returns The metadata of this Instance.

Return type dict(str, str)

region

Gets the region of this Instance. The region that contains the Availability Domain the instance is running in.

Example: *phx*

Returns The region of this Instance.

Return type str

shape

Gets the shape of this Instance. The shape of the instance. The shape determines the number of CPUs and the amount of memory allocated to the instance. You can enumerate all available shapes by calling `list_shapes()`.

Returns The shape of this Instance.

Return type str

time_created

Gets the time_created of this Instance. The date and time the instance was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The time_created of this Instance.

Return type datetime

class `oci.core.models.InstanceConsoleConnection`

compartment_id

Gets the compartment_id of this InstanceConsoleConnection. The OCID of the compartment to contain the ConsoleConnection

Returns The compartment_id of this InstanceConsoleConnection.

Return type str

connection_string

Gets the connection_string of this InstanceConsoleConnection. The ssh connection string to the instance console

Returns The connection_string of this InstanceConsoleConnection.

Return type str

fingerprint

Gets the fingerprint of this InstanceConsoleConnection. The fingerprint of the ssh publicKey.

Returns The fingerprint of this InstanceConsoleConnection.

Return type str

id

Gets the id of this InstanceConsoleConnection. The OCID of the instance console connection

Returns The id of this InstanceConsoleConnection.

Return type str

instance_id

Gets the instance_id of this InstanceConsoleConnection. The host instance OCID

Returns The instance_id of this InstanceConsoleConnection.

Return type str

lifecycle_state

Gets the lifecycle_state of this InstanceConsoleConnection. The current state of the instance console connection.

Allowed values for this property are: "ACTIVE", "CREATING", "DELETED", "DELETING", "FAILED", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The lifecycle_state of this InstanceConsoleConnection.

Return type str

class oci.core.models.**InstanceCredentials**

password

Gets the password of this InstanceCredentials. The password for the username.

Returns The password of this InstanceCredentials.

Return type str

username

Gets the username of this InstanceCredentials. The username.

Returns The username of this InstanceCredentials.

Return type str

class oci.core.models.**InternetGateway**

compartment_id

Gets the compartment_id of this InternetGateway. The OCID of the compartment containing the Internet Gateway.

Returns The compartment_id of this InternetGateway.

Return type str

display_name

Gets the display_name of this InternetGateway. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The display_name of this InternetGateway.

Return type str

id

Gets the id of this InternetGateway. The Internet Gateway's Oracle ID (OCID).

Returns The id of this InternetGateway.

Return type str

is_enabled

Gets the is_enabled of this InternetGateway. Whether the gateway is enabled. When the gateway is disabled, traffic is not routed to/from the Internet, regardless of route rules.

Returns The is_enabled of this InternetGateway.

Return type bool

lifecycle_state

Gets the lifecycle_state of this InternetGateway. The Internet Gateway's current state.

Allowed values for this property are: "PROVISIONING", "AVAILABLE", "TERMINATING", "TERMINATED", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The lifecycle_state of this InternetGateway.

Return type str

time_created

Gets the time_created of this InternetGateway. The date and time the Internet Gateway was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The time_created of this InternetGateway.

Return type datetime

vcn_id

Gets the vcn_id of this InternetGateway. The OCID of the VCN the Internet Gateway belongs to.

Returns The vcn_id of this InternetGateway.

Return type str

class oci.core.models.**LaunchInstanceDetails**

availability_domain

Gets the availability_domain of this LaunchInstanceDetails. The Availability Domain of the instance.

Example: *Uocm:PHX-AD-1*

Returns The `availability_domain` of this `LaunchInstanceDetails`.

Return type str

compartment_id

Gets the `compartment_id` of this `LaunchInstanceDetails`. The OCID of the compartment.

Returns The `compartment_id` of this `LaunchInstanceDetails`.

Return type str

create_vnic_details

Gets the `create_vnic_details` of this `LaunchInstanceDetails`. Details for the primary VNIC, which is automatically created and attached when the instance is launched.

Returns The `create_vnic_details` of this `LaunchInstanceDetails`.

Return type *CreateVnicDetails*

display_name

Gets the `display_name` of this `LaunchInstanceDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Example: *My bare metal instance*

Returns The `display_name` of this `LaunchInstanceDetails`.

Return type str

extended_metadata

Gets the `extended_metadata` of this `LaunchInstanceDetails`. Additional metadata key/value pairs that you provide. They serve a similar purpose and functionality from fields in the 'metadata' object.

They are distinguished from 'metadata' fields in that these can be nested JSON objects (whereas 'metadata' fields are string/string maps only).

If you don't need nested metadata values, it is strongly advised to avoid using this object and use the Metadata object instead.

Returns The `extended_metadata` of this `LaunchInstanceDetails`.

Return type dict(str, object)

hostname_label

Gets the `hostname_label` of this `LaunchInstanceDetails`. Deprecated. Instead use *hostnameLabel* in *CreateVnicDetails*. If you provide both, the values must match.

Returns The `hostname_label` of this `LaunchInstanceDetails`.

Return type str

image_id

Gets the `image_id` of this `LaunchInstanceDetails`. The OCID of the image used to boot the instance.

Returns The `image_id` of this `LaunchInstanceDetails`.

Return type str

ipxe_script

Gets the `ipxe_script` of this `LaunchInstanceDetails`. This is an advanced option.

When an Oracle Bare Metal Cloud Services or virtual machine instance boots, the iPXE firmware that runs on the instance is configured to run an iPXE script to continue the boot process.

If you want more control over the boot process, you can provide your own custom iPXE script that will run when the instance boots; however, you should be aware that the same iPXE script will run every time an instance boots; not only after the initial `LaunchInstance` call.

The default iPXE script connects to the instance's local boot volume over iSCSI and performs a network boot. If you use a custom iPXE script and want to network-boot from the instance's local boot volume over iSCSI the same way as the default iPXE script, you should use the following iSCSI IP address: 169.254.0.2, and boot volume IQN: `iqn.2015-02.oracle.boot`.

For more information about the Bring Your Own Image feature of Oracle Bare Metal Cloud Services, see [Bring Your Own Image](#).

For more information about iPXE, see <http://ipxe.org>.

Returns The `ipxe_script` of this `LaunchInstanceDetails`.

Return type `str`

metadata

Gets the metadata of this `LaunchInstanceDetails`. Custom metadata key/value pairs that you provide, such as the SSH public key required to connect to the instance.

A metadata service runs on every launched instance. The service is an HTTP endpoint listening on 169.254.169.254. You can use the service to:

- Provide information to [Cloud-Init](#) to be used for various system initialization tasks.
- Get information about the instance, including the custom metadata that you provide when you launch the instance.

Providing Cloud-Init Metadata

You can use the following metadata key names to provide information to Cloud-Init:

“ssh_authorized_keys” - Provide one or more public SSH keys to be included in the `~/.ssh/authorized_keys` file for the default user on the instance. Use a newline character to separate multiple keys. The SSH keys must be in the format necessary for the `authorized_keys` file, as shown in the example below.

“user_data” - Provide your own base64-encoded data to be used by Cloud-Init to run custom scripts or provide custom Cloud-Init configuration. For information about how to take advantage of user data, see the [Cloud-Init Documentation](#).

Note: Cloud-Init does not pull this data from the `http://169.254.169.254/oci/v1/instance/metadata/` path. When the instance launches and either of these keys are provided, the key values are formatted as OpenStack metadata and copied to the following locations, which are recognized by Cloud-Init:

`http://169.254.169.254/openstack/latest/meta_data.json` - This JSON blob contains, among other things, the SSH keys that you provided for

“ssh_authorized_keys”.

`http://169.254.169.254/openstack/latest/user_data` - Contains the base64-decoded data that you provided for **“user_data”**.

Metadata Example

```
“metadata” [{ “quake_bot_level” : “Severe”, “ssh_authorized_keys” : “ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQCAQCZ06fccNTQfq+xubFIJ5ZR3kt+uzspdH9tXL+IAejSM1NXM+
ryan.smith@company.com ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAAQEAzJSAtwEPoB3Jmr58IXrDGzLuD
rsa-key-20160227”, “user_data” : “SWYgeW91IGNhbiBzZWUgdGhpcywgGhlbiBpdCB3b3JrZWQgbWF5Yn
```

```
}

```

Getting Metadata on the Instance

To get information about your instance, connect to the instance using SSH and issue any of the following GET requests:

```
curl http://169.254.169.254/oci/v1/instance/ curl http://169.254.169.254/oci/v1/instance/metadata/ curl http://169.254.169.254/oci/v1/instance/metadata/<any-key-name>
```

You'll get back a response that includes all the instance information; only the metadata information; or the metadata information for the specified key name, respectively.

Returns The metadata of this LaunchInstanceDetails.

Return type dict(str, str)

shape

Gets the shape of this LaunchInstanceDetails. The shape of an instance. The shape determines the number of CPUs, amount of memory, and other resources allocated to the instance.

You can enumerate all available shapes by calling `list_shapes()`.

Returns The shape of this LaunchInstanceDetails.

Return type str

subnet_id

Gets the `subnet_id` of this LaunchInstanceDetails. *Deprecated.* Instead use `subnetId` in `CreateVnicDetails`. At least one of them is required; if you provide both, the values must match.

Returns The `subnet_id` of this LaunchInstanceDetails.

Return type str

class `oci.core.models.LetterOfAuthority`

circuit_type

Gets the `circuit_type` of this LetterOfAuthority. The type of cross-connect fiber, termination, and optical specification.

Allowed values for this property are: "Single_mode_LC", "Single_mode_SC", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The `circuit_type` of this LetterOfAuthority.

Return type str

cross_connect_id

Gets the `cross_connect_id` of this LetterOfAuthority. The OCID of the cross-connect.

Returns The `cross_connect_id` of this LetterOfAuthority.

Return type str

facility_location

Gets the `facility_location` of this LetterOfAuthority. The address of the FastConnect location.

Returns The `facility_location` of this LetterOfAuthority.

Return type str

port_name

Gets the port_name of this LetterOfAuthority. The meet-me room port for this cross-connect.

Returns The port_name of this LetterOfAuthority.

Return type str

time_expires

Gets the time_expires of this LetterOfAuthority. The date and time when the Letter of Authority expires, in the format defined by RFC3339.

Returns The time_expires of this LetterOfAuthority.

Return type datetime

time_issued

Gets the time_issued of this LetterOfAuthority. The date and time the Letter of Authority was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The time_issued of this LetterOfAuthority.

Return type datetime

class oci.core.models.**PortRange**

max

Gets the max of this PortRange. The maximum port number. Must not be lower than the minimum port number. To specify a single port number, set both the min and max to the same value.

Returns The max of this PortRange.

Return type int

min

Gets the min of this PortRange. The minimum port number. Must not be greater than the maximum port number.

Returns The min of this PortRange.

Return type int

class oci.core.models.**PrivateIp**

availability_domain

Gets the availability_domain of this PrivateIp. The private IP's Availability Domain.

Example: *Uocm:PHX-AD-1*

Returns The availability_domain of this PrivateIp.

Return type str

compartment_id

Gets the compartment_id of this PrivateIp. The OCID of the compartment containing the private IP.

Returns The compartment_id of this PrivateIp.

Return type str

display_name

Gets the display_name of this PrivateIp. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `PrivateIp`.

Return type str

hostname_label

Gets the `hostname_label` of this `PrivateIp`. The hostname for the private IP. Used for DNS. The value is the hostname portion of the private IP's fully qualified domain name (FQDN) (for example, *binstance-1* in FQDN *binstance-1.subnet123.vcn1.oraclevcn.com*). Must be unique across all VNICs in the subnet and comply with [RFC 952](#) and [RFC 1123](#).

For more information, see [DNS in Your Virtual Cloud Network](#).

Example: *binstance-1*

Returns The `hostname_label` of this `PrivateIp`.

Return type str

id

Gets the `id` of this `PrivateIp`. The private IP's Oracle ID (OCID).

Returns The `id` of this `PrivateIp`.

Return type str

ip_address

Gets the `ip_address` of this `PrivateIp`. The private IP address of the *privateIp* object. The address is within the CIDR of the VNIC's subnet.

Example: *10.0.3.3*

Returns The `ip_address` of this `PrivateIp`.

Return type str

is_primary

Gets the `is_primary` of this `PrivateIp`. Whether this private IP is the primary one on the VNIC. Primary private IPs are unassigned and deleted automatically when the VNIC is terminated.

Example: *true*

Returns The `is_primary` of this `PrivateIp`.

Return type bool

subnet_id

Gets the `subnet_id` of this `PrivateIp`. The OCID of the subnet the VNIC is in.

Returns The `subnet_id` of this `PrivateIp`.

Return type str

time_created

Gets the `time_created` of this `PrivateIp`. The date and time the private IP was created, in the format defined by [RFC3339](#).

Example: *2016-08-25T21:10:29.600Z*

Returns The `time_created` of this `PrivateIp`.

Return type datetime

vnic_id

Gets the `vnic_id` of this `PrivateIp`. The OCID of the VNIC the private IP is assigned to. The VNIC and private IP must be in the same subnet.

Returns The `vnic_id` of this `PrivateIp`.

Return type str

class `oci.core.models.RouteRule`

cidr_block

Gets the `cidr_block` of this `RouteRule`. A destination IP address range in CIDR notation. Matching packets will be routed to the indicated network entity (the target).

Example: `0.0.0.0/0`

Returns The `cidr_block` of this `RouteRule`.

Return type str

network_entity_id

Gets the `network_entity_id` of this `RouteRule`. The OCID for the route rule's target.

Returns The `network_entity_id` of this `RouteRule`.

Return type str

class `oci.core.models.RouteTable`

compartment_id

Gets the `compartment_id` of this `RouteTable`. The OCID of the compartment containing the route table.

Returns The `compartment_id` of this `RouteTable`.

Return type str

display_name

Gets the `display_name` of this `RouteTable`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `RouteTable`.

Return type str

id

Gets the `id` of this `RouteTable`. The route table's Oracle ID (OCID).

Returns The `id` of this `RouteTable`.

Return type str

lifecycle_state

Gets the `lifecycle_state` of this `RouteTable`. The route table's current state.

Allowed values for this property are: "PROVISIONING", "AVAILABLE", "TERMINATING", "TERMINATED", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The `lifecycle_state` of this `RouteTable`.

Return type str

route_rules

Gets the `route_rules` of this `RouteTable`. The collection of rules for routing destination IPs to network devices.

Returns The `route_rules` of this `RouteTable`.

Return type list[RouteRule]

time_created

Gets the `time_created` of this `RouteTable`. The date and time the route table was created, in the format defined by RFC3339.

Example: `2016-08-25T21:10:29.600Z`

Returns The `time_created` of this `RouteTable`.

Return type `datetime`

vcn_id

Gets the `vcn_id` of this `RouteTable`. The OCID of the VCN the route table list belongs to.

Returns The `vcn_id` of this `RouteTable`.

Return type `str`

class `oci.core.models.SecurityList`

compartment_id

Gets the `compartment_id` of this `SecurityList`. The OCID of the compartment containing the security list.

Returns The `compartment_id` of this `SecurityList`.

Return type `str`

display_name

Gets the `display_name` of this `SecurityList`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `SecurityList`.

Return type `str`

egress_security_rules

Gets the `egress_security_rules` of this `SecurityList`. Rules for allowing egress IP packets.

Returns The `egress_security_rules` of this `SecurityList`.

Return type `list[EgressSecurityRule]`

id

Gets the `id` of this `SecurityList`. The security list's Oracle Cloud ID (OCID).

Returns The `id` of this `SecurityList`.

Return type `str`

ingress_security_rules

Gets the `ingress_security_rules` of this `SecurityList`. Rules for allowing ingress IP packets.

Returns The `ingress_security_rules` of this `SecurityList`.

Return type `list[IngressSecurityRule]`

lifecycle_state

Gets the `lifecycle_state` of this `SecurityList`. The security list's current state.

Allowed values for this property are: "PROVISIONING", "AVAILABLE", "TERMINATING", "TERMINATED", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The `lifecycle_state` of this `SecurityList`.

Return type `str`

time_created

Gets the `time_created` of this SecurityList. The date and time the security list was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The `time_created` of this SecurityList.

Return type `datetime`

vcn_id

Gets the `vcn_id` of this SecurityList. The OCID of the VCN the security list belongs to.

Returns The `vcn_id` of this SecurityList.

Return type `str`

class `oci.core.models.Shape`

shape

Gets the `shape` of this Shape. The name of the shape. You can enumerate all available shapes by calling `list_shapes()`.

Returns The `shape` of this Shape.

Return type `str`

class `oci.core.models.Subnet`

availability_domain

Gets the `availability_domain` of this Subnet. The subnet's Availability Domain.

Example: *Uocm:PHX-AD-1*

Returns The `availability_domain` of this Subnet.

Return type `str`

cidr_block

Gets the `cidr_block` of this Subnet. The subnet's CIDR block.

Example: *172.16.1.0/24*

Returns The `cidr_block` of this Subnet.

Return type `str`

compartment_id

Gets the `compartment_id` of this Subnet. The OCID of the compartment containing the subnet.

Returns The `compartment_id` of this Subnet.

Return type `str`

dhcp_options_id

Gets the `dhcp_options_id` of this Subnet. The OCID of the set of DHCP options associated with the subnet.

Returns The `dhcp_options_id` of this Subnet.

Return type `str`

display_name

Gets the `display_name` of this Subnet. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this Subnet.

Return type str

dns_label

Gets the `dns_label` of this Subnet. A DNS label for the subnet, used in conjunction with the VNIC's hostname and VCN's DNS label to form a fully qualified domain name (FQDN) for each VNIC within this subnet (for example, *binstance-1.subnet123.vcn1.oraclevcn.com*). Must be an alphanumeric string that begins with a letter and is unique within the VCN. The value cannot be changed.

The absence of this parameter means the Internet and VCN Resolver will not resolve hostnames of instances in this subnet.

For more information, see [DNS in Your Virtual Cloud Network](#).

Example: *subnet123*

Returns The `dns_label` of this Subnet.

Return type str

id

Gets the `id` of this Subnet. The subnet's Oracle ID (OCID).

Returns The `id` of this Subnet.

Return type str

lifecycle_state

Gets the `lifecycle_state` of this Subnet. The subnet's current state.

Allowed values for this property are: "PROVISIONING", "AVAILABLE", "TERMINATING", "TERMINATED", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The `lifecycle_state` of this Subnet.

Return type str

prohibit_public_ip_on_vnic

Gets the `prohibit_public_ip_on_vnic` of this Subnet. Whether VNICs within this subnet can have public IP addresses. Defaults to false, which means VNICs created in this subnet will automatically be assigned public IP addresses unless specified otherwise during instance launch or VNIC creation (with the *assignPublicIp* flag in *CreateVnicDetails*). If *prohibitPublicIpOnVnic* is set to true, VNICs created in this subnet cannot have public IP addresses (that is, it's a private subnet).

Example: *true*

Returns The `prohibit_public_ip_on_vnic` of this Subnet.

Return type bool

route_table_id

Gets the `route_table_id` of this Subnet. The OCID of the route table the subnet is using.

Returns The `route_table_id` of this Subnet.

Return type str

security_list_ids

Gets the `security_list_ids` of this Subnet. OCIDs for the security lists to use for VNICs in this subnet.

Returns The `security_list_ids` of this Subnet.

Return type list[str]

subnet_domain_name

Gets the `subnet_domain_name` of this Subnet. The subnet's domain name, which consists of the subnet's DNS label, the VCN's DNS label, and the `oraclevcn.com` domain.

For more information, see [DNS in Your Virtual Cloud Network](#).

Example: `subnet123.vcn1.oraclevcn.com`

Returns The `subnet_domain_name` of this Subnet.

Return type str

time_created

Gets the `time_created` of this Subnet. The date and time the subnet was created, in the format defined by RFC3339.

Example: `2016-08-25T21:10:29.600Z`

Returns The `time_created` of this Subnet.

Return type datetime

vcn_id

Gets the `vcn_id` of this Subnet. The OCID of the VCN the subnet is in.

Returns The `vcn_id` of this Subnet.

Return type str

virtual_router_ip

Gets the `virtual_router_ip` of this Subnet. The IP address of the virtual router.

Example: `10.0.14.1`

Returns The `virtual_router_ip` of this Subnet.

Return type str

virtual_router_mac

Gets the `virtual_router_mac` of this Subnet. The MAC address of the virtual router.

Example: `00:00:17:B6:4D:DD`

Returns The `virtual_router_mac` of this Subnet.

Return type str

class `oci.core.models.TcpOptions`

destination_port_range

Gets the `destination_port_range` of this `TcpOptions`. An inclusive range of allowed destination ports. Use the same number for the min and max to indicate a single port. Defaults to all ports if not specified.

Returns The `destination_port_range` of this `TcpOptions`.

Return type *PortRange*

source_port_range

Gets the `source_port_range` of this `TcpOptions`. An inclusive range of allowed source ports. Use the same number for the min and max to indicate a single port. Defaults to all ports if not specified.

Returns The `source_port_range` of this `TcpOptions`.

Return type *PortRange*

class `oci.core.models.TunnelConfig`

ip_address

Gets the `ip_address` of this `TunnelConfig`. The IP address of Oracle's VPN headend.

Example: *129.146.17.50*

Returns The `ip_address` of this `TunnelConfig`.

Return type str

shared_secret

Gets the `shared_secret` of this `TunnelConfig`. The shared secret of the IPsec tunnel.

Example: *vFG2IF6TWq4UToUiLSRDoJEUs6j1c.p8G.dVQxiMfMO0yXMLi.IZTbYIWhGu4V8o*

Returns The `shared_secret` of this `TunnelConfig`.

Return type str

time_created

Gets the `time_created` of this `TunnelConfig`. The date and time the IPsec connection was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The `time_created` of this `TunnelConfig`.

Return type datetime

class `oci.core.models.TunnelStatus`

ip_address

Gets the `ip_address` of this `TunnelStatus`. The IP address of Oracle's VPN headend.

Example: *129.146.17.50*

Returns The `ip_address` of this `TunnelStatus`.

Return type str

lifecycle_state

Gets the `lifecycle_state` of this `TunnelStatus`. The tunnel's current state.

Allowed values for this property are: "UP", "DOWN", "DOWN_FOR_MAINTENANCE", "UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The `lifecycle_state` of this `TunnelStatus`.

Return type str

time_created

Gets the `time_created` of this `TunnelStatus`. The date and time the IPsec connection was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The `time_created` of this `TunnelStatus`.

Return type datetime

time_state_modified

Gets the `time_state_modified` of this `TunnelStatus`. When the state of the tunnel last changed, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The `time_state_modified` of this `TunnelStatus`.

Return type `datetime`

class `oci.core.models.UdpOptions`

destination_port_range

Gets the `destination_port_range` of this `UdpOptions`. An inclusive range of allowed destination ports. Use the same number for the min and max to indicate a single port. Defaults to all ports if not specified.

Returns The `destination_port_range` of this `UdpOptions`.

Return type *PortRange*

source_port_range

Gets the `source_port_range` of this `UdpOptions`. An inclusive range of allowed source ports. Use the same number for the min and max to indicate a single port. Defaults to all ports if not specified.

Returns The `source_port_range` of this `UdpOptions`.

Return type *PortRange*

class `oci.core.models.UpdateCpeDetails`

display_name

Gets the `display_name` of this `UpdateCpeDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `UpdateCpeDetails`.

Return type `str`

class `oci.core.models.UpdateCrossConnectDetails`

display_name

Gets the `display_name` of this `UpdateCrossConnectDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `UpdateCrossConnectDetails`.

Return type `str`

is_active

Gets the `is_active` of this `UpdateCrossConnectDetails`. Set to true to activate the cross-connect. You activate it after the physical cabling is complete, and you've confirmed the cross-connect's light levels are good and your side of the interface is up. Activation indicates to Oracle that the physical connection is ready.

Example: *true*

Returns The `is_active` of this `UpdateCrossConnectDetails`.

Return type `bool`

class `oci.core.models.UpdateCrossConnectGroupDetails`

display_name

Gets the `display_name` of this `UpdateCrossConnectGroupDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `UpdateCrossConnectGroupDetails`.

Return type str

class `oci.core.models.UpdateDhcpDetails`

display_name

Gets the `display_name` of this `UpdateDhcpDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `UpdateDhcpDetails`.

Return type str

options

Gets the options of this `UpdateDhcpDetails`.

Returns The options of this `UpdateDhcpDetails`.

Return type list[DhcpOption]

class `oci.core.models.UpdateDrgAttachmentDetails`

display_name

Gets the `display_name` of this `UpdateDrgAttachmentDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `UpdateDrgAttachmentDetails`.

Return type str

class `oci.core.models.UpdateDrgDetails`

display_name

Gets the `display_name` of this `UpdateDrgDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `UpdateDrgDetails`.

Return type str

class `oci.core.models.UpdateIPSecConnectionDetails`

display_name

Gets the `display_name` of this `UpdateIPSecConnectionDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `UpdateIPSecConnectionDetails`.

Return type str

class `oci.core.models.UpdateImageDetails`

display_name

Gets the `display_name` of this `UpdateImageDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Example: *My custom Oracle Linux image*

Returns The `display_name` of this `UpdateImageDetails`.

Return type str

class `oci.core.models.UpdateInstanceDetails`

display_name

Gets the `display_name` of this `UpdateInstanceDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Example: *My bare metal instance*

Returns The `display_name` of this `UpdateInstanceDetails`.

Return type str

class `oci.core.models.UpdateInternetGatewayDetails`

display_name

Gets the `display_name` of this `UpdateInternetGatewayDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `UpdateInternetGatewayDetails`.

Return type str

is_enabled

Gets the `is_enabled` of this `UpdateInternetGatewayDetails`. Whether the gateway is enabled.

Returns The `is_enabled` of this `UpdateInternetGatewayDetails`.

Return type bool

class `oci.core.models.UpdatePrivateIpDetails`

display_name

Gets the `display_name` of this `UpdatePrivateIpDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `UpdatePrivateIpDetails`.

Return type str

hostname_label

Gets the `hostname_label` of this `UpdatePrivateIpDetails`. The hostname for the private IP. Used for DNS. The value is the hostname portion of the private IP's fully qualified domain name (FQDN) (for example, *binstance-1* in FQDN *binstance-1.subnet123.vcn1.oraclevcn.com*). Must be unique across all VNICs in the subnet and comply with [RFC 952](#) and [RFC 1123](#).

For more information, see [DNS in Your Virtual Cloud Network](#).

Example: *binstance-1*

Returns The `hostname_label` of this `UpdatePrivateIpDetails`.

Return type str

vnic_id

Gets the `vnic_id` of this `UpdatePrivateIpDetails`. The OCID of the VNIC to reassign the private IP to. The VNIC must be in the same subnet as the current VNIC.

Returns The `vnic_id` of this `UpdatePrivateIpDetails`.

Return type str

```
class oci.core.models.UpdateRouteTableDetails
```

display_name

Gets the `display_name` of this `UpdateRouteTableDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `UpdateRouteTableDetails`.

Return type str

route_rules

Gets the `route_rules` of this `UpdateRouteTableDetails`. The collection of rules used for routing destination IPs to network devices.

Returns The `route_rules` of this `UpdateRouteTableDetails`.

Return type list[RouteRule]

```
class oci.core.models.UpdateSecurityListDetails
```

display_name

Gets the `display_name` of this `UpdateSecurityListDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `UpdateSecurityListDetails`.

Return type str

egress_security_rules

Gets the `egress_security_rules` of this `UpdateSecurityListDetails`. Rules for allowing egress IP packets.

Returns The `egress_security_rules` of this `UpdateSecurityListDetails`.

Return type list[EgressSecurityRule]

ingress_security_rules

Gets the `ingress_security_rules` of this `UpdateSecurityListDetails`. Rules for allowing ingress IP packets.

Returns The `ingress_security_rules` of this `UpdateSecurityListDetails`.

Return type list[IngressSecurityRule]

```
class oci.core.models.UpdateSubnetDetails
```

display_name

Gets the `display_name` of this `UpdateSubnetDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `UpdateSubnetDetails`.

Return type str

```
class oci.core.models.UpdateVcnDetails
```

display_name

Gets the `display_name` of this `UpdateVcnDetails`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `UpdateVcnDetails`.

Return type str

class `oci.core.models.UpdateVirtualCircuitDetails`

bandwidth_shape_name

Gets the `bandwidth_shape_name` of this `UpdateVirtualCircuitDetails`. The provisioned data rate of the connection. To get a list of the available bandwidth levels (that is, shapes), see `list_virtual_circuit_bandwidth_shapes()`.

To be updated only by the customer who owns the virtual circuit.

Returns The `bandwidth_shape_name` of this `UpdateVirtualCircuitDetails`.

Return type str

cross_connect_mappings

Gets the `cross_connect_mappings` of this `UpdateVirtualCircuitDetails`. An array of mappings, each containing properties for a cross-connect or cross-connect group associated with this virtual circuit.

The customer and provider can update different properties in the mapping depending on the situation. See the description of the [CrossConnectMapping](#).

Returns The `cross_connect_mappings` of this `UpdateVirtualCircuitDetails`.

Return type list[CrossConnectMapping]

customer_bgp_asn

Gets the `customer_bgp_asn` of this `UpdateVirtualCircuitDetails`. The BGP ASN of the network at the other end of the BGP session from Oracle.

If the BGP session is from the customer's edge router to Oracle, the required value is the customer's ASN, and it can be updated only by the customer.

If the BGP session is from the provider's edge router to Oracle, the required value is the provider's ASN, and it can be updated only by the provider.

Returns The `customer_bgp_asn` of this `UpdateVirtualCircuitDetails`.

Return type int

display_name

Gets the `display_name` of this `UpdateVirtualCircuitDetails`. A user-friendly name. Does not have to be unique. Avoid entering confidential information.

To be updated only by the customer who owns the virtual circuit.

Returns The `display_name` of this `UpdateVirtualCircuitDetails`.

Return type str

gateway_id

Gets the `gateway_id` of this `UpdateVirtualCircuitDetails`. The OCID of the [Drg](#) that this virtual circuit uses.

To be updated only by the customer who owns the virtual circuit.

Returns The `gateway_id` of this `UpdateVirtualCircuitDetails`.

Return type str

provider_state

Gets the `provider_state` of this `UpdateVirtualCircuitDetails`. The provider's state in relation to this virtual circuit. Relevant only if the customer is using FastConnect via a provider. `ACTIVE` means the provider has provisioned the virtual circuit from their end. `INACTIVE` means the provider has not yet provisioned the virtual circuit, or has de-provisioned it.

To be updated only by the provider.

Allowed values for this property are: “ACTIVE”, “INACTIVE”

Returns The provider_state of this UpdateVirtualCircuitDetails.

Return type str

reference_comment

Gets the reference_comment of this UpdateVirtualCircuitDetails. Provider-supplied reference information about this virtual circuit. Relevant only if the customer is using FastConnect via a provider.

To be updated only by the provider.

Returns The reference_comment of this UpdateVirtualCircuitDetails.

Return type str

class oci.core.models.UpdateVnicDetails

display_name

Gets the display_name of this UpdateVnicDetails. A user-friendly name. Does not have to be unique, and it's changeable.

Returns The display_name of this UpdateVnicDetails.

Return type str

hostname_label

Gets the hostname_label of this UpdateVnicDetails. The hostname for the VNIC's primary private IP. Used for DNS. The value is the hostname portion of the primary private IP's fully qualified domain name (FQDN) (for example, *binstance-1* in FQDN *binstance-1.subnet123.vcn1.oraclevcn.com*). Must be unique across all VNICs in the subnet and comply with [RFC 952](#) and [RFC 1123](#). The value appears in the *Vnic* object and also the *PrivateIp* object returned by `list_private_ips()` and `get_private_ip()`.

For more information, see [DNS in Your Virtual Cloud Network](#).

Returns The hostname_label of this UpdateVnicDetails.

Return type str

class oci.core.models.UpdateVolumeBackupDetails

display_name

Gets the display_name of this UpdateVolumeBackupDetails. A friendly user-specified name for the volume backup. Avoid entering confidential information.

Returns The display_name of this UpdateVolumeBackupDetails.

Return type str

class oci.core.models.UpdateVolumeDetails

display_name

Gets the display_name of this UpdateVolumeDetails. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The display_name of this UpdateVolumeDetails.

Return type str

`class oci.core.models.Vcn`

cidr_block

Gets the `cidr_block` of this `Vcn`. The CIDR IP address block of the VCN.

Example: *172.16.0.0/16*

Returns The `cidr_block` of this `Vcn`.

Return type str

compartment_id

Gets the `compartment_id` of this `Vcn`. The OCID of the compartment containing the VCN.

Returns The `compartment_id` of this `Vcn`.

Return type str

default_dhcp_options_id

Gets the `default_dhcp_options_id` of this `Vcn`. The OCID for the VCN's default set of DHCP options.

Returns The `default_dhcp_options_id` of this `Vcn`.

Return type str

default_route_table_id

Gets the `default_route_table_id` of this `Vcn`. The OCID for the VCN's default route table.

Returns The `default_route_table_id` of this `Vcn`.

Return type str

default_security_list_id

Gets the `default_security_list_id` of this `Vcn`. The OCID for the VCN's default security list.

Returns The `default_security_list_id` of this `Vcn`.

Return type str

display_name

Gets the `display_name` of this `Vcn`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `Vcn`.

Return type str

dns_label

Gets the `dns_label` of this `Vcn`. A DNS label for the VCN, used in conjunction with the VNIC's hostname and subnet's DNS label to form a fully qualified domain name (FQDN) for each VNIC within this subnet (for example, *binstance-1.subnet123.vcn1.oraclevcn.com*). Must be an alphanumeric string that begins with a letter. The value cannot be changed.

The absence of this parameter means the Internet and VCN Resolver will not work for this VCN.

For more information, see [DNS in Your Virtual Cloud Network](#).

Example: *vcn1*

Returns The `dns_label` of this `Vcn`.

Return type str

id

Gets the `id` of this `Vcn`. The VCN's Oracle ID (OCID).

Returns The id of this Vcn.

Return type str

lifecycle_state

Gets the `lifecycle_state` of this Vcn. The VCN's current state.

Allowed values for this property are: "PROVISIONING", "AVAILABLE", "TERMINATING", "TERMINATED", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The `lifecycle_state` of this Vcn.

Return type str

time_created

Gets the `time_created` of this Vcn. The date and time the VCN was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The `time_created` of this Vcn.

Return type datetime

vcn_domain_name

Gets the `vcn_domain_name` of this Vcn. The VCN's domain name, which consists of the VCN's DNS label, and the *oraclevcn.com* domain.

For more information, see [DNS in Your Virtual Cloud Network](#).

Example: *vcn1.oraclevcn.com*

Returns The `vcn_domain_name` of this Vcn.

Return type str

class `oci.core.models.VirtualCircuit`

bandwidth_shape_name

Gets the `bandwidth_shape_name` of this VirtualCircuit. The provisioned data rate of the connection.

Returns The `bandwidth_shape_name` of this VirtualCircuit.

Return type str

bgp_session_state

Gets the `bgp_session_state` of this VirtualCircuit. The state of the BGP session associated with the virtual circuit.

Allowed values for this property are: "UP", "DOWN", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The `bgp_session_state` of this VirtualCircuit.

Return type str

compartment_id

Gets the `compartment_id` of this VirtualCircuit. The OCID of the compartment containing the virtual circuit.

Returns The `compartment_id` of this VirtualCircuit.

Return type str

cross_connect_mappings

Gets the `cross_connect_mappings` of this `VirtualCircuit`. An array of mappings, each containing properties for a cross-connect or cross-connect group that is associated with this virtual circuit.

Returns The `cross_connect_mappings` of this `VirtualCircuit`.

Return type list[CrossConnectMapping]

customer_bgp_asn

Gets the `customer_bgp_asn` of this `VirtualCircuit`. The BGP ASN of the network at the other end of the BGP session from Oracle. If the session is between the customer's edge router and Oracle, the value is the customer's ASN. If the BGP session is between the provider's edge router and Oracle, the value is the provider's ASN.

Returns The `customer_bgp_asn` of this `VirtualCircuit`.

Return type int

display_name

Gets the `display_name` of this `VirtualCircuit`. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The `display_name` of this `VirtualCircuit`.

Return type str

gateway_id

Gets the `gateway_id` of this `VirtualCircuit`. The OCID of the customer's *Drg* that this virtual circuit uses.

Returns The `gateway_id` of this `VirtualCircuit`.

Return type str

id

Gets the `id` of this `VirtualCircuit`. The virtual circuit's Oracle ID (OCID).

Returns The `id` of this `VirtualCircuit`.

Return type str

lifecycle_state

Gets the `lifecycle_state` of this `VirtualCircuit`. The virtual circuit's current state. For information about the different states, see [FastConnect Overview](#).

Allowed values for this property are: "PENDING_PROVIDER", "VERIFYING", "PROVISIONING", "PROVISIONED", "FAILED", "INACTIVE", "TERMINATING", "TERMINATED", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The `lifecycle_state` of this `VirtualCircuit`.

Return type str

oracle_bgp_asn

Gets the `oracle_bgp_asn` of this `VirtualCircuit`. The Oracle BGP ASN.

Returns The `oracle_bgp_asn` of this `VirtualCircuit`.

Return type int

provider_name

Gets the `provider_name` of this `VirtualCircuit`. The name of the provider (if the customer is connecting via a provider).

Returns The `provider_name` of this `VirtualCircuit`.

Return type str

provider_service_name

Gets the `provider_service_name` of this `VirtualCircuit`. The name of the service offered by the provider (if the customer is connecting via a provider).

Returns The `provider_service_name` of this `VirtualCircuit`.

Return type str

provider_state

Gets the `provider_state` of this `VirtualCircuit`. The provider's state in relation to this virtual circuit (if the customer is connecting via a provider). `ACTIVE` means the provider has provisioned the virtual circuit from their end. `INACTIVE` means the provider has not yet provisioned the virtual circuit, or has de-provisioned it.

Allowed values for this property are: "ACTIVE", "INACTIVE", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The `provider_state` of this `VirtualCircuit`.

Return type str

reference_comment

Gets the `reference_comment` of this `VirtualCircuit`. Provider-supplied reference information about this virtual circuit (if the customer is connecting via a provider).

Returns The `reference_comment` of this `VirtualCircuit`.

Return type str

region

Gets the region of this `VirtualCircuit`. The Oracle Bare Metal Cloud Services region where this virtual circuit is located.

Returns The region of this `VirtualCircuit`.

Return type str

time_created

Gets the `time_created` of this `VirtualCircuit`. The date and time the virtual circuit was created, in the format defined by RFC3339.

Example: `2016-08-25T21:10:29.600Z`

Returns The `time_created` of this `VirtualCircuit`.

Return type datetime

type

Gets the type of this `VirtualCircuit`. The type of IP addresses used in this virtual circuit. `PRIVATE` means RFC 1918 addresses (10.0.0.0/8, 172.16/12, and 192.168/16). Only `PRIVATE` is supported.

Allowed values for this property are: "PUBLIC", "PRIVATE", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The type of this `VirtualCircuit`.

Return type str

`class oci.core.models.VirtualCircuitBandwidthShape`

bandwidth_in_mbps

Gets the `bandwidth_in_mbps` of this `VirtualCircuitBandwidthShape`. The bandwidth in Mbps.

Example: *10000*

Returns The `bandwidth_in_mbps` of this `VirtualCircuitBandwidthShape`.

Return type int

name

Gets the name of this `VirtualCircuitBandwidthShape`. The name of the bandwidth shape.

Example: *10 Gbps*

Returns The name of this `VirtualCircuitBandwidthShape`.

Return type str

class `oci.core.models.Vnic`

availability_domain

Gets the `availability_domain` of this `Vnic`. The VNIC's Availability Domain.

Example: *Uocm:PHX-AD-1*

Returns The `availability_domain` of this `Vnic`.

Return type str

compartment_id

Gets the `compartment_id` of this `Vnic`. The OCID of the compartment containing the VNIC.

Returns The `compartment_id` of this `Vnic`.

Return type str

display_name

Gets the `display_name` of this `Vnic`. A user-friendly name. Does not have to be unique. Avoid entering confidential information.

Returns The `display_name` of this `Vnic`.

Return type str

hostname_label

Gets the `hostname_label` of this `Vnic`. The hostname for the VNIC's primary private IP. Used for DNS. The value is the hostname portion of the primary private IP's fully qualified domain name (FQDN) (for example, *binstance-1* in FQDN *binstance-1.subnet123.vcn1.oraclevcn.com*). Must be unique across all VNICs in the subnet and comply with [RFC 952](#) and [RFC 1123](#).

For more information, see [DNS in Your Virtual Cloud Network](#).

Example: *binstance-1*

Returns The `hostname_label` of this `Vnic`.

Return type str

id

Gets the `id` of this `Vnic`. The OCID of the VNIC.

Returns The `id` of this `Vnic`.

Return type str

is_primary

Gets the `is_primary` of this `Vnic`. Whether the VNIC is the primary VNIC (the VNIC that is automatically created and attached during instance launch).

Returns The `is_primary` of this Vnic.

Return type bool

lifecycle_state

Gets the `lifecycle_state` of this Vnic. The current state of the VNIC.

Allowed values for this property are: “PROVISIONING”, “AVAILABLE”, “TERMINATING”, “TERMINATED”, ‘UNKNOWN_ENUM_VALUE’. Any unrecognized values returned by a service will be mapped to ‘UNKNOWN_ENUM_VALUE’.

Returns The `lifecycle_state` of this Vnic.

Return type str

mac_address

Gets the `mac_address` of this Vnic. The MAC address of the VNIC.

Example: `00:00:17:B6:4D:DD`

Returns The `mac_address` of this Vnic.

Return type str

private_ip

Gets the `private_ip` of this Vnic. The private IP address of the primary `privateIp` object on the VNIC. The address is within the CIDR of the VNIC’s subnet.

Example: `10.0.3.3`

Returns The `private_ip` of this Vnic.

Return type str

public_ip

Gets the `public_ip` of this Vnic. The public IP address of the VNIC, if one is assigned.

Returns The `public_ip` of this Vnic.

Return type str

subnet_id

Gets the `subnet_id` of this Vnic. The OCID of the subnet the VNIC is in.

Returns The `subnet_id` of this Vnic.

Return type str

time_created

Gets the `time_created` of this Vnic. The date and time the VNIC was created, in the format defined by RFC3339.

Example: `2016-08-25T21:10:29.600Z`

Returns The `time_created` of this Vnic.

Return type datetime

class `oci.core.models.VnicAttachment`

availability_domain

Gets the `availability_domain` of this VnicAttachment. The Availability Domain of the instance.

Example: `Ucm:PHX-AD-1`

Returns The `availability_domain` of this VnicAttachment.

Return type str

compartment_id

Gets the compartment_id of this VnicAttachment. The OCID of the compartment the VNIC attachment is in, which is the same compartment the instance is in.

Returns The compartment_id of this VnicAttachment.

Return type str

display_name

Gets the display_name of this VnicAttachment. A user-friendly name. Does not have to be unique. Avoid entering confidential information.

Returns The display_name of this VnicAttachment.

Return type str

id

Gets the id of this VnicAttachment. The OCID of the VNIC attachment.

Returns The id of this VnicAttachment.

Return type str

instance_id

Gets the instance_id of this VnicAttachment. The OCID of the instance.

Returns The instance_id of this VnicAttachment.

Return type str

lifecycle_state

Gets the lifecycle_state of this VnicAttachment. The current state of the VNIC attachment.

Allowed values for this property are: "ATTACHING", "ATTACHED", "DETACHING", "DETACHED", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The lifecycle_state of this VnicAttachment.

Return type str

subnet_id

Gets the subnet_id of this VnicAttachment. The OCID of the VNIC's subnet.

Returns The subnet_id of this VnicAttachment.

Return type str

time_created

Gets the time_created of this VnicAttachment. The date and time the VNIC attachment was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The time_created of this VnicAttachment.

Return type datetime

vlan_tag

Gets the vlan_tag of this VnicAttachment. The Oracle-assigned VLAN tag of the attached VNIC. Available after the attachment process is complete.

Example: *0*

Returns The vlan_tag of this VnicAttachment.

Return type int

vnic_id

Gets the vnic_id of this VnicAttachment. The OCID of the VNIC. Available after the attachment process is complete.

Returns The vnic_id of this VnicAttachment.

Return type str

class `oci.core.models.Volume`

availability_domain

Gets the availability_domain of this Volume. The Availability Domain of the volume.

Example: *Uocm:PHX-AD-1*

Returns The availability_domain of this Volume.

Return type str

compartment_id

Gets the compartment_id of this Volume. The OCID of the compartment that contains the volume.

Returns The compartment_id of this Volume.

Return type str

display_name

Gets the display_name of this Volume. A user-friendly name. Does not have to be unique, and it's changeable. Avoid entering confidential information.

Returns The display_name of this Volume.

Return type str

id

Gets the id of this Volume. The volume's Oracle ID (OCID).

Returns The id of this Volume.

Return type str

lifecycle_state

Gets the lifecycle_state of this Volume. The current state of a volume.

Allowed values for this property are: "PROVISIONING", "RESTORING", "AVAILABLE", "TERMINATING", "TERMINATED", "FAULTY", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The lifecycle_state of this Volume.

Return type str

size_in_mbs

Gets the size_in_mbs of this Volume. The size of the volume in MBs. The value must be a multiple of 1024.

Returns The size_in_mbs of this Volume.

Return type int

time_created

Gets the time_created of this Volume. The date and time the volume was created. Format defined by RFC3339.

Returns The `time_created` of this Volume.

Return type `datetime`

class `oci.core.models.VolumeAttachment`

attachment_type

Gets the `attachment_type` of this VolumeAttachment. The type of volume attachment.

Returns The `attachment_type` of this VolumeAttachment.

Return type `str`

availability_domain

Gets the `availability_domain` of this VolumeAttachment. The Availability Domain of an instance.

Example: *Uocm:PHX-AD-1*

Returns The `availability_domain` of this VolumeAttachment.

Return type `str`

compartment_id

Gets the `compartment_id` of this VolumeAttachment. The OCID of the compartment.

Returns The `compartment_id` of this VolumeAttachment.

Return type `str`

display_name

Gets the `display_name` of this VolumeAttachment. A user-friendly name. Does not have to be unique, and it cannot be changed. Avoid entering confidential information.

Example: *My volume attachment*

Returns The `display_name` of this VolumeAttachment.

Return type `str`

static get_subtype (*object_dictionary*)

Given the hash representation of a subtype of this class, use the info in the hash to return the class of the subtype.

id

Gets the `id` of this VolumeAttachment. The OCID of the volume attachment.

Returns The `id` of this VolumeAttachment.

Return type `str`

instance_id

Gets the `instance_id` of this VolumeAttachment. The OCID of the instance the volume is attached to.

Returns The `instance_id` of this VolumeAttachment.

Return type `str`

lifecycle_state

Gets the `lifecycle_state` of this VolumeAttachment. The current state of the volume attachment.

Allowed values for this property are: "ATTACHING", "ATTACHED", "DETACHING", "DETACHED", "UNKNOWN_ENUM_VALUE". Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The `lifecycle_state` of this VolumeAttachment.

Return type str

time_created

Gets the time_created of this VolumeAttachment. The date and time the volume was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The time_created of this VolumeAttachment.

Return type datetime

volume_id

Gets the volume_id of this VolumeAttachment. The OCID of the volume.

Returns The volume_id of this VolumeAttachment.

Return type str

class `oci.core.models.VolumeBackup`

compartment_id

Gets the compartment_id of this VolumeBackup. The OCID of the compartment that contains the volume backup.

Returns The compartment_id of this VolumeBackup.

Return type str

display_name

Gets the display_name of this VolumeBackup. A user-friendly name for the volume backup. Does not have to be unique and it's changeable. Avoid entering confidential information.

Returns The display_name of this VolumeBackup.

Return type str

id

Gets the id of this VolumeBackup. The OCID of the volume backup.

Returns The id of this VolumeBackup.

Return type str

lifecycle_state

Gets the lifecycle_state of this VolumeBackup. The current state of a volume backup.

Allowed values for this property are: "CREATING", "AVAILABLE", "TERMINATING", "TERMINATED", "FAULTY", "REQUEST_RECEIVED", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The lifecycle_state of this VolumeBackup.

Return type str

size_in_mbs

Gets the size_in_mbs of this VolumeBackup. The size of the volume, in MBs. The value must be a multiple of 1024.

Returns The size_in_mbs of this VolumeBackup.

Return type int

time_created

Gets the `time_created` of this `VolumeBackup`. The date and time the volume backup was created. This is the time the actual point-in-time image of the volume data was taken. Format defined by RFC3339.

Returns The `time_created` of this `VolumeBackup`.

Return type `datetime`

time_request_received

Gets the `time_request_received` of this `VolumeBackup`. The date and time the request to create the volume backup was received. Format defined by RFC3339.

Returns The `time_request_received` of this `VolumeBackup`.

Return type `datetime`

unique_size_in_mbs

Gets the `unique_size_in_mbs` of this `VolumeBackup`. The size used by the backup, in MBs. It is typically smaller than `sizeInMBs`, depending on the space consumed on the volume and whether the backup is full or incremental.

Returns The `unique_size_in_mbs` of this `VolumeBackup`.

Return type `int`

volume_id

Gets the `volume_id` of this `VolumeBackup`. The OCID of the volume.

Returns The `volume_id` of this `VolumeBackup`.

Return type `str`

Identity

Client

`class oci.identity.identity_client.IdentityClient` (*config*)

`add_user_to_group` (*add_user_to_group_details*, ***kwargs*)

`AddUserToGroup` Adds the specified user to the specified group and returns a `UserGroupMembership` object with its own OCID.

After you send your request, the new object's `lifecycleState` will temporarily be `CREATING`. Before using the object, first make sure its `lifecycleState` has changed to `ACTIVE`.

Parameters

- **`add_user_to_group_details`** (`AddUserToGroupDetails`) – (required) Request object for adding a user to a group.
- **`opc_retry_token`** (`str`) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (e.g., if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A `Response` object with data of type `UserGroupMembership`

Return type `Response`

create_compartment (*create_compartment_details*, ***kwargs*)

CreateCompartment Creates a new compartment in your tenancy.

Important: Compartments cannot be renamed or deleted.

You must specify your tenancy's OCID as the compartment ID in the request object. Remember that the tenancy is simply the root compartment. For information about OCIDs, see [Resource Identifiers](#).

You must also specify a *name* for the compartment, which must be unique across all compartments in your tenancy and cannot be changed. You can use this name or the OCID when writing policies that apply to the compartment. For more information about policies, see [How Policies Work](#).

You must also specify a *description* for the compartment (although it can be an empty string). It does not have to be unique, and you can change it anytime with `update_compartment()`.

After you send your request, the new object's *lifecycleState* will temporarily be CREATING. Before using the object, first make sure its *lifecycleState* has changed to ACTIVE.

Parameters

- **create_compartment_details** (*CreateCompartmentDetails*) – (required) Request object for creating a new compartment.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (e.g., if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *Compartment*

Return type *Response*

create_customer_secret_key (*create_customer_secret_key_details*, *user_id*, ***kwargs*)

CreateCustomerSecretKey Creates a new secret key for the specified user. Secret keys are used for authentication with the Object Storage Service's Amazon S3 compatible API. For information, see [Managing User Credentials](#).

You must specify a *description* for the secret key (although it can be an empty string). It does not have to be unique, and you can change it anytime with `update_customer_secret_key()`.

Every user has permission to create a secret key for *their own user ID*. An administrator in your organization does not need to write a policy to give users this ability. To compare, administrators who have permission to the tenancy can use this operation to create a secret key for any user, including themselves.

Parameters

- **create_customer_secret_key_details** (*CreateCustomerSecretKeyDetails*) – (required) Request object for creating a new secret key.
- **user_id** (*str*) – (required) The OCID of the user.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (e.g., if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *CustomerSecretKey*

Return type *Response*

create_group (*create_group_details*, ***kwargs*)

CreateGroup Creates a new group in your tenancy.

You must specify your tenancy's OCID as the compartment ID in the request object (remember that the tenancy is simply the root compartment). Notice that IAM resources (users, groups, compartments, and some policies) reside within the tenancy itself, unlike cloud resources such as compute instances, which typically reside within compartments inside the tenancy. For information about OCIDs, see [Resource Identifiers](#).

You must also specify a *name* for the group, which must be unique across all groups in your tenancy and cannot be changed. You can use this name or the OCID when writing policies that apply to the group. For more information about policies, see [How Policies Work](#).

You must also specify a *description* for the group (although it can be an empty string). It does not have to be unique, and you can change it anytime with `update_group()`.

After you send your request, the new object's *lifecycleState* will temporarily be `CREATING`. Before using the object, first make sure its *lifecycleState* has changed to `ACTIVE`.

After creating the group, you need to put users in it and write policies for it. See `add_user_to_group()` and `create_policy()`.

Parameters

- **`create_group_details`** (`CreateGroupDetails`) – (required) Request object for creating a new group.
- **`opc_retry_token`** (`str`) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (e.g., if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A `Response` object with data of type `Group`

Return type `Response`

`create_identity_provider` (`create_identity_provider_details`, `**kwargs`)

`CreateIdentityProvider` Creates a new identity provider in your tenancy. For more information, see [Identity Providers and Federation](#).

You must specify your tenancy's OCID as the compartment ID in the request object. Remember that the tenancy is simply the root compartment. For information about OCIDs, see [Resource Identifiers](#).

You must also specify a *name* for the `IdentityProvider`, which must be unique across all `IdentityProvider` objects in your tenancy and cannot be changed.

You must also specify a *description* for the `IdentityProvider` (although it can be an empty string). It does not have to be unique, and you can change it anytime with `update_identity_provider()`.

After you send your request, the new object's *lifecycleState* will temporarily be `CREATING`. Before using the object, first make sure its *lifecycleState* has changed to `ACTIVE`.

Parameters

- **`create_identity_provider_details`** (`CreateIdentityProviderDetails`) – (required) Request object for creating a new SAML2 identity provider.
- **`opc_retry_token`** (`str`) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (e.g., if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A `Response` object with data of type `IdentityProvider`

Return type *Response*

create_idp_group_mapping (*create_idp_group_mapping_details*, *identity_provider_id*, ***kwargs*)

CreateIdpGroupMapping Creates a single mapping between an IdP group and an IAM Service Group.

Parameters

- **create_idp_group_mapping_details** (*CreateIdpGroupMappingDetails*) – (required) Add a mapping from an SAML2.0 identity provider group to a BMC group.
- **identity_provider_id** (*str*) – (required) The OCID of the identity provider.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (e.g., if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *IdpGroupMapping*

Return type *Response*

create_or_reset_ui_password (*user_id*, ***kwargs*)

CreateOrResetUIPassword Creates a new Console one-time password for the specified user. For more information about user credentials, see [User Credentials](#).

Use this operation after creating a new user, or if a user forgets their password. The new one-time password is returned to you in the response, and you must securely deliver it to the user. They'll be prompted to change this password the next time they sign in to the Console. If they don't change it within 7 days, the password will expire and you'll need to create a new one-time password for the user.

Note: The user's Console login is the unique name you specified when you created the user (see `create_user()`).

Parameters

- **user_id** (*str*) – (required) The OCID of the user.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (e.g., if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *UIPassword*

Return type *Response*

create_policy (*create_policy_details*, ***kwargs*)

CreatePolicy Creates a new policy in the specified compartment (either the tenancy or another of your compartments). If you're new to policies, see [Getting Started with Policies](#).

You must specify a *name* for the policy, which must be unique across all policies in your tenancy and cannot be changed.

You must also specify a *description* for the policy (although it can be an empty string). It does not have to be unique, and you can change it anytime with `update_policy()`.

You must specify one or more policy statements in the *statements* array. For information about writing policies, see [How Policies Work](#) and [Common Policies](#).

After you send your request, the new object's *lifecycleState* will temporarily be `CREATING`. Before using the object, first make sure its *lifecycleState* has changed to `ACTIVE`.

New policies take effect typically within 10 seconds.

Parameters

- **create_policy_details** (`CreatePolicyDetails`) – (required) Request object for creating a new policy.
- **opc_retry_token** (`str`) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (e.g., if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A `Response` object with data of type `Policy`

Return type `Response`

create_region_subscription (`create_region_subscription_details`, `tenancy_id`, `**kwargs`)
`CreateRegionSubscription` Creates a subscription to a region for a tenancy.

Parameters

- **create_region_subscription_details** (`CreateRegionSubscriptionDetails`) – (required) Request object for activate a new region.
- **tenancy_id** (`str`) – (required) The OCID of the tenancy.
- **opc_retry_token** (`str`) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (e.g., if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A `Response` object with data of type `RegionSubscription`

Return type `Response`

create_swift_password (`create_swift_password_details`, `user_id`, `**kwargs`)

`CreateSwiftPassword` Creates a new Swift password for the specified user. For information about what Swift passwords are for, see [Managing User Credentials](#).

You must specify a *description* for the Swift password (although it can be an empty string). It does not have to be unique, and you can change it anytime with `update_swift_password()`.

Every user has permission to create a Swift password for *their own user ID*. An administrator in your organization does not need to write a policy to give users this ability. To compare, administrators who have permission to the tenancy can use this operation to create a Swift password for any user, including themselves.

Parameters

- **create_swift_password_details** (`CreateSwiftPasswordDetails`) – (required) Request object for creating a new swift password.
- **user_id** (`str`) – (required) The OCID of the user.
- **opc_retry_token** (`str`) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (e.g., if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A `Response` object with data of type `SwiftPassword`

Return type *Response*

create_user (*create_user_details*, ***kwargs*)

CreateUser Creates a new user in your tenancy. For conceptual information about users, your tenancy, and other IAM Service components, see [Overview of the IAM Service](#).

You must specify your tenancy's OCID as the compartment ID in the request object (remember that the tenancy is simply the root compartment). Notice that IAM resources (users, groups, compartments, and some policies) reside within the tenancy itself, unlike cloud resources such as compute instances, which typically reside within compartments inside the tenancy. For information about OCIDs, see [Resource Identifiers](#).

You must also specify a *name* for the user, which must be unique across all users in your tenancy and cannot be changed. Allowed characters: No spaces. Only letters, numerals, hyphens, periods, underscores, +, and @. If you specify a name that's already in use, you'll get a 409 error. This name will be the user's login to the Console. You might want to pick a name that your company's own identity system (e.g., Active Directory, LDAP, etc.) already uses. If you delete a user and then create a new user with the same name, they'll be considered different users because they have different OCIDs.

You must also specify a *description* for the user (although it can be an empty string). It does not have to be unique, and you can change it anytime with `update_user()`. You can use the field to provide the user's full name, a description, a nickname, or other information to generally identify the user.

After you send your request, the new object's *lifecycleState* will temporarily be CREATING. Before using the object, first make sure its *lifecycleState* has changed to ACTIVE.

A new user has no permissions until you place the user in one or more groups (see `add_user_to_group()`). If the user needs to access the Console, you need to provide the user a password (see `create_or_reset_ui_password()`). If the user needs to access the Oracle Bare Metal Cloud Services REST API, you need to upload a public API signing key for that user (see [Required Keys and OCIDs](#) and also `upload_api_key()`).

Important: Make sure to inform the new user which compartment(s) they have access to.

Parameters

- **create_user_details** (*CreateUserDetails*) – (required) Request object for creating a new user.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (e.g., if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *User*

Return type *Response*

delete_api_key (*user_id*, *fingerprint*, ***kwargs*)

DeleteApiKey Deletes the specified API signing key for the specified user.

Every user has permission to use this operation to delete a key for *their own user ID*. An administrator in your organization does not need to write a policy to give users this ability. To compare, administrators who have permission to the tenancy can use this operation to delete a key for any user, including themselves.

Parameters

- **user_id** (*str*) – (required) The OCID of the user.
- **fingerprint** (*str*) – (required) The key's fingerprint.

- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type None

Return type *Response*

delete_customer_secret_key (*user_id*, *customer_secret_key_id*, ***kwargs*)

DeleteCustomerSecretKey Deletes the specified secret key for the specified user.

Parameters

- **user_id** (*str*) – (required) The OCID of the user.
- **customer_secret_key_id** (*str*) – (required) The OCID of the secret key.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type None

Return type *Response*

delete_group (*group_id*, ***kwargs*)

DeleteGroup Deletes the specified group. The group must be empty.

Parameters

- **group_id** (*str*) – (required) The OCID of the group.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type None

Return type *Response*

delete_identity_provider (*identity_provider_id*, ***kwargs*)

DeleteIdentityProvider Deletes the specified identity provider. The identity provider must not have any group mappings (see *IdpGroupMapping*).

Parameters

- **identity_provider_id** (*str*) – (required) The OCID of the identity provider.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type None

Return type *Response*

delete_idp_group_mapping (*identity_provider_id*, *mapping_id*, ***kwargs*)

DeleteIdpGroupMapping Deletes the specified group mapping.

Parameters

- **identity_provider_id** (*str*) – (required) The OCID of the identity provider.

- **mapping_id** (*str*) – (required) The OCID of the group mapping.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type None

Return type *Response*

delete_policy (*policy_id*, ***kwargs*)

DeletePolicy Deletes the specified policy. The deletion takes effect typically within 10 seconds.

Parameters

- **policy_id** (*str*) – (required) The OCID of the policy.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type None

Return type *Response*

delete_swift_password (*user_id*, *swift_password_id*, ***kwargs*)

DeleteSwiftPassword Deletes the specified Swift password for the specified user.

Parameters

- **user_id** (*str*) – (required) The OCID of the user.
- **swift_password_id** (*str*) – (required) The OCID of the Swift password.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type None

Return type *Response*

delete_user (*user_id*, ***kwargs*)

DeleteUser Deletes the specified user. The user must not be in any groups.

Parameters

- **user_id** (*str*) – (required) The OCID of the user.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type None

Return type *Response*

get_compartment (*compartment_id*, ***kwargs*)

GetCompartment Gets the specified compartment’s information.

This operation does not return a list of all the resources inside the compartment. There is no single API operation that does that. Compartments can contain multiple types of resources (instances, block storage

volumes, etc.). To find out what's in a compartment, you must call the "List" operation for each resource type and specify the compartment's OCID as a query parameter in the request. For example, call the `list_instances()` operation in the Cloud Compute Service or the `list_volumes()` operation in Cloud Block Storage.

Parameters `compartment_id` (*str*) – (required) The OCID of the compartment.

Returns A *Response* object with data of type *Compartment*

Return type *Response*

get_group (*group_id*, ***kwargs*)

GetGroup Gets the specified group's information.

This operation does not return a list of all the users in the group. To do that, use `list_user_group_memberships()` and provide the group's OCID as a query parameter in the request.

Parameters `group_id` (*str*) – (required) The OCID of the group.

Returns A *Response* object with data of type *Group*

Return type *Response*

get_identity_provider (*identity_provider_id*, ***kwargs*)

GetIdentityProvider Gets the specified identity provider's information.

Parameters `identity_provider_id` (*str*) – (required) The OCID of the identity provider.

Returns A *Response* object with data of type *IdentityProvider*

Return type *Response*

get_idp_group_mapping (*identity_provider_id*, *mapping_id*, ***kwargs*)

GetIdpGroupMapping Gets the specified group mapping.

Parameters

- `identity_provider_id` (*str*) – (required) The OCID of the identity provider.
- `mapping_id` (*str*) – (required) The OCID of the group mapping.

Returns A *Response* object with data of type *IdpGroupMapping*

Return type *Response*

get_policy (*policy_id*, ***kwargs*)

GetPolicy Gets the specified policy's information.

Parameters `policy_id` (*str*) – (required) The OCID of the policy.

Returns A *Response* object with data of type *Policy*

Return type *Response*

get_tenancy (*tenancy_id*, ***kwargs*)

GetTenancy Get the specified tenancy's information.

Parameters `tenancy_id` (*str*) – (required) The OCID of the tenancy.

Returns A *Response* object with data of type *Tenancy*

Return type *Response*

get_user (*user_id*, ***kwargs*)

GetUser Gets the specified user's information.

Parameters `user_id` (*str*) – (required) The OCID of the user.

Returns A *Response* object with data of type *User*

Return type *Response*

get_user_group_membership (*user_group_membership_id*, ***kwargs*)

GetUserGroupMembership Gets the specified UserGroupMembership’s information.

Parameters `user_group_membership_id` (*str*) – (required) The OCID of the user-GroupMembership.

Returns A *Response* object with data of type *UserGroupMembership*

Return type *Response*

list_api_keys (*user_id*, ***kwargs*)

ListApiKeys Lists the API signing keys for the specified user. A user can have a maximum of three keys.

Every user has permission to use this API call for *their own user ID*. An administrator in your organization does not need to write a policy to give users this ability.

Parameters `user_id` (*str*) – (required) The OCID of the user.

Returns A *Response* object with data of type list of *ApiKey*

Return type *Response*

list_availability_domains (*compartment_id*, ***kwargs*)

ListAvailabilityDomains Lists the Availability Domains in your tenancy. Specify the OCID of either the tenancy or another of your compartments as the value for the compartment ID (remember that the tenancy is simply the root compartment). See [Where to Get the Tenancy’s OCID and User’s OCID](#).

Parameters `compartment_id` (*str*) – (required) The OCID of the compartment (remember that the tenancy is simply the root compartment).

Returns A *Response* object with data of type list of *AvailabilityDomain*

Return type *Response*

list_compartments (*compartment_id*, ***kwargs*)

ListCompartments Lists the compartments in your tenancy. You must specify your tenancy’s OCID as the value for the compartment ID (remember that the tenancy is simply the root compartment). See [Where to Get the Tenancy’s OCID and User’s OCID](#).

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment (remember that the tenancy is simply the root compartment).
- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Returns A *Response* object with data of type list of *Compartment*

Return type *Response*

list_customer_secret_keys (*user_id*, ***kwargs*)

ListCustomerSecretKeys Lists the secret keys for the specified user. The returned object contains the secret key’s OCID, but not the secret key itself. The actual secret key is returned only upon creation.

Parameters `user_id` (*str*) – (required) The OCID of the user.

Returns A *Response* object with data of type list of *CustomerSecretKeySummary*

Return type *Response*

list_groups (*compartment_id*, ***kwargs*)

ListGroups Lists the groups in your tenancy. You must specify your tenancy's OCID as the value for the compartment ID (remember that the tenancy is simply the root compartment). See [Where to Get the Tenancy's OCID and User's OCID](#).

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment (remember that the tenancy is simply the root compartment).
- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Returns A *Response* object with data of type list of *Group*

Return type *Response*

list_identity_providers (*protocol*, *compartment_id*, ***kwargs*)

ListIdentityProviders Lists all the identity providers in your tenancy. You must specify the identity provider type (e.g., *SAML2* for identity providers using the *SAML2.0* protocol). You must specify your tenancy's OCID as the value for the compartment ID (remember that the tenancy is simply the root compartment). See [Where to Get the Tenancy's OCID and User's OCID](#).

Parameters

- **protocol** (*str*) – (required) The protocol used for federation.
- **compartment_id** (*str*) – (required) The OCID of the compartment (remember that the tenancy is simply the root compartment).
- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Returns A *Response* object with data of type list of *IdentityProvider*

Return type *Response*

list_idp_group_mappings (*identity_provider_id*, ***kwargs*)

ListIdpGroupMappings Lists the group mappings for the specified identity provider.

Parameters

- **identity_provider_id** (*str*) – (required) The OCID of the identity provider.
- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Returns A *Response* object with data of type list of *IdpGroupMapping*

Return type *Response*

list_policies (*compartment_id*, ***kwargs*)

ListPolicies Lists the policies in the specified compartment (either the tenancy or another of your compartments). See [Where to Get the Tenancy’s OCID and User’s OCID](#).

To determine which policies apply to a particular group or compartment, you must view the individual statements inside all your policies. There isn’t a way to automatically obtain that information via the API.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment (remember that the tenancy is simply the root compartment).
- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Returns A *Response* object with data of type list of *Policy*

Return type *Response*

list_region_subscriptions (*tenancy_id*, ***kwargs*)

ListRegionSubscriptions Lists the region subscriptions for the specified tenancy.

Parameters **tenancy_id** (*str*) – (required) The OCID of the tenancy.

Returns A *Response* object with data of type list of *RegionSubscription*

Return type *Response*

list_regions (***kwargs*)

ListRegions Lists all the regions offered by Oracle Bare Metal Cloud Services.

Returns A *Response* object with data of type list of *Region*

Return type *Response*

list_swift_passwords (*user_id*, ***kwargs*)

ListSwiftPasswords Lists the Swift passwords for the specified user. The returned object contains the password’s OCID, but not the password itself. The actual password is returned only upon creation.

Parameters **user_id** (*str*) – (required) The OCID of the user.

Returns A *Response* object with data of type list of *SwiftPassword*

Return type *Response*

list_user_group_memberships (*compartment_id*, ***kwargs*)

ListUserGroupMemberships Lists the *UserGroupMembership* objects in your tenancy. You must specify your tenancy’s OCID as the value for the compartment ID (see [Where to Get the Tenancy’s OCID and User’s OCID](#)). You must also then filter the list in one of these ways:

- You can limit the results to just the memberships for a given user by specifying a *userId*.
- Similarly, you can limit the results to just the memberships for a given group by specifying a *groupId*.
- You can set both the *userId* and *groupId* to determine if the specified user is in the specified group.

If the answer is no, the response is an empty list.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment (remember that the tenancy is simply the root compartment).
- **user_id** (*str*) – (optional) The OCID of the user.

- **group_id** (*str*) – (optional) The OCID of the group.
- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Returns A *Response* object with data of type list of *UserGroupMembership*

Return type *Response*

list_users (*compartment_id*, ***kwargs*)

ListUsers Lists the users in your tenancy. You must specify your tenancy’s OCID as the value for the compartment ID (remember that the tenancy is simply the root compartment). See [Where to Get the Tenancy’s OCID and User’s OCID](#).

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment (remember that the tenancy is simply the root compartment).
- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Returns A *Response* object with data of type list of *User*

Return type *Response*

remove_user_from_group (*user_group_membership_id*, ***kwargs*)

RemoveUserFromGroup Removes a user from a group by deleting the corresponding *UserGroupMembership*.

Parameters

- **user_group_membership_id** (*str*) – (required) The OCID of the userGroupMembership.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type None

Return type *Response*

update_compartment (*compartment_id*, *update_compartment_details*, ***kwargs*)

UpdateCompartment Updates the specified compartment’s description.

Parameters

- **compartment_id** (*str*) – (required) The OCID of the compartment.
- **update_compartment_details** (*UpdateCompartmentDetails*) – (required) Request object for updating a compartment.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *Compartment*

Return type *Response*

update_customer_secret_key (*user_id*, *customer_secret_key_id*, *update_customer_secret_key_details*, ***kwargs*)

UpdateCustomerSecretKey Updates the specified secret key's description.

Parameters

- **user_id** (*str*) – (required) The OCID of the user.
- **customer_secret_key_id** (*str*) – (required) The OCID of the secret key.
- **update_customer_secret_key_details** (*UpdateCustomerSecretKeyDetails*) – (required) Request object for updating a secret key.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type *CustomerSecretKeySummary*

Return type *Response*

update_group (*group_id*, *update_group_details*, ***kwargs*)

UpdateGroup Updates the specified group.

Parameters

- **group_id** (*str*) – (required) The OCID of the group.
- **update_group_details** (*UpdateGroupDetails*) – (required) Request object for updating a group.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type *Group*

Return type *Response*

update_identity_provider (*identity_provider_id*, *update_identity_provider_details*, ***kwargs*)

UpdateIdentityProvider Updates the specified identity provider.

Parameters

- **identity_provider_id** (*str*) – (required) The OCID of the identity provider.
- **update_identity_provider_details** (*UpdateIdentityProviderDetails*) – (required) Request object for updating a identity provider.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource's current etag value.

Returns A *Response* object with data of type *IdentityProvider*

Return type *Response*

update_idp_group_mapping (*identity_provider_id*, *mapping_id*, *update_idp_group_mapping_details*, ****kwargs**)
UpdateIdpGroupMapping Updates the specified group mapping.

Parameters

- **identity_provider_id** (*str*) – (required) The OCID of the identity provider.
- **mapping_id** (*str*) – (required) The OCID of the group mapping.
- **update_idp_group_mapping_details** (*UpdateIdpGroupMappingDetails*) – (required) Request object for updating an identity provider group mapping
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *IdpGroupMapping*

Return type *Response*

update_policy (*policy_id*, *update_policy_details*, ****kwargs**)
UpdatePolicy Updates the specified policy. You can update the description or the policy statements themselves.

Policy changes take effect typically within 10 seconds.

Parameters

- **policy_id** (*str*) – (required) The OCID of the policy.
- **update_policy_details** (*UpdatePolicyDetails*) – (required) Request object for updating a policy.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *Policy*

Return type *Response*

update_swift_password (*user_id*, *swift_password_id*, *update_swift_password_details*, ****kwargs**)
UpdateSwiftPassword Updates the specified Swift password’s description.

Parameters

- **user_id** (*str*) – (required) The OCID of the user.
- **swift_password_id** (*str*) – (required) The OCID of the Swift password.
- **update_swift_password_details** (*UpdateSwiftPasswordDetails*) – (required) Request object for updating a Swift password.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *SwiftPassword*

Return type *Response*

update_user (*user_id*, *update_user_details*, ***kwargs*)

UpdateUser Updates the description of the specified user.

Parameters

- **user_id** (*str*) – (required) The OCID of the user.
- **update_user_details** (*UpdateUserDetails*) – (required) Request object for updating a user.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *User*

Return type *Response*

update_user_state (*user_id*, *update_state_details*, ***kwargs*)

UpdateUserState Updates the state of the specified user.

Parameters

- **user_id** (*str*) – (required) The OCID of the user.
- **update_state_details** (*UpdateStateDetails*) – (required) Request object for updating a user state.
- **if_match** (*str*) – (optional) For optimistic concurrency control. In the PUT or DELETE call for a resource, set the *if-match* parameter to the value of the etag from a previous GET or POST response for that resource. The resource will be updated or deleted only if the etag you provide matches the resource’s current etag value.

Returns A *Response* object with data of type *User*

Return type *Response*

upload_api_key (*user_id*, *create_api_key_details*, ***kwargs*)

UploadApiKey Uploads an API signing key for the specified user.

Every user has permission to use this operation to upload a key for *their own user ID*. An administrator in your organization does not need to write a policy to give users this ability. To compare, administrators who have permission to the tenancy can use this operation to upload a key for any user, including themselves.

Important: Even though you have permission to upload an API key, you might not yet have permission to do much else. If you try calling an operation unrelated to your own credential management (e.g., *ListUsers*, *LaunchInstance*) and receive an “unauthorized” error, check with an administrator to confirm which IAM Service group(s) you’re in and what access you have. Also confirm you’re working in the correct compartment.

After you send your request, the new object’s *lifecycleState* will temporarily be CREATING. Before using the object, first make sure its *lifecycleState* has changed to ACTIVE.

Parameters

- **user_id** (*str*) – (required) The OCID of the user.
- **create_api_key_details** (*CreateApiKeyDetails*) – (required) Request object for uploading an API key for a user.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due

to conflicting operations (e.g., if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type *ApiKey*

Return type *Response*

Models

class `oci.identity.models.AddUserToGroupDetails`

group_id

Gets the `group_id` of this `AddUserToGroupDetails`. The OCID of the group.

Returns The `group_id` of this `AddUserToGroupDetails`.

Return type `str`

user_id

Gets the `user_id` of this `AddUserToGroupDetails`. The OCID of the user.

Returns The `user_id` of this `AddUserToGroupDetails`.

Return type `str`

class `oci.identity.models.ApiKey`

fingerprint

Gets the fingerprint of this `ApiKey`. The key's fingerprint (e.g., 12:34:56:78:90:ab:cd:ef:12:34:56:78:90:ab:cd:ef).

Returns The fingerprint of this `ApiKey`.

Return type `str`

inactive_status

Gets the `inactive_status` of this `ApiKey`. The detailed status of `INACTIVE` lifecycleState.

Returns The `inactive_status` of this `ApiKey`.

Return type `int`

key_id

Gets the `key_id` of this `ApiKey`. An Oracle-assigned identifier for the key, in this format: TENANCY_OCID/USER_OCID/KEY_FINGERPRINT.

Returns The `key_id` of this `ApiKey`.

Return type `str`

key_value

Gets the `key_value` of this `ApiKey`. The key's value.

Returns The `key_value` of this `ApiKey`.

Return type `str`

lifecycle_state

Gets the `lifecycle_state` of this `ApiKey`. The API key's current state. After creating an `ApiKey` object, make sure its `lifecycleState` changes from `CREATING` to `ACTIVE` before using it.

Allowed values for this property are: “CREATING”, “ACTIVE”, “INACTIVE”, “DELETING”, “DELETED”, ‘UNKNOWN_ENUM_VALUE’. Any unrecognized values returned by a service will be mapped to ‘UNKNOWN_ENUM_VALUE’.

Returns The lifecycle_state of this ApiKey.

Return type str

time_created

Gets the time_created of this ApiKey. Date and time the *ApiKey* object was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The time_created of this ApiKey.

Return type datetime

user_id

Gets the user_id of this ApiKey. The OCID of the user the key belongs to.

Returns The user_id of this ApiKey.

Return type str

class oci.identity.models.**AvailabilityDomain**

compartment_id

Gets the compartment_id of this AvailabilityDomain. The OCID of the tenancy.

Returns The compartment_id of this AvailabilityDomain.

Return type str

name

Gets the name of this AvailabilityDomain. The name of the Availability Domain.

Returns The name of this AvailabilityDomain.

Return type str

class oci.identity.models.**Compartment**

compartment_id

Gets the compartment_id of this Compartment. The OCID of the tenancy containing the compartment.

Returns The compartment_id of this Compartment.

Return type str

description

Gets the description of this Compartment. The description you assign to the compartment. Does not have to be unique, and it's changeable.

Returns The description of this Compartment.

Return type str

id

Gets the id of this Compartment. The OCID of the compartment.

Returns The id of this Compartment.

Return type str

inactive_status

Gets the `inactive_status` of this Compartment. The detailed status of INACTIVE lifecycleState.

Returns The `inactive_status` of this Compartment.

Return type int

lifecycle_state

Gets the `lifecycle_state` of this Compartment. The compartment's current state. After creating a compartment, make sure its `lifecycleState` changes from CREATING to ACTIVE before using it.

Allowed values for this property are: "CREATING", "ACTIVE", "INACTIVE", "DELETING", "DELETED", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The `lifecycle_state` of this Compartment.

Return type str

name

Gets the name of this Compartment. The name you assign to the compartment during creation. The name must be unique across all compartments in the tenancy and cannot be changed.

Returns The name of this Compartment.

Return type str

time_created

Gets the `time_created` of this Compartment. Date and time the compartment was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The `time_created` of this Compartment.

Return type datetime

class `oci.identity.models.CreateApiKeyDetails`**key**

Gets the key of this `CreateApiKeyDetails`. The public key. Must be an RSA key in PEM format.

Returns The key of this `CreateApiKeyDetails`.

Return type str

class `oci.identity.models.CreateCompartmentDetails`**compartment_id**

Gets the `compartment_id` of this `CreateCompartmentDetails`. The OCID of the tenancy containing the compartment.

Returns The `compartment_id` of this `CreateCompartmentDetails`.

Return type str

description

Gets the description of this `CreateCompartmentDetails`. The description you assign to the compartment during creation. Does not have to be unique, and it's changeable.

Returns The description of this `CreateCompartmentDetails`.

Return type str

name

Gets the name of this CreateCompartmentDetails. The name you assign to the compartment during creation. The name must be unique across all compartments in the tenancy.

Returns The name of this CreateCompartmentDetails.

Return type str

class `oci.identity.models.CreateCustomerSecretKeyDetails`

display_name

Gets the display_name of this CreateCustomerSecretKeyDetails. The name you assign to the secret key during creation. Does not have to be unique, and it's changeable.

Returns The display_name of this CreateCustomerSecretKeyDetails.

Return type str

class `oci.identity.models.CreateGroupDetails`

compartment_id

Gets the compartment_id of this CreateGroupDetails. The OCID of the tenancy containing the group.

Returns The compartment_id of this CreateGroupDetails.

Return type str

description

Gets the description of this CreateGroupDetails. The description you assign to the group during creation. Does not have to be unique, and it's changeable.

Returns The description of this CreateGroupDetails.

Return type str

name

Gets the name of this CreateGroupDetails. The name you assign to the group during creation. The name must be unique across all groups in the tenancy and cannot be changed.

Returns The name of this CreateGroupDetails.

Return type str

class `oci.identity.models.CreateIdentityProviderDetails`

compartment_id

Gets the compartment_id of this CreateIdentityProviderDetails. The OCID of your tenancy.

Returns The compartment_id of this CreateIdentityProviderDetails.

Return type str

description

Gets the description of this CreateIdentityProviderDetails. The description you assign to the *IdentityProvider* during creation. Does not have to be unique, and it's changeable.

Returns The description of this CreateIdentityProviderDetails.

Return type str

static get_subtype (*object_dictionary*)

Given the hash representation of a subtype of this class, use the info in the hash to return the class of the subtype.

name

Gets the name of this `CreateIdentityProviderDetails`. The name you assign to the *IdentityProvider* during creation. The name must be unique across all *IdentityProvider* objects in the tenancy and cannot be changed.

Returns The name of this `CreateIdentityProviderDetails`.

Return type str

product_type

Gets the `product_type` of this `CreateIdentityProviderDetails`. The identity provider service or product. Supported identity providers are Oracle Identity Cloud Service (IDCS) and Microsoft Active Directory Federation Services (ADFS).

Example: *IDCS*

Allowed values for this property are: “IDCS”, “ADFS”

Returns The `product_type` of this `CreateIdentityProviderDetails`.

Return type str

protocol

Gets the protocol of this `CreateIdentityProviderDetails`. The protocol used for federation.

Example: *SAML2*

Allowed values for this property are: “SAML2”

Returns The protocol of this `CreateIdentityProviderDetails`.

Return type str

class `oci.identity.models.CreateIdpGroupMappingDetails`**group_id**

Gets the `group_id` of this `CreateIdpGroupMappingDetails`. The OCID of the IAM Service *Group* you want to map to the IdP group.

Returns The `group_id` of this `CreateIdpGroupMappingDetails`.

Return type str

idp_group_name

Gets the `idp_group_name` of this `CreateIdpGroupMappingDetails`. The name of the IdP group you want to map.

Returns The `idp_group_name` of this `CreateIdpGroupMappingDetails`.

Return type str

class `oci.identity.models.CreatePolicyDetails`**compartment_id**

Gets the `compartment_id` of this `CreatePolicyDetails`. The OCID of the compartment containing the policy (either the tenancy or another compartment).

Returns The `compartment_id` of this `CreatePolicyDetails`.

Return type str

description

Gets the description of this `CreatePolicyDetails`. The description you assign to the policy during creation. Does not have to be unique, and it's changeable.

Returns The description of this CreatePolicyDetails.

Return type str

name

Gets the name of this CreatePolicyDetails. The name you assign to the policy during creation. The name must be unique across all policies in the tenancy and cannot be changed.

Returns The name of this CreatePolicyDetails.

Return type str

statements

Gets the statements of this CreatePolicyDetails. An array of policy statements written in the policy language. See [How Policies Work](#) and [Common Policies](#).

Returns The statements of this CreatePolicyDetails.

Return type list[str]

version_date

Gets the version_date of this CreatePolicyDetails. The version of the policy. If null or set to an empty string, when a request comes in for authorization, the policy will be evaluated according to the current behavior of the services at that moment. If set to a particular date (YYYY-MM-DD), the policy will be evaluated according to the behavior of the services on that date.

Returns The version_date of this CreatePolicyDetails.

Return type datetime

class oci.identity.models.**CreateRegionSubscriptionDetails**

region_key

Gets the region_key of this CreateRegionSubscriptionDetails. The regions's key.

Allowed values are: - *PHX* - *IAD* - *FRA*

Example: *PHX*

Returns The region_key of this CreateRegionSubscriptionDetails.

Return type str

class oci.identity.models.**CreateSaml2IdentityProviderDetails**

metadata

Gets the metadata of this CreateSaml2IdentityProviderDetails. The XML that contains the information required for federating.

Returns The metadata of this CreateSaml2IdentityProviderDetails.

Return type str

metadata_url

Gets the metadata_url of this CreateSaml2IdentityProviderDetails. The URL for retrieving the identity provider's metadata, which contains information required for federating.

Returns The metadata_url of this CreateSaml2IdentityProviderDetails.

Return type str

class oci.identity.models.**CreateSwiftPasswordDetails**

description

Gets the description of this CreateSwiftPasswordDetails. The description you assign to the Swift password during creation. Does not have to be unique, and it's changeable.

Returns The description of this CreateSwiftPasswordDetails.

Return type str

class `oci.identity.models.CreateUserDetails`

compartment_id

Gets the compartment_id of this CreateUserDetails. The OCID of the tenancy containing the user.

Returns The compartment_id of this CreateUserDetails.

Return type str

description

Gets the description of this CreateUserDetails. The description you assign to the user during creation. Does not have to be unique, and it's changeable.

Returns The description of this CreateUserDetails.

Return type str

name

Gets the name of this CreateUserDetails. The name you assign to the user during creation. This is the user's login for the Console. The name must be unique across all users in the tenancy and cannot be changed.

Returns The name of this CreateUserDetails.

Return type str

class `oci.identity.models.CustomerSecretKey`

display_name

Gets the display_name of this CustomerSecretKey. The display name you assign to the secret key. Does not have to be unique, and it's changeable.

Returns The display_name of this CustomerSecretKey.

Return type str

id

Gets the id of this CustomerSecretKey. The OCID of the secret key.

Returns The id of this CustomerSecretKey.

Return type str

inactive_status

Gets the inactive_status of this CustomerSecretKey. The detailed status of INACTIVE lifecycleState.

Returns The inactive_status of this CustomerSecretKey.

Return type int

key

Gets the key of this CustomerSecretKey. The secret key.

Returns The key of this CustomerSecretKey.

Return type str

lifecycle_state

Gets the `lifecycle_state` of this `CustomerSecretKey`. The secret key's current state. After creating a secret key, make sure its `lifecycleState` changes from `CREATING` to `ACTIVE` before using it.

Allowed values for this property are: “`CREATING`”, “`ACTIVE`”, “`INACTIVE`”, “`DELETING`”, “`DELETED`”, “`UNKNOWN_ENUM_VALUE`”. Any unrecognized values returned by a service will be mapped to ‘`UNKNOWN_ENUM_VALUE`’.

Returns The `lifecycle_state` of this `CustomerSecretKey`.

Return type str

time_created

Gets the `time_created` of this `CustomerSecretKey`. Date and time the `CustomerSecretKey` object was created, in the format defined by RFC3339.

Example: `2016-08-25T21:10:29.600Z`

Returns The `time_created` of this `CustomerSecretKey`.

Return type datetime

time_expires

Gets the `time_expires` of this `CustomerSecretKey`. Date and time when this password will expire, in the format defined by RFC3339. Null if it never expires.

Example: `2016-08-25T21:10:29.600Z`

Returns The `time_expires` of this `CustomerSecretKey`.

Return type datetime

user_id

Gets the `user_id` of this `CustomerSecretKey`. The OCID of the user the password belongs to.

Returns The `user_id` of this `CustomerSecretKey`.

Return type str

class `oci.identity.models.CustomerSecretKeySummary`

display_name

Gets the `display_name` of this `CustomerSecretKeySummary`. The `displayName` you assign to the secret key. Does not have to be unique, and it's changeable.

Returns The `display_name` of this `CustomerSecretKeySummary`.

Return type str

id

Gets the `id` of this `CustomerSecretKeySummary`. The OCID of the secret key.

Returns The `id` of this `CustomerSecretKeySummary`.

Return type str

inactive_status

Gets the `inactive_status` of this `CustomerSecretKeySummary`. The detailed status of `INACTIVE` lifecycleState.

Returns The `inactive_status` of this `CustomerSecretKeySummary`.

Return type int

lifecycle_state

Gets the `lifecycle_state` of this `CustomerSecretKeySummary`. The secret key's current state. After creating a secret key, make sure its `lifecycleState` changes from `CREATING` to `ACTIVE` before using it.

Allowed values for this property are: “`CREATING`”, “`ACTIVE`”, “`INACTIVE`”, “`DELETING`”, “`DELETED`”, “`UNKNOWN_ENUM_VALUE`”. Any unrecognized values returned by a service will be mapped to ‘`UNKNOWN_ENUM_VALUE`’.

Returns The `lifecycle_state` of this `CustomerSecretKeySummary`.

Return type str

time_created

Gets the `time_created` of this `CustomerSecretKeySummary`. Date and time the `CustomerSecretKey` object was created, in the format defined by RFC3339.

Example: `2016-08-25T21:10:29.600Z`

Returns The `time_created` of this `CustomerSecretKeySummary`.

Return type datetime

time_expires

Gets the `time_expires` of this `CustomerSecretKeySummary`. Date and time when this password will expire, in the format defined by RFC3339. Null if it never expires.

Example: `2016-08-25T21:10:29.600Z`

Returns The `time_expires` of this `CustomerSecretKeySummary`.

Return type datetime

user_id

Gets the `user_id` of this `CustomerSecretKeySummary`. The OCID of the user the password belongs to.

Returns The `user_id` of this `CustomerSecretKeySummary`.

Return type str

class `oci.identity.models.Group`

compartment_id

Gets the `compartment_id` of this `Group`. The OCID of the tenancy containing the group.

Returns The `compartment_id` of this `Group`.

Return type str

description

Gets the description of this `Group`. The description you assign to the group. Does not have to be unique, and it's changeable.

Returns The description of this `Group`.

Return type str

id

Gets the `id` of this `Group`. The OCID of the group.

Returns The `id` of this `Group`.

Return type str

inactive_status

Gets the `inactive_status` of this `Group`. The detailed status of `INACTIVE` lifecycleState.

Returns The `inactive_status` of this Group.

Return type int

lifecycle_state

Gets the `lifecycle_state` of this Group. The group's current state. After creating a group, make sure its `lifecycleState` changes from `CREATING` to `ACTIVE` before using it.

Allowed values for this property are: “`CREATING`”, “`ACTIVE`”, “`INACTIVE`”, “`DELETING`”, “`DELETED`”, “`UNKNOWN_ENUM_VALUE`”. Any unrecognized values returned by a service will be mapped to “`UNKNOWN_ENUM_VALUE`”.

Returns The `lifecycle_state` of this Group.

Return type str

name

Gets the name of this Group. The name you assign to the group during creation. The name must be unique across all groups in the tenancy and cannot be changed.

Returns The name of this Group.

Return type str

time_created

Gets the `time_created` of this Group. Date and time the group was created, in the format defined by RFC3339.

Example: `2016-08-25T21:10:29.600Z`

Returns The `time_created` of this Group.

Return type datetime

class `oci.identity.models.IdentityProvider`

compartment_id

Gets the `compartment_id` of this `IdentityProvider`. The OCID of the tenancy containing the `IdentityProvider`.

Returns The `compartment_id` of this `IdentityProvider`.

Return type str

description

Gets the description of this `IdentityProvider`. The description you assign to the `IdentityProvider` during creation. Does not have to be unique, and it's changeable.

Returns The description of this `IdentityProvider`.

Return type str

static `get_subtype` (*object_dictionary*)

Given the hash representation of a subtype of this class, use the info in the hash to return the class of the subtype.

id

Gets the id of this `IdentityProvider`. The OCID of the `IdentityProvider`.

Returns The id of this `IdentityProvider`.

Return type str

inactive_status

Gets the `inactive_status` of this `IdentityProvider`. The detailed status of `INACTIVE` lifecycleState.

Returns The `inactive_status` of this `IdentityProvider`.

Return type `int`

lifecycle_state

Gets the `lifecycle_state` of this `IdentityProvider`. The current state. After creating an `IdentityProvider`, make sure its `lifecycleState` changes from `CREATING` to `ACTIVE` before using it.

Allowed values for this property are: “`CREATING`”, “`ACTIVE`”, “`INACTIVE`”, “`DELETING`”, “`DELETED`”, “`UNKNOWN_ENUM_VALUE`”. Any unrecognized values returned by a service will be mapped to ‘`UNKNOWN_ENUM_VALUE`’.

Returns The `lifecycle_state` of this `IdentityProvider`.

Return type `str`

name

Gets the name of this `IdentityProvider`. The name you assign to the `IdentityProvider` during creation. The name must be unique across all `IdentityProvider` objects in the tenancy and cannot be changed. This is the name federated users see when choosing which identity provider to use when signing in to the Oracle Bare Metal Cloud Services Console.

Returns The name of this `IdentityProvider`.

Return type `str`

product_type

Gets the `product_type` of this `IdentityProvider`. The identity provider service or product. Supported identity providers are Oracle Identity Cloud Service (IDCS) and Microsoft Active Directory Federation Services (ADFS).

Allowed values are: - `ADFS` - `IDCS`

Example: `IDCS`

Returns The `product_type` of this `IdentityProvider`.

Return type `str`

protocol

Gets the protocol of this `IdentityProvider`. The protocol used for federation. Allowed value: `SAML2`.

Example: `SAML2`

Returns The protocol of this `IdentityProvider`.

Return type `str`

time_created

Gets the `time_created` of this `IdentityProvider`. Date and time the `IdentityProvider` was created, in the format defined by RFC3339.

Example: `2016-08-25T21:10:29.600Z`

Returns The `time_created` of this `IdentityProvider`.

Return type `datetime`

class `oci.identity.models.IdpGroupMapping`

compartment_id

Gets the `compartment_id` of this `IdpGroupMapping`. The OCID of the tenancy containing the `IdentityProvider`.

Returns The `compartment_id` of this `IdpGroupMapping`.

Return type str

group_id

Gets the `group_id` of this `IdpGroupMapping`. The OCID of the IAM Service group that is mapped to the IdP group.

Returns The `group_id` of this `IdpGroupMapping`.

Return type str

id

Gets the `id` of this `IdpGroupMapping`. The OCID of the *IdpGroupMapping*.

Returns The `id` of this `IdpGroupMapping`.

Return type str

idp_group_name

Gets the `idp_group_name` of this `IdpGroupMapping`. The name of the IdP group that is mapped to the IAM Service group.

Returns The `idp_group_name` of this `IdpGroupMapping`.

Return type str

idp_id

Gets the `idp_id` of this `IdpGroupMapping`. The OCID of the *IdentityProvider* this mapping belongs to.

Returns The `idp_id` of this `IdpGroupMapping`.

Return type str

inactive_status

Gets the `inactive_status` of this `IdpGroupMapping`. The detailed status of INACTIVE lifecycleState.

Returns The `inactive_status` of this `IdpGroupMapping`.

Return type int

lifecycle_state

Gets the `lifecycle_state` of this `IdpGroupMapping`. The mapping's current state. After creating a mapping object, make sure its *lifecycleState* changes from CREATING to ACTIVE before using it.

Allowed values for this property are: "CREATING", "ACTIVE", "INACTIVE", "DELETING", "DELETED", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The `lifecycle_state` of this `IdpGroupMapping`.

Return type str

time_created

Gets the `time_created` of this `IdpGroupMapping`. Date and time the mapping was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The `time_created` of this `IdpGroupMapping`.

Return type datetime

class `oci.identity.models.Policy`

compartment_id

Gets the compartment_id of this Policy. The OCID of the compartment containing the policy (either the tenancy or another compartment).

Returns The compartment_id of this Policy.

Return type str

description

Gets the description of this Policy. The description you assign to the policy. Does not have to be unique, and it's changeable.

Returns The description of this Policy.

Return type str

id

Gets the id of this Policy. The OCID of the policy.

Returns The id of this Policy.

Return type str

inactive_status

Gets the inactive_status of this Policy. The detailed status of INACTIVE lifecycleState.

Returns The inactive_status of this Policy.

Return type int

lifecycle_state

Gets the lifecycle_state of this Policy. The policy's current state. After creating a policy, make sure its *lifecycleState* changes from CREATING to ACTIVE before using it.

Allowed values for this property are: "CREATING", "ACTIVE", "INACTIVE", "DELETING", "DELETED", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The lifecycle_state of this Policy.

Return type str

name

Gets the name of this Policy. The name you assign to the policy during creation. The name must be unique across all policies in the tenancy and cannot be changed.

Returns The name of this Policy.

Return type str

statements

Gets the statements of this Policy. An array of one or more policy statements written in the policy language.

Returns The statements of this Policy.

Return type list[str]

time_created

Gets the time_created of this Policy. Date and time the policy was created, in the format defined by RFC3339.

Example: 2016-08-25T21:10:29.600Z

Returns The time_created of this Policy.

Return type datetime

version_date

Gets the `version_date` of this Policy. The version of the policy. If null or set to an empty string, when a request comes in for authorization, the policy will be evaluated according to the current behavior of the services at that moment. If set to a particular date (YYYY-MM-DD), the policy will be evaluated according to the behavior of the services on that date.

Returns The `version_date` of this Policy.

Return type `datetime`

class `oci.identity.models.Region`

key

Gets the `key` of this Region. The key of the region.

Allowed values are: - *PHX* - *IAD* - 'FRA'

Returns The `key` of this Region.

Return type `str`

name

Gets the `name` of this Region. The name of the region.

Allowed values are: - *us-phoenix-1* - *us-ashburn-1* - 'de-frankfurt-1'

Returns The `name` of this Region.

Return type `str`

class `oci.identity.models.RegionSubscription`

is_home_region

Gets the `is_home_region` of this RegionSubscription. Indicates if the region is the home region or not.

Returns The `is_home_region` of this RegionSubscription.

Return type `bool`

region_key

Gets the `region_key` of this RegionSubscription. The region's key.

Allowed values are: - *PHX* - *IAD* - 'FRA'

Returns The `region_key` of this RegionSubscription.

Return type `str`

region_name

Gets the `region_name` of this RegionSubscription. The region's name.

Allowed values are: - *us-phoenix-1* - *us-ashburn-1* - 'de-frankfurt-1'

Returns The `region_name` of this RegionSubscription.

Return type `str`

status

Gets the `status` of this RegionSubscription. The region subscription status.

Allowed values for this property are: "READY", "IN_PROGRESS", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The `status` of this RegionSubscription.

Return type str

class `oci.identity.models.Saml2IdentityProvider`

metadata_url

Gets the `metadata_url` of this `Saml2IdentityProvider`. The URL for retrieving the identity provider's metadata, which contains information required for federating.

Returns The `metadata_url` of this `Saml2IdentityProvider`.

Return type str

redirect_url

Gets the `redirect_url` of this `Saml2IdentityProvider`. The URL to redirect federated users to for authentication with the identity provider.

Returns The `redirect_url` of this `Saml2IdentityProvider`.

Return type str

signing_certificate

Gets the `signing_certificate` of this `Saml2IdentityProvider`. The identity provider's signing certificate used by the IAM Service to validate the SAML2 token.

Returns The `signing_certificate` of this `Saml2IdentityProvider`.

Return type str

class `oci.identity.models.SwiftPassword`

description

Gets the description of this `SwiftPassword`. The description you assign to the Swift password. Does not have to be unique, and it's changeable.

Returns The description of this `SwiftPassword`.

Return type str

expires_on

Gets the `expires_on` of this `SwiftPassword`. Date and time when this password will expire, in the format defined by RFC3339. Null if it never expires.

Example: `2016-08-25T21:10:29.600Z`

Returns The `expires_on` of this `SwiftPassword`.

Return type datetime

id

Gets the `id` of this `SwiftPassword`. The OCID of the Swift password.

Returns The `id` of this `SwiftPassword`.

Return type str

inactive_status

Gets the `inactive_status` of this `SwiftPassword`. The detailed status of `INACTIVE` lifecycleState.

Returns The `inactive_status` of this `SwiftPassword`.

Return type int

lifecycle_state

Gets the `lifecycle_state` of this `SwiftPassword`. The password's current state. After creating a password, make sure its `lifecycleState` changes from `CREATING` to `ACTIVE` before using it.

Allowed values for this property are: “CREATING”, “ACTIVE”, “INACTIVE”, “DELETING”, “DELETED”, ‘UNKNOWN_ENUM_VALUE’. Any unrecognized values returned by a service will be mapped to ‘UNKNOWN_ENUM_VALUE’.

Returns The `lifecycle_state` of this `SwiftPassword`.

Return type str

password

Gets the password of this `SwiftPassword`. The Swift password. The value is available only in the response for `CreateSwiftPassword`, and not for `ListSwiftPasswords` or `UpdateSwiftPassword`.

Returns The password of this `SwiftPassword`.

Return type str

time_created

Gets the `time_created` of this `SwiftPassword`. Date and time the `SwiftPassword` object was created, in the format defined by RFC3339.

Example: `2016-08-25T21:10:29.600Z`

Returns The `time_created` of this `SwiftPassword`.

Return type datetime

user_id

Gets the `user_id` of this `SwiftPassword`. The OCID of the user the password belongs to.

Returns The `user_id` of this `SwiftPassword`.

Return type str

class `oci.identity.models.Tenancy`

description

Gets the description of this `Tenancy`. The description of the tenancy.

Returns The description of this `Tenancy`.

Return type str

home_region_key

Gets the `home_region_key` of this `Tenancy`. The region key for the tenancy's home region.

Allowed values are: - `IAD` - `PHX` - `FRA`

Returns The `home_region_key` of this `Tenancy`.

Return type str

id

Gets the `id` of this `Tenancy`. The OCID of the tenancy.

Returns The `id` of this `Tenancy`.

Return type str

name

Gets the name of this `Tenancy`. The name of the tenancy.

Returns The name of this `Tenancy`.

Return type str

class `oci.identity.models.UIPassword`

inactive_status

Gets the `inactive_status` of this `UIPassword`. The detailed status of `INACTIVE` lifecycleState.

Returns The `inactive_status` of this `UIPassword`.

Return type int

lifecycle_state

Gets the `lifecycle_state` of this `UIPassword`. The password's current state. After creating a password, make sure its `lifecycleState` changes from `CREATING` to `ACTIVE` before using it.

Allowed values for this property are: “`CREATING`”, “`ACTIVE`”, “`INACTIVE`”, “`DELETING`”, “`DELETED`”, “`UNKNOWN_ENUM_VALUE`”. Any unrecognized values returned by a service will be mapped to ‘`UNKNOWN_ENUM_VALUE`’.

Returns The `lifecycle_state` of this `UIPassword`.

Return type str

password

Gets the password of this `UIPassword`. The user's password for the Console.

Returns The password of this `UIPassword`.

Return type str

time_created

Gets the `time_created` of this `UIPassword`. Date and time the password was created, in the format defined by RFC3339.

Example: `2016-08-25T21:10:29.600Z`

Returns The `time_created` of this `UIPassword`.

Return type datetime

user_id

Gets the `user_id` of this `UIPassword`. The OCID of the user.

Returns The `user_id` of this `UIPassword`.

Return type str

class `oci.identity.models.UpdateCompartmentDetails`

description

Gets the description of this `UpdateCompartmentDetails`. The description you assign to the compartment. Does not have to be unique, and it's changeable.

Returns The description of this `UpdateCompartmentDetails`.

Return type str

name

Gets the name of this `UpdateCompartmentDetails`. The new name you assign to the compartment. The name must be unique across all compartments in the tenancy.

Returns The name of this `UpdateCompartmentDetails`.

Return type str

class `oci.identity.models.UpdateCustomerSecretKeyDetails`

display_name

Gets the `display_name` of this `UpdateCustomerSecretKeyDetails`. The description you assign to the secret key. Does not have to be unique, and it's changeable.

Returns The `display_name` of this `UpdateCustomerSecretKeyDetails`.

Return type `str`

class `oci.identity.models.UpdateGroupDetails`

description

Gets the description of this `UpdateGroupDetails`. The description you assign to the group. Does not have to be unique, and it's changeable.

Returns The description of this `UpdateGroupDetails`.

Return type `str`

class `oci.identity.models.UpdateIdentityProviderDetails`

description

Gets the description of this `UpdateIdentityProviderDetails`. The description you assign to the *IdentityProvider*. Does not have to be unique, and it's changeable.

Returns The description of this `UpdateIdentityProviderDetails`.

Return type `str`

static `get_subtype` (*object_dictionary*)

Given the hash representation of a subtype of this class, use the info in the hash to return the class of the subtype.

protocol

Gets the protocol of this `UpdateIdentityProviderDetails`. The protocol used for federation.

Example: *SAML2*

Allowed values for this property are: "SAML2"

Returns The protocol of this `UpdateIdentityProviderDetails`.

Return type `str`

class `oci.identity.models.UpdateIdpGroupMappingDetails`

group_id

Gets the `group_id` of this `UpdateIdpGroupMappingDetails`. The OCID of the group.

Returns The `group_id` of this `UpdateIdpGroupMappingDetails`.

Return type `str`

idp_group_name

Gets the `idp_group_name` of this `UpdateIdpGroupMappingDetails`. The idp group name.

Returns The `idp_group_name` of this `UpdateIdpGroupMappingDetails`.

Return type `str`

class `oci.identity.models.UpdatePolicyDetails`

description

Gets the description of this `UpdatePolicyDetails`. The description you assign to the policy. Does not have to be unique, and it's changeable.

Returns The description of this `UpdatePolicyDetails`.

Return type `str`

statements

Gets the statements of this `UpdatePolicyDetails`. An array of policy statements written in the policy language. See [How Policies Work](#) and [Common Policies](#).

Returns The statements of this `UpdatePolicyDetails`.

Return type `list[str]`

version_date

Gets the `version_date` of this `UpdatePolicyDetails`. The version of the policy. If null or set to an empty string, when a request comes in for authorization, the policy will be evaluated according to the current behavior of the services at that moment. If set to a particular date (YYYY-MM-DD), the policy will be evaluated according to the behavior of the services on that date.

Returns The `version_date` of this `UpdatePolicyDetails`.

Return type `datetime`

class `oci.identity.models.UpdateSaml2IdentityProviderDetails`

metadata

Gets the metadata of this `UpdateSaml2IdentityProviderDetails`. The XML that contains the information required for federating.

Returns The metadata of this `UpdateSaml2IdentityProviderDetails`.

Return type `str`

metadata_url

Gets the `metadata_url` of this `UpdateSaml2IdentityProviderDetails`. The URL for retrieving the identity provider's metadata, which contains information required for federating.

Returns The `metadata_url` of this `UpdateSaml2IdentityProviderDetails`.

Return type `str`

class `oci.identity.models.UpdateStateDetails`

blocked

Gets the `blocked` of this `UpdateStateDetails`. Update state to blocked or unblocked. Only "false" is supported (for changing the state to unblocked).

Returns The `blocked` of this `UpdateStateDetails`.

Return type `bool`

class `oci.identity.models.UpdateSwiftPasswordDetails`

description

Gets the description of this `UpdateSwiftPasswordDetails`. The description you assign to the Swift password. Does not have to be unique, and it's changeable.

Returns The description of this UpdateSwiftPasswordDetails.

Return type str

class `oci.identity.models.UpdateUserDetails`

description

Gets the description of this UpdateUserDetails. The description you assign to the user. Does not have to be unique, and it's changeable.

Returns The description of this UpdateUserDetails.

Return type str

class `oci.identity.models.User`

compartment_id

Gets the compartment_id of this User. The OCID of the tenancy containing the user.

Returns The compartment_id of this User.

Return type str

description

Gets the description of this User. The description you assign to the user. Does not have to be unique, and it's changeable.

Returns The description of this User.

Return type str

id

Gets the id of this User. The OCID of the user.

Returns The id of this User.

Return type str

inactive_status

Gets the inactive_status of this User. Returned only if the user's *lifecycleState* is INACTIVE. A 16-bit value showing the reason why the user is inactive:

- bit 0: SUSPENDED (reserved for future use)
- bit 1: DISABLED (reserved for future use)
- bit 2: BLOCKED (the user has exceeded the maximum number of failed login attempts for the Console)

Returns The inactive_status of this User.

Return type int

lifecycle_state

Gets the lifecycle_state of this User. The user's current state. After creating a user, make sure its *lifecycleState* changes from CREATING to ACTIVE before using it.

Allowed values for this property are: "CREATING", "ACTIVE", "INACTIVE", "DELETING", "DELETED", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The lifecycle_state of this User.

Return type str

name

Gets the name of this User. The name you assign to the user during creation. This is the user's login for the Console. The name must be unique across all users in the tenancy and cannot be changed.

Returns The name of this User.

Return type str

time_created

Gets the time_created of this User. Date and time the user was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The time_created of this User.

Return type datetime

class `oci.identity.models.UserGroupMembership`

compartment_id

Gets the compartment_id of this UserGroupMembership. The OCID of the tenancy containing the user, group, and membership object.

Returns The compartment_id of this UserGroupMembership.

Return type str

group_id

Gets the group_id of this UserGroupMembership. The OCID of the group.

Returns The group_id of this UserGroupMembership.

Return type str

id

Gets the id of this UserGroupMembership. The OCID of the membership.

Returns The id of this UserGroupMembership.

Return type str

inactive_status

Gets the inactive_status of this UserGroupMembership. The detailed status of INACTIVE lifecycleState.

Returns The inactive_status of this UserGroupMembership.

Return type int

lifecycle_state

Gets the lifecycle_state of this UserGroupMembership. The membership's current state. After creating a membership object, make sure its *lifecycleState* changes from CREATING to ACTIVE before using it.

Allowed values for this property are: "CREATING", "ACTIVE", "INACTIVE", "DELETING", "DELETED", "UNKNOWN_ENUM_VALUE". Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The lifecycle_state of this UserGroupMembership.

Return type str

time_created

Gets the time_created of this UserGroupMembership. Date and time the membership was created, in the format defined by RFC3339.

Example: `2016-08-25T21:10:29.600Z`

Returns The `time_created` of this `UserGroupMembership`.

Return type `datetime`

user_id

Gets the `user_id` of this `UserGroupMembership`. The OCID of the user.

Returns The `user_id` of this `UserGroupMembership`.

Return type `str`

Load Balancer

Client

`class oci.load_balancer.load_balancer_client.LoadBalancerClient` (*config*)

create_backend (*create_backend_details*, *load_balancer_id*, *backend_set_name*, ***kwargs*)

CreateBackend Adds a backend server to a backend set.

Parameters

- **create_backend_details** (`CreateBackendDetails`) – (required) The details to add a backend server to a backend set.
- **load_balancer_id** (`str`) – (required) The OCID of the load balancer associated with the backend set and servers.
- **backend_set_name** (`str`) – (required) The name of the backend set to add the backend server to.

Example: `My_backend_set`

- **opc_request_id** (`str`) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.
- **opc_retry_token** (`str`) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (e.g., if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A `Response` object with data of type `None`

Return type `Response`

create_backend_set (*create_backend_set_details*, *load_balancer_id*, ***kwargs*)

CreateBackendSet Adds a backend set to a load balancer.

Parameters

- **create_backend_set_details** (`CreateBackendSetDetails`) – (required) The details for adding a backend set.
- **load_balancer_id** (`str`) – (required) The OCID of the load balancer on which to add a backend set.

- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (e.g., if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type None

Return type *Response*

create_certificate (*create_certificate_details*, *load_balancer_id*, ***kwargs*)

CreateCertificate Creates an asynchronous request to add an SSL certificate.

Parameters

- **create_certificate_details** (*CreateCertificateDetails*) – (required) The details of the certificate to add.
- **load_balancer_id** (*str*) – (required) The **OCID** of the load balancer on which to add the certificate.
- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (e.g., if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type None

Return type *Response*

create_listener (*create_listener_details*, *load_balancer_id*, ***kwargs*)

CreateListener Adds a listener to a load balancer.

Parameters

- **create_listener_details** (*CreateListenerDetails*) – (required) Details to add a listener.
- **load_balancer_id** (*str*) – (required) The **OCID** of the load balancer on which to add a listener.
- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (e.g., if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type None

Return type *Response*

create_load_balancer (*create_load_balancer_details*, ***kwargs*)

CreateLoadBalancer Creates a new load balancer in the specified compartment. For general information about load balancers, see [Overview of the Load Balancing Service](#).

For the purposes of access control, you must provide the OCID of the compartment where you want the load balancer to reside. Notice that the load balancer doesn't have to be in the same compartment as the VCN or backend set. If you're not sure which compartment to use, put the load balancer in the same compartment as the VCN. For information about access control and compartments, see [Overview of the IAM Service](#).

You must specify a display name for the load balancer. It does not have to be unique, and you can change it.

For information about Availability Domains, see [Regions and Availability Domains](#). To get a list of Availability Domains, use the *ListAvailabilityDomains* operation in the Identity and Access Management Service API.

All Oracle Bare Metal Cloud Services resources, including load balancers, get an Oracle-assigned, unique ID called an Oracle Cloud Identifier (OCID). When you create a resource, you can find its OCID in the response. You can also retrieve a resource's OCID by using a List API operation on that resource type, or by viewing the resource in the Console. For more information, see [Resource Identifiers](#).

After you send your request, the new object's state will temporarily be PROVISIONING. Before using the object, first make sure its state has changed to RUNNING.

When you create a load balancer, the system assigns an IP address. To get the IP address, use the *get_load_balancer()* operation.

Parameters

- **create_load_balancer_details** (*CreateLoadBalancerDetails*) – (required) The configuration details for creating a load balancer.
- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (e.g., if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type None

Return type *Response*

delete_backend (*load_balancer_id*, *backend_set_name*, *backend_name*, ***kwargs*)

DeleteBackend Removes a backend server from a given load balancer and backend set.

Parameters

- **load_balancer_id** (*str*) – (required) The OCID of the load balancer associated with the backend set and server.
- **backend_set_name** (*str*) – (required) The name of the backend set associated with the backend server.

Example: *My_backend_set*

- **backend_name** (*str*) – (required) The IP address and port of the backend server to remove.

Example: *1.1.1.7:42*

- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.

Returns A *Response* object with data of type None

Return type *Response*

delete_backend_set (*load_balancer_id*, *backend_set_name*, ***kwargs*)

DeleteBackendSet Deletes the specified backend set. Note that deleting a backend set removes its backend servers from the load balancer.

Before you can delete a backend set, you must remove it from any active listeners.

Parameters

- **load_balancer_id** (*str*) – (required) The **OCID** of the load balancer associated with the backend set.
- **backend_set_name** (*str*) – (required) The name of the backend set to delete.

Example: *My_backend_set*

- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.

Returns A *Response* object with data of type None

Return type *Response*

delete_certificate (*load_balancer_id*, *certificate_name*, ***kwargs*)

DeleteCertificate Deletes an SSL certificate from a load balancer.

Parameters

- **load_balancer_id** (*str*) – (required) The **OCID** of the load balancer associated with the certificate to be deleted.
- **certificate_name** (*str*) – (required) The name of the certificate to delete.

Example: *My_certificate_bundle*

- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.

Returns A *Response* object with data of type None

Return type *Response*

delete_listener (*load_balancer_id*, *listener_name*, ***kwargs*)

DeleteListener Deletes a listener from a load balancer.

Parameters

- **load_balancer_id** (*str*) – (required) The **OCID** of the load balancer associated with the listener to delete.
- **listener_name** (*str*) – (required) The name of the listener to delete.

Example: *My listener*

- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.

Returns A *Response* object with data of type *None*

Return type *Response*

delete_load_balancer (*load_balancer_id*, ***kwargs*)

DeleteLoadBalancer Stops a load balancer and removes it from service.

Parameters

- **load_balancer_id** (*str*) – (required) The **OCID** of the load balancer to delete.
- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.

Returns A *Response* object with data of type *None*

Return type *Response*

get_backend (*load_balancer_id*, *backend_set_name*, *backend_name*, ***kwargs*)

GetBackend Gets the specified backend server's configuration information.

Parameters

- **load_balancer_id** (*str*) – (required) The **OCID** of the load balancer associated with the backend set and server.
- **backend_set_name** (*str*) – (required) The name of the backend set that includes the backend server.

Example: *My_backend_set*
- **backend_name** (*str*) – (required) The IP address and port of the backend server to retrieve.

Example: *1.1.1.7:42*
- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.

Returns A *Response* object with data of type *Backend*

Return type *Response*

get_backend_health (*load_balancer_id*, *backend_set_name*, *backend_name*, ***kwargs*)

BackendHealth Gets the current health status of the specified backend server.

Parameters

- **load_balancer_id** (*str*) – (required) The **OCID** of the load balancer associated with the backend server health status to be retrieved.
- **backend_set_name** (*str*) – (required) The name of the backend set associated with the backend server to retrieve the health status for.

Example: *My_backend_set*
- **backend_name** (*str*) – (required) The IP address and port of the backend server to retrieve the health status for.

Example: *1.1.1.7:42*

- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.

Returns A *Response* object with data of type *BackendHealth*

Return type *Response*

get_backend_set (*load_balancer_id*, *backend_set_name*, ***kwargs*)

GetBackendSet Gets the specified backend set's configuration information.

Parameters

- **load_balancer_id** (*str*) – (required) The **OCID** of the specified load balancer.
- **backend_set_name** (*str*) – (required) The name of the backend set to retrieve.

Example: *My_backend_set*

- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.

Returns A *Response* object with data of type *BackendSet*

Return type *Response*

get_backend_set_health (*load_balancer_id*, *backend_set_name*, ***kwargs*)

BackendSetHealth Gets the health status for the specified backend set.

Parameters

- **load_balancer_id** (*str*) – (required) The **OCID** of the load balancer associated with the backend set health status to be retrieved.
- **backend_set_name** (*str*) – (required) The name of the backend set to retrieve the health status for.

Example: *My_backend_set*

- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.

Returns A *Response* object with data of type *BackendSetHealth*

Return type *Response*

get_health_checker (*load_balancer_id*, *backend_set_name*, ***kwargs*)

GetHealthChecker Gets the health check policy information for a given load balancer and backend set.

Parameters

- **load_balancer_id** (*str*) – (required) The **OCID** of the load balancer associated with the health check policy to be retrieved.
- **backend_set_name** (*str*) – (required) The name of the backend set associated with the health check policy to be retrieved.

Example: *My_backend_set*

- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.

Returns A *Response* object with data of type *HealthChecker*

Return type *Response*

get_load_balancer (*load_balancer_id*, ***kwargs*)

GetLoadBalancer Gets the specified load balancer's configuration information.

Parameters

- **load_balancer_id** (*str*) – (required) The **OCID** of the load balancer to retrieve.
- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.

Returns A *Response* object with data of type *LoadBalancer*

Return type *Response*

get_load_balancer_health (*load_balancer_id*, ***kwargs*)

LoadBalancerHealth Gets the health status for the specified load balancer.

Parameters

- **load_balancer_id** (*str*) – (required) The **OCID** of the load balancer to return health status for.
- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.

Returns A *Response* object with data of type *LoadBalancerHealth*

Return type *Response*

get_work_request (*work_request_id*, ***kwargs*)

GetWorkRequest Gets the details of a work request.

Parameters

- **work_request_id** (*str*) – (required) The **OCID** of the work request to retrieve.
- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.

Returns A *Response* object with data of type *WorkRequest*

Return type *Response*

list_backend_sets (*load_balancer_id*, ***kwargs*)

ListBackendSets Lists all backend sets associated with a given load balancer.

Parameters

- **load_balancer_id** (*str*) – (required) The **OCID** of the load balancer associated with the backend sets to retrieve.
- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.

Returns A *Response* object with data of type list of *BackendSet*

Return type *Response*

list_backend_servers (*load_balancer_id*, *backend_set_name*, ***kwargs*)

ListBackendServers Lists the backend servers for a given load balancer and backend set.

Parameters

- **load_balancer_id** (*str*) – (required) The **OCID** of the load balancer associated with the backend set and servers.
- **backend_set_name** (*str*) – (required) The name of the backend set associated with the backend servers.

Example: *My_backend_set*

- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.

Returns A *Response* object with data of type list of *Backend*

Return type *Response*

list_certificates (*load_balancer_id*, ***kwargs*)

ListCertificates Lists all SSL certificates associated with a given load balancer.

Parameters

- **load_balancer_id** (*str*) – (required) The **OCID** of the load balancer associated with the certificates to be listed.
- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.

Returns A *Response* object with data of type list of *Certificate*

Return type *Response*

list_load_balancer_healths (*compartment_id*, ***kwargs*)

ListLoadBalancerHealths Lists the summary health statuses for all load balancers in the specified compartment.

Parameters

- **compartment_id** (*str*) – (required) The **OCID** of the compartment containing the load balancers to return health status information for.
- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: *500*

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Example: *3*

Returns A *Response* object with data of type list of *LoadBalancerHealthSummary*

Return type *Response*

list_load_balancers (*compartment_id*, ***kwargs*)

ListLoadBalancers Lists all load balancers in the specified compartment.

Parameters

- **compartment_id** (*str*) – (required) The **OCID** of the compartment containing the load balancers to list.
- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: *500*

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Example: *3*

- **detail** (*str*) – (optional) The level of detail to return for each result. Can be *full* or *simple*.

Example: *full*

Returns A *Response* object with data of type list of *LoadBalancer*

Return type *Response*

list_policies (*compartment_id*, ***kwargs*)

ListPolicies Lists the available load balancer policies.

Parameters

- **compartment_id** (*str*) – (required) The **OCID** of the compartment containing the load balancer policies to list.
- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.
- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: *500*

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Example: *3*

Returns A *Response* object with data of type list of *LoadBalancerPolicy*

Return type *Response*

list_protocols (*compartment_id*, ***kwargs*)

ListProtocols Lists all supported traffic protocols.

Parameters

- **compartment_id** (*str*) – (required) The **OCID** of the compartment containing the load balancer protocols to list.
- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.

- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: 500

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Example: 3

Returns A *Response* object with data of type list of *LoadBalancerProtocol*

Return type *Response*

list_shapes (*compartment_id*, ***kwargs*)

ListShapes Lists the valid load balancer shapes.

Parameters

- **compartment_id** (*str*) – (required) The **OCID** of the compartment containing the load balancer shapes to list.

- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.

- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: 500

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Example: 3

Returns A *Response* object with data of type list of *LoadBalancerShape*

Return type *Response*

list_work_requests (*load_balancer_id*, ***kwargs*)

ListWorkRequests Lists the work requests for a given load balancer.

Parameters

- **load_balancer_id** (*str*) – (required) The **OCID** of the load balancer associated with the work requests to retrieve.

- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.

- **limit** (*int*) – (optional) The maximum number of items to return in a paginated “List” call.

Example: 500

- **page** (*str*) – (optional) The value of the *opc-next-page* response header from the previous “List” call.

Example: 3

Returns A *Response* object with data of type list of *WorkRequest*

Return type *Response*

update_backend (*update_backend_details*, *load_balancer_id*, *backend_set_name*, *backend_name*, ***kwargs*)

UpdateBackend Updates the configuration of a backend server within the specified backend set.

Parameters

- **update_backend_details** (*UpdateBackendDetails*) – (required) Details for updating a backend server.
- **load_balancer_id** (*str*) – (required) The OCID of the load balancer associated with the backend set and server.
- **backend_set_name** (*str*) – (required) The name of the backend set associated with the backend server.

Example: *My_backend_set*

- **backend_name** (*str*) – (required) The IP address and port of the backend server to update.

Example: *1.1.1.7:42*

- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.

- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (e.g., if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type None

Return type *Response*

update_backend_set (*update_backend_set_details*, *load_balancer_id*, *backend_set_name*, ***kwargs*)

UpdateBackendSet Updates a backend set.

Parameters

- **update_backend_set_details** (*UpdateBackendSetDetails*) – (required) The details to update a backend set.
- **load_balancer_id** (*str*) – (required) The OCID of the load balancer associated with the backend set.
- **backend_set_name** (*str*) – (required) The name of the backend set to update.

Example: *My_backend_set*

- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.

- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (e.g., if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type None

Return type *Response*

update_health_checker (*health_checker*, *load_balancer_id*, *backend_set_name*, ***kwargs*)

UpdateHealthChecker Updates the health check policy for a given load balancer and backend set.

Parameters

- **health_checker** (*UpdateHealthCheckerDetails*) – (required) The health check policy configuration details.
- **load_balancer_id** (*str*) – (required) The OCID of the load balancer associated with the health check policy to be updated.
- **backend_set_name** (*str*) – (required) The name of the backend set associated with the health check policy to be retrieved.

Example: *My_backend_set*

- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (e.g., if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type None

Return type *Response*

update_listener (*update_listener_details*, *load_balancer_id*, *listener_name*, ***kwargs*)

UpdateListener Updates a listener for a given load balancer.

Parameters

- **update_listener_details** (*UpdateListenerDetails*) – (required) Details to update a listener.
- **load_balancer_id** (*str*) – (required) The OCID of the load balancer associated with the listener to update.
- **listener_name** (*str*) – (required) The name of the listener to update.

Example: *My listener*

- **opc_request_id** (*str*) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.
- **opc_retry_token** (*str*) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (e.g., if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type None

Return type *Response*

update_load_balancer (*update_load_balancer_details*, *load_balancer_id*, ***kwargs*)

UpdateLoadBalancer Updates a load balancer's configuration.

Parameters

- **update_load_balancer_details** (`UpdateLoadBalancerDetails`) – (required) The details for updating a load balancer’s configuration.
- **load_balancer_id** (`str`) – (required) The OCID of the load balancer to update.
- **opc_request_id** (`str`) – (optional) The unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.
- **opc_retry_token** (`str`) – (optional) A token that uniquely identifies a request so it can be retried in case of a timeout or server error without risk of executing that same action again. Retry tokens expire after 24 hours, but can be invalidated before then due to conflicting operations (e.g., if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

Returns A *Response* object with data of type `None`

Return type *Response*

Models

`class oci.load_balancer.models.Backend`

backup

Gets the backup of this Backend. Whether the load balancer should treat this server as a backup unit. If *true*, the load balancer forwards no ingress traffic to this backend server unless all other backend servers not marked as “backup” fail the health check policy.

Example: *true*

Returns The backup of this Backend.

Return type `bool`

drain

Gets the drain of this Backend. Whether the load balancer should drain this server. Servers marked “drain” receive no new incoming traffic.

Example: *true*

Returns The drain of this Backend.

Return type `bool`

ip_address

Gets the `ip_address` of this Backend. The IP address of the backend server.

Example: *10.10.10.4*

Returns The `ip_address` of this Backend.

Return type `str`

name

Gets the name of this Backend. A read-only field showing the IP address and port that uniquely identify this backend server in the backend set.

Example: *10.10.10.4:8080*

Returns The name of this Backend.

Return type `str`

offline

Gets the offline of this Backend. Whether the load balancer should treat this server as offline. Offline servers receive no incoming traffic.

Example: *true*

Returns The offline of this Backend.

Return type bool

port

Gets the port of this Backend. The communication port for the backend server.

Example: *8080*

Returns The port of this Backend.

Return type int

weight

Gets the weight of this Backend. The load balancing policy weight assigned to the server. Backend servers with a higher weight receive a larger proportion of incoming traffic. For example, a server weighted '3' receives 3 times the number of new connections as a server weighted '1'. For more information on load balancing policies, see [How Load Balancing Policies Work](#).

Example: *3*

Returns The weight of this Backend.

Return type int

class `oci.load_balancer.models.BackendDetails`

backup

Gets the backup of this BackendDetails. Whether the load balancer should treat this server as a backup unit. If *true*, the load balancer forwards no ingress traffic to this backend server unless all other backend servers not marked as "backup" fail the health check policy.

Example: *true*

Returns The backup of this BackendDetails.

Return type bool

drain

Gets the drain of this BackendDetails. Whether the load balancer should drain this server. Servers marked "drain" receive no new incoming traffic.

Example: *true*

Returns The drain of this BackendDetails.

Return type bool

ip_address

Gets the ip_address of this BackendDetails. The IP address of the backend server.

Example: *10.10.10.4*

Returns The ip_address of this BackendDetails.

Return type str

offline

Gets the offline of this BackendDetails. Whether the load balancer should treat this server as offline. Offline servers receive no incoming traffic.

Example: *true*

Returns The offline of this BackendDetails.

Return type bool

port

Gets the port of this BackendDetails. The communication port for the backend server.

Example: *8080*

Returns The port of this BackendDetails.

Return type int

weight

Gets the weight of this BackendDetails. The load balancing policy weight assigned to the server. Backend servers with a higher weight receive a larger proportion of incoming traffic. For example, a server weighted '3' receives 3 times the number of new connections as a server weighted '1'. For more information on load balancing policies, see [How Load Balancing Policies Work](#).

Example: *3*

Returns The weight of this BackendDetails.

Return type int

class `oci.load_balancer.models.BackendHealth`

health_check_results

Gets the health_check_results of this BackendHealth. A list of the most recent health check results returned for the specified backend server.

Returns The health_check_results of this BackendHealth.

Return type list[HealthCheckResult]

status

Gets the status of this BackendHealth. The general health status of the specified backend server as reported by the primary and stand-by load balancers.

- OK:** Both health checks returned *OK*.
- WARNING:** One health check returned *OK* and one did not.
- CRITICAL:** Neither health check returned *OK*.
- UNKNOWN:** One or both health checks returned *UNKNOWN*, or the system was unable to retrieve metrics at this time.

Allowed values for this property are: "OK", "WARNING", "CRITICAL", "UNKNOWN", "UNKNOWN_ENUM_VALUE". Any unrecognized values returned by a service will be mapped to "UNKNOWN_ENUM_VALUE".

Returns The status of this BackendHealth.

Return type str

class `oci.load_balancer.models.BackendSet`

backends

Gets the backends of this BackendSet.

Returns The backends of this BackendSet.

Return type list[Backend]

health_checker

Gets the health_checker of this BackendSet.

Returns The health_checker of this BackendSet.

Return type *HealthChecker*

name

Gets the name of this BackendSet. A friendly name for the backend set. It must be unique and it cannot be changed.

Valid backend set names include only alphanumeric characters, dashes, and underscores. Backend set names cannot contain spaces. Avoid entering confidential information.

Example: *My_backend_set*

Returns The name of this BackendSet.

Return type str

policy

Gets the policy of this BackendSet. The load balancer policy for the backend set. The default load balancing policy is 'ROUND_ROBIN' To get a list of available policies, use the `list_policies()` operation.

Example: *LEAST_CONNECTIONS*

Returns The policy of this BackendSet.

Return type str

session_persistence_configuration

Gets the session_persistence_configuration of this BackendSet.

Returns The session_persistence_configuration of this BackendSet.

Return type *SessionPersistenceConfigurationDetails*

ssl_configuration

Gets the ssl_configuration of this BackendSet.

Returns The ssl_configuration of this BackendSet.

Return type *SSLConfiguration*

class `oci.load_balancer.models.BackendSetDetails`

backends

Gets the backends of this BackendSetDetails.

Returns The backends of this BackendSetDetails.

Return type list[BackendDetails]

health_checker

Gets the health_checker of this BackendSetDetails.

Returns The health_checker of this BackendSetDetails.

Return type *HealthCheckerDetails*

policy

Gets the policy of this BackendSetDetails. The load balancer policy for the backend set. The default load balancing policy is 'ROUND_ROBIN'. To get a list of available policies, use the `list_policies()` operation.

Example: *LEAST_CONNECTIONS*

Returns The policy of this BackendSetDetails.

Return type str

session_persistence_configuration

Gets the session_persistence_configuration of this BackendSetDetails.

Returns The session_persistence_configuration of this BackendSetDetails.

Return type *SessionPersistenceConfigurationDetails*

ssl_configuration

Gets the ssl_configuration of this BackendSetDetails.

Returns The ssl_configuration of this BackendSetDetails.

Return type *SSLConfigurationDetails*

class `oci.load_balancer.models.BackendSetHealth`

critical_state_backend_names

Gets the critical_state_backend_names of this BackendSetHealth. A list of backend servers that are currently in the *CRITICAL* health state. The list identifies each backend server by IP address and port.

Example: *1.1.1.1:80*

Returns The critical_state_backend_names of this BackendSetHealth.

Return type list[str]

status

Gets the status of this BackendSetHealth. Overall health status of the backend set.

- OK:** All backend servers in the backend set return a status of *OK*.
- WARNING:** Half or more of the backend set's backend servers return a status of *OK* and at least one backend

server returns a status of *WARNING*, *CRITICAL*, or *UNKNOWN*.

- CRITICAL:** Fewer than half of the backend set's backend servers return a status of *OK*.
- UNKNOWN:** More than half of the backend set's backend servers return a status of *UNKNOWN*, the system was

unable to retrieve metrics, or the backend set does not have a listener attached.

Allowed values for this property are: "OK", "WARNING", "CRITICAL", "UNKNOWN", "UNKNOWN_ENUM_VALUE". Any unrecognized values returned by a service will be mapped to "UNKNOWN_ENUM_VALUE".

Returns The status of this BackendSetHealth.

Return type str

total_backend_count

Gets the total_backend_count of this BackendSetHealth. The total number of backend servers in this backend set.

Example: 5

Returns The `total_backend_count` of this `BackendSetHealth`.

Return type int

unknown_state_backend_names

Gets the `unknown_state_backend_names` of this `BackendSetHealth`. A list of backend servers that are currently in the `UNKNOWN` health state. The list identifies each backend server by IP address and port.

Example: `1.1.1.5:80`

Returns The `unknown_state_backend_names` of this `BackendSetHealth`.

Return type list[str]

warning_state_backend_names

Gets the `warning_state_backend_names` of this `BackendSetHealth`. A list of backend servers that are currently in the `WARNING` health state. The list identifies each backend server by IP address and port.

Example: `1.1.1.7:42`

Returns The `warning_state_backend_names` of this `BackendSetHealth`.

Return type list[str]

class `oci.load_balancer.models.Certificate`

ca_certificate

Gets the `ca_certificate` of this `Certificate`. The Certificate Authority certificate, or any interim certificate, that you received from your SSL certificate provider.

Example:

```
-----BEGIN CERTIFICATE----- MIIeczCCA1ugAwIBAgIBADANBgkqhkiG9w0BAQQFAD..AkGA1UEBhMCR0Ix
EzARBgNVBAgTCiNvbWU3RhdGUxZDASBgNVBAoTC0..0EgTHRkMTcwNQYD
VQQLZy5DbGFzcyAxIFB1YmtpYyBQcm1tYXJ5IENlcn..XRpb24gQXV0aG9y
aXR5MRQwEgYDVQQDEwtCZXN0IENBIEEx0ZDAeFw0wMD..TUwMTZaFw0wMTAy
... -----END CERTIFICATE-----
```

Returns The `ca_certificate` of this `Certificate`.

Return type str

certificate_name

Gets the `certificate_name` of this `Certificate`. A friendly name for the certificate bundle. It must be unique and it cannot be changed. Valid certificate bundle names include only alphanumeric characters, dashes, and underscores. Certificate bundle names cannot contain spaces. Avoid entering confidential information.

Example: `My_certificate_bundle`

Returns The `certificate_name` of this `Certificate`.

Return type str

public_certificate

Gets the `public_certificate` of this `Certificate`. The public certificate, in PEM format, that you received from your SSL certificate provider.

Example:

```
-----BEGIN CERTIFICATE----- MIIC2jCCAkMCAg38MA0GCSqGSIb3DQEBBQUAMIGbMQswCQYDVQQGEwJK
A1UECBMFVG9reW8xEDAObgNVBAcTB0NodW8ta3UxETAPBgNVBAoTCCEZyYW5rNERE
```

```
MRgwFgYDVQQLew9XZWNJZDZlJ0IFN1cHBvcnQxGDAwBgNVBAMTD0ZyYW5rNEREIFdl
YiBDQTEjMCEGCSqGSIb3DQEJARYUc3VwcG9ydEBmcmFuazRkZC5jb20wHhcNMTIw
... —END CERTIFICATE—
```

Returns The `public_certificate` of this Certificate.

Return type str

class `oci.load_balancer.models.CertificateDetails`

ca_certificate

Gets the `ca_certificate` of this CertificateDetails. The Certificate Authority certificate, or any interim certificate, that you received from your SSL certificate provider.

Example:

```
—BEGIN CERTIFICATE— MIIeczCCA1ugAwIBAgIBADANBgkqhkiG9w0BAQQFAD..AkGA1UEBhMCR0Ix
EzARBgNVBAgTCiNvbWUuU3RhdGUxGDAwBgNVBAoTC0..0EgTHRkMTcwNQYD
VQQLey5DbGFzcyAxIFB1YmtpYyBQcm1tYXJ5IENlcn..XRpb24gQXV0aG9y
aXR5MRQwEgYDVQQDEwtCZXN0IENBIEx0ZDAeFw0wMD..TUwMTZaFw0wMTAy
... —END CERTIFICATE—
```

Returns The `ca_certificate` of this CertificateDetails.

Return type str

certificate_name

Gets the `certificate_name` of this CertificateDetails. A friendly name for the certificate bundle. It must be unique and it cannot be changed. Valid certificate bundle names include only alphanumeric characters, dashes, and underscores. Certificate bundle names cannot contain spaces. Avoid entering confidential information.

Example: *My_certificate_bundle*

Returns The `certificate_name` of this CertificateDetails.

Return type str

passphrase

Gets the `passphrase` of this CertificateDetails. A passphrase for encrypted private keys. This is needed only if you created your certificate with a passphrase.

Example: *Mysecretunlockingcode42!!!*

Returns The `passphrase` of this CertificateDetails.

Return type str

private_key

Gets the `private_key` of this CertificateDetails. The SSL private key for your certificate, in PEM format.

Example:

```
—BEGIN RSA PRIVATE KEY— jO1O1v2ftXMsawM90tnXwc6xhOAT1gDBC9S8DKeca..JZNUgYYwNS0dP2UK
tmyN+XqVcAKw4HqVmChXy5b5msu8eIq3uc2NqNVtR..2ksSLukP8pxXcHyb
+sEwvM4uf8qbnHAqwnOnP9+KV9vds6BaH1eRA4CHz..n+NVZlZBsTxTIS16
/Umr7wJzVrMqK5sDiSu4WuaaBdqMGfL5hLsTjcBFD..Da2iyQmSKuVD4IIZ ... —END
RSA PRIVATE KEY—
```

Returns The `private_key` of this CertificateDetails.

Return type str

public_certificate

Gets the `public_certificate` of this `CertificateDetails`. The public certificate, in PEM format, that you received from your SSL certificate provider.

Example:

```
-----BEGIN CERTIFICATE----- MIIC2jCCAkMCAg38MA0GCSqGSIb3DQEBBQUAMIGbMQswCQYDVQQGEwJK
A1UECBMFVG9reW8xEDAOBgNVBAcTB0NodW8ta3UxETAPBgNVBAoTCEZyYW5rNERE
MRgwFgYDVQQLew9XZWJDZXJ0IFN1cHBvcnQxGDAWBgNVBAMTD0ZyYW5rNEREIFdl
YiBDQTEjMCEGCSqGSIb3DQEJARYUc3VwcG9ydEBmcmFuazRkZC5jb20wHhcNMTIw
... -----END CERTIFICATE-----
```

Returns The `public_certificate` of this `CertificateDetails`.

Return type str

class `oci.load_balancer.models.CreateBackendDetails`

backup

Gets the `backup` of this `CreateBackendDetails`. Whether the load balancer should treat this server as a backup unit. If `true`, the load balancer forwards no ingress traffic to this backend server unless all other backend servers not marked as “backup” fail the health check policy.

Example: `true`

Returns The `backup` of this `CreateBackendDetails`.

Return type bool

drain

Gets the `drain` of this `CreateBackendDetails`. Whether the load balancer should drain this server. Servers marked “drain” receive no new incoming traffic.

Example: `true`

Returns The `drain` of this `CreateBackendDetails`.

Return type bool

ip_address

Gets the `ip_address` of this `CreateBackendDetails`. The IP address of the backend server.

Example: `10.10.10.4`

Returns The `ip_address` of this `CreateBackendDetails`.

Return type str

offline

Gets the `offline` of this `CreateBackendDetails`. Whether the load balancer should treat this server as offline. Offline servers receive no incoming traffic.

Example: `true`

Returns The `offline` of this `CreateBackendDetails`.

Return type bool

port

Gets the `port` of this `CreateBackendDetails`. The communication port for the backend server.

Example: *8080*

Returns The port of this CreateBackendDetails.

Return type int

weight

Gets the weight of this CreateBackendDetails. The load balancing policy weight assigned to the server. Backend servers with a higher weight receive a larger proportion of incoming traffic. For example, a server weighted '3' receives 3 times the number of new connections as a server weighted '1'. For more information on load balancing policies, see [How Load Balancing Policies Work](#).

Example: *3*

Returns The weight of this CreateBackendDetails.

Return type int

class `oci.load_balancer.models.CreateBackendSetDetails`

backends

Gets the backends of this CreateBackendSetDetails.

Returns The backends of this CreateBackendSetDetails.

Return type list[BackendDetails]

health_checker

Gets the health_checker of this CreateBackendSetDetails.

Returns The health_checker of this CreateBackendSetDetails.

Return type *HealthCheckerDetails*

name

Gets the name of this CreateBackendSetDetails. A friendly name for the backend set. It must be unique and it cannot be changed.

Valid backend set names include only alphanumeric characters, dashes, and underscores. Backend set names cannot contain spaces. Avoid entering confidential information.

Example: *My_backend_set*

Returns The name of this CreateBackendSetDetails.

Return type str

policy

Gets the policy of this CreateBackendSetDetails. The load balancer policy for the backend set. The default load balancing policy is 'ROUND_ROBIN' To get a list of available policies, use the `list_policies()` operation.

Example: *LEAST_CONNECTIONS*

Returns The policy of this CreateBackendSetDetails.

Return type str

session_persistence_configuration

Gets the session_persistence_configuration of this CreateBackendSetDetails.

Returns The session_persistence_configuration of this CreateBackendSetDetails.

Return type *SessionPersistenceConfigurationDetails*

ssl_configuration

Gets the `ssl_configuration` of this `CreateBackendSetDetails`.

Returns The `ssl_configuration` of this `CreateBackendSetDetails`.

Return type *SSLConfigurationDetails*

class oci.load_balancer.models.CreateCertificateDetails**ca_certificate**

Gets the `ca_certificate` of this `CreateCertificateDetails`. The Certificate Authority certificate, or any interim certificate, that you received from your SSL certificate provider.

Example:

```
-----BEGIN CERTIFICATE----- MIIEcZCCA1ugAwIBAgIBADANBgkqhkiG9w0BAQQFAD..AkGA1UEBhMCR0Ix
EzARBgNVBAgTCiNvbWUtU3RhdGUxFDASBgNVBAoTC0..0EgTHRkMTcwNQYD
VQQLEx5DbGFzcyAxIFB1YmtpYyBQcm1tYXJ5IENlcn..XRpb24gQXV0aG9y
aXR5MRQwEgYDVQQDEwtCZXN0IENBIEx0ZDAeFw0wMD..TUwMTZaFw0wMTAy
... -----END CERTIFICATE-----
```

Returns The `ca_certificate` of this `CreateCertificateDetails`.

Return type `str`

certificate_name

Gets the `certificate_name` of this `CreateCertificateDetails`. A friendly name for the certificate bundle. It must be unique and it cannot be changed. Valid certificate bundle names include only alphanumeric characters, dashes, and underscores. Certificate bundle names cannot contain spaces. Avoid entering confidential information.

Example: *My_certificate_bundle*

Returns The `certificate_name` of this `CreateCertificateDetails`.

Return type `str`

passphrase

Gets the `passphrase` of this `CreateCertificateDetails`. A passphrase for encrypted private keys. This is needed only if you created your certificate with a passphrase.

Example: *Mysecretunlockingcode42!!*

Returns The `passphrase` of this `CreateCertificateDetails`.

Return type `str`

private_key

Gets the `private_key` of this `CreateCertificateDetails`. The SSL private key for your certificate, in PEM format.

Example:

```
-----BEGIN RSA PRIVATE KEY----- jO1O1v2ftXMsawM90tnXwc6xhOAT1gDBC9S8DKeca..JZNUgYYwNS0dP2UK
tmyN+XqVcAKw4HqVmChXy5b5msu8eIq3uc2NqNVtR..2ksSLukP8pxXcHyb
+sEwvM4uf8qbnHAqwnOnP9+KV9vds6BaH1eRA4CHz..n+NVZlZBsTxTIS16
/Umr7wJzVrMqK5sDiSu4WuaaBdqMGfL5hLsTjcBFD..Da2iyQmSKuVD4IIZ ... -----END
RSA PRIVATE KEY-----
```

Returns The `private_key` of this `CreateCertificateDetails`.

Return type `str`

public_certificate

Gets the `public_certificate` of this `CreateCertificateDetails`. The public certificate, in PEM format, that you received from your SSL certificate provider.

Example:

```
-----BEGIN CERTIFICATE----- MIIC2jCCAkMCAg38MA0GCSqGSIb3DQEBBQUAMIGbMQswCQYDVQQGEwJK
A1UECBMFVG9reW8xEDAOBgNVBAcTB0NodW8ta3UxETAPBgNVBAoTCEZyYW5rNERE
MRgwFgYDVQQLEw9XZWJDZXJ0IFN1cHBvcnQxGDAWBgNVBAMTD0ZyYW5rNEREIFdl
YiBDQTEjMCEGCSqGSIb3DQEJARYUc3VwcG9ydEBmcmFuazRkZC5jb20wHhcNMTIw
... -----END CERTIFICATE-----
```

Returns The `public_certificate` of this `CreateCertificateDetails`.

Return type `str`

class oci.load_balancer.models.CreateListenerDetails**default_backend_set_name**

Gets the `default_backend_set_name` of this `CreateListenerDetails`. The name of the associated backend set.

Returns The `default_backend_set_name` of this `CreateListenerDetails`.

Return type `str`

name

Gets the name of this `CreateListenerDetails`. A friendly name for the listener. It must be unique and it cannot be changed. Avoid entering confidential information.

Example: *My listener*

Returns The name of this `CreateListenerDetails`.

Return type `str`

port

Gets the port of this `CreateListenerDetails`. The communication port for the listener.

Example: *80*

Returns The port of this `CreateListenerDetails`.

Return type `int`

protocol

Gets the protocol of this `CreateListenerDetails`. The protocol on which the listener accepts connection requests. To get a list of valid protocols, use the `list_protocols()` operation.

Example: *HTTP*

Returns The protocol of this `CreateListenerDetails`.

Return type `str`

ssl_configuration

Gets the `ssl_configuration` of this `CreateListenerDetails`.

Returns The `ssl_configuration` of this `CreateListenerDetails`.

Return type *SSLConfigurationDetails*

class oci.load_balancer.models.CreateLoadBalancerDetails

backend_sets

Gets the backend_sets of this CreateLoadBalancerDetails.

Returns The backend_sets of this CreateLoadBalancerDetails.

Return type dict(str, BackendSetDetails)

certificates

Gets the certificates of this CreateLoadBalancerDetails.

Returns The certificates of this CreateLoadBalancerDetails.

Return type dict(str, CertificateDetails)

compartment_id

Gets the compartment_id of this CreateLoadBalancerDetails. The [OCID](#) of the compartment in which to create the load balancer.

Returns The compartment_id of this CreateLoadBalancerDetails.

Return type str

display_name

Gets the display_name of this CreateLoadBalancerDetails. A user-friendly name. It does not have to be unique, and it is changeable. Avoid entering confidential information.

Example: *My load balancer*

Returns The display_name of this CreateLoadBalancerDetails.

Return type str

is_private

Gets the is_private of this CreateLoadBalancerDetails. Whether the load balancer has a VCN-local (private) IP address.

If “true”, the service assigns a private IP address to the load balancer. The load balancer requires only one subnet to host both the primary and secondary load balancers. The private IP address is local to the subnet. The load balancer is accessible only from within the VCN that contains the associated subnet, or as further restricted by your security list rules. The load balancer can route traffic to any backend server that is reachable from the VCN.

For a private load balancer, both the primary and secondary load balancer hosts are within the same Availability Domain.

If “false”, the service assigns a public IP address to the load balancer. A load balancer with a public IP address requires two subnets, each in a different Availability Domain. One subnet hosts the primary load balancer and the other hosts the secondary (stand-by) load balancer. A public load balancer is accessible from the internet, depending on your VCN’s [security list rules](#).

Example: *false*

Returns The is_private of this CreateLoadBalancerDetails.

Return type bool

listeners

Gets the listeners of this CreateLoadBalancerDetails.

Returns The listeners of this CreateLoadBalancerDetails.

Return type dict(str, ListenerDetails)

shape_name

Gets the `shape_name` of this `CreateLoadBalancerDetails`. A template that determines the total pre-provisioned bandwidth (ingress plus egress). To get a list of available shapes, use the `list_shapes()` operation.

Example: *100Mbps*

Returns The `shape_name` of this `CreateLoadBalancerDetails`.

Return type str

subnet_ids

Gets the `subnet_ids` of this `CreateLoadBalancerDetails`. An array of subnet OCIDs.

Returns The `subnet_ids` of this `CreateLoadBalancerDetails`.

Return type list[str]

class `oci.load_balancer.models.HealthCheckResult`

health_check_status

Gets the `health_check_status` of this `HealthCheckResult`. The result of the most recent health check.

Allowed values for this property are: "OK", "INVALID_STATUS_CODE", "TIMED_OUT", "REGEX_MISMATCH", "CONNECT_FAILED", "IO_ERROR", "OFFLINE", "UNKNOWN", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The `health_check_status` of this `HealthCheckResult`.

Return type str

source_ip_address

Gets the `source_ip_address` of this `HealthCheckResult`. The IP address of the health check status report provider. This identifier helps you differentiate same-subnet (private) load balancers that report health check status.

Example: *10.2.0.1*

Returns The `source_ip_address` of this `HealthCheckResult`.

Return type str

subnet_id

Gets the `subnet_id` of this `HealthCheckResult`. The OCID of the subnet hosting the load balancer that reported this health check status.

Returns The `subnet_id` of this `HealthCheckResult`.

Return type str

timestamp

Gets the `timestamp` of this `HealthCheckResult`. The date and time the data was retrieved, in the format defined by RFC3339.

Example: *2017-06-02T18:28:11+00:00*

Returns The `timestamp` of this `HealthCheckResult`.

Return type datetime

class `oci.load_balancer.models.HealthChecker`

interval_in_millis

Gets the `interval_in_millis` of this HealthChecker. The interval between health checks, in milliseconds. The default is 10000 (10 seconds).

Example: *30000*

Returns The `interval_in_millis` of this HealthChecker.

Return type `int`

port

Gets the port of this HealthChecker. The backend server port against which to run the health check. If the port is not specified, the load balancer uses the port information from the *Backend* object.

Example: *8080*

Returns The port of this HealthChecker.

Return type `int`

protocol

Gets the protocol of this HealthChecker. The protocol the health check must use; either HTTP or TCP.

Example: *HTTP*

Returns The protocol of this HealthChecker.

Return type `str`

response_body_regex

Gets the `response_body_regex` of this HealthChecker. A regular expression for parsing the response body from the backend server.

Example: *^(500|40[1348])\$*

Returns The `response_body_regex` of this HealthChecker.

Return type `str`

retries

Gets the `retries` of this HealthChecker. The number of retries to attempt before a backend server is considered “unhealthy”. Defaults to 3.

Example: *3*

Returns The `retries` of this HealthChecker.

Return type `int`

return_code

Gets the `return_code` of this HealthChecker. The status code a healthy backend server should return. If you configure the health check policy to use the HTTP protocol, you can use common HTTP status codes such as “200”.

Example: *200*

Returns The `return_code` of this HealthChecker.

Return type `int`

timeout_in_millis

Gets the `timeout_in_millis` of this HealthChecker. The maximum time, in milliseconds, to wait for a reply to a health check. A health check is successful only if a reply returns within this timeout period. Defaults to 3000 (3 seconds).

Example: *6000*

Returns The `timeout_in_millis` of this `HealthChecker`.

Return type `int`

url_path

Gets the `url_path` of this `HealthChecker`. The path against which to run the health check.

Example: `/healthcheck`

Returns The `url_path` of this `HealthChecker`.

Return type `str`

class `oci.load_balancer.models.HealthCheckerDetails`

interval_in_millis

Gets the `interval_in_millis` of this `HealthCheckerDetails`. The interval between health checks, in milliseconds.

Example: `30000`

Returns The `interval_in_millis` of this `HealthCheckerDetails`.

Return type `int`

port

Gets the port of this `HealthCheckerDetails`. The backend server port against which to run the health check. If the port is not specified, the load balancer uses the port information from the `Backend` object.

Example: `8080`

Returns The port of this `HealthCheckerDetails`.

Return type `int`

protocol

Gets the protocol of this `HealthCheckerDetails`. The protocol the health check must use; either HTTP or TCP.

Example: `HTTP`

Returns The protocol of this `HealthCheckerDetails`.

Return type `str`

response_body_regex

Gets the `response_body_regex` of this `HealthCheckerDetails`. A regular expression for parsing the response body from the backend server.

Example: `^(500|40[1348])$`

Returns The `response_body_regex` of this `HealthCheckerDetails`.

Return type `str`

retries

Gets the `retries` of this `HealthCheckerDetails`. The number of retries to attempt before a backend server is considered “unhealthy”.

Example: `3`

Returns The `retries` of this `HealthCheckerDetails`.

Return type `int`

return_code

Gets the return_code of this HealthCheckerDetails. The status code a healthy backend server should return.

Example: *200*

Returns The return_code of this HealthCheckerDetails.

Return type int

timeout_in_millis

Gets the timeout_in_millis of this HealthCheckerDetails. The maximum time, in milliseconds, to wait for a reply to a health check. A health check is successful only if a reply returns within this timeout period.

Example: *6000*

Returns The timeout_in_millis of this HealthCheckerDetails.

Return type int

url_path

Gets the url_path of this HealthCheckerDetails. The path against which to run the health check.

Example: */healthcheck*

Returns The url_path of this HealthCheckerDetails.

Return type str

class oci.load_balancer.models.**IpAddress**

ip_address

Gets the ip_address of this IpAddress. An IP address.

Example: *128.148.10.20*

Returns The ip_address of this IpAddress.

Return type str

is_public

Gets the is_public of this IpAddress. Whether the IP address is public or private.

If “true”, the IP address is public and accessible from the internet.

If “false”, the IP address is private and accessible only from within the associated VCN.

Returns The is_public of this IpAddress.

Return type bool

class oci.load_balancer.models.**Listener**

default_backend_set_name

Gets the default_backend_set_name of this Listener. The name of the associated backend set.

Returns The default_backend_set_name of this Listener.

Return type str

name

Gets the name of this Listener. A friendly name for the listener. It must be unique and it cannot be changed.

Example: *My listener*

Returns The name of this Listener.

Return type str

port

Gets the port of this Listener. The communication port for the listener.

Example: *80*

Returns The port of this Listener.

Return type int

protocol

Gets the protocol of this Listener. The protocol on which the listener accepts connection requests. To get a list of valid protocols, use the `list_protocols()` operation.

Example: *HTTP*

Returns The protocol of this Listener.

Return type str

ssl_configuration

Gets the ssl_configuration of this Listener.

Returns The ssl_configuration of this Listener.

Return type *SSLConfiguration*

class `oci.load_balancer.models.ListenerDetails`

default_backend_set_name

Gets the default_backend_set_name of this ListenerDetails. The name of the associated backend set.

Returns The default_backend_set_name of this ListenerDetails.

Return type str

port

Gets the port of this ListenerDetails. The communication port for the listener.

Example: *80*

Returns The port of this ListenerDetails.

Return type int

protocol

Gets the protocol of this ListenerDetails. The protocol on which the listener accepts connection requests. To get a list of valid protocols, use the `list_protocols()` operation.

Example: *HTTP*

Returns The protocol of this ListenerDetails.

Return type str

ssl_configuration

Gets the ssl_configuration of this ListenerDetails.

Returns The ssl_configuration of this ListenerDetails.

Return type *SSLConfigurationDetails*

class `oci.load_balancer.models.LoadBalancer`

backend_sets

Gets the backend_sets of this LoadBalancer.

Returns The backend_sets of this LoadBalancer.

Return type dict(str, BackendSet)

certificates

Gets the certificates of this LoadBalancer.

Returns The certificates of this LoadBalancer.

Return type dict(str, Certificate)

compartment_id

Gets the compartment_id of this LoadBalancer. The **OCID** of the compartment containing the load balancer.

Returns The compartment_id of this LoadBalancer.

Return type str

display_name

Gets the display_name of this LoadBalancer. A user-friendly name. It does not have to be unique, and it is changeable.

Example: *My load balancer*

Returns The display_name of this LoadBalancer.

Return type str

id

Gets the id of this LoadBalancer. The **OCID** of the load balancer.

Returns The id of this LoadBalancer.

Return type str

ip_addresses

Gets the ip_addresses of this LoadBalancer. An array of IP addresses.

Returns The ip_addresses of this LoadBalancer.

Return type list[IpAddress]

is_private

Gets the is_private of this LoadBalancer. Whether the load balancer has a VCN-local (private) IP address.

If “true”, the service assigns a private IP address to the load balancer. The load balancer requires only one subnet to host both the primary and secondary load balancers. The private IP address is local to the subnet. The load balancer is accessible only from within the VCN that contains the associated subnet, or as further restricted by your security list rules. The load balancer can route traffic to any backend server that is reachable from the VCN.

For a private load balancer, both the primary and secondary load balancer hosts are within the same Availability Domain.

If “false”, the service assigns a public IP address to the load balancer. A load balancer with a public IP address requires two subnets, each in a different Availability Domain. One subnet hosts the primary load balancer and the other hosts the secondary (stand-by) load balancer. A public load balancer is accessible from the internet, depending on your VCN’s *security list rules*.

Returns The is_private of this LoadBalancer.

Return type bool

lifecycle_state

Gets the `lifecycle_state` of this `LoadBalancer`. Allowed values for this property are: “CREATING”, “FAILED”, “ACTIVE”, “DELETING”, “DELETED”, ‘UNKNOWN_ENUM_VALUE’. Any unrecognized values returned by a service will be mapped to ‘UNKNOWN_ENUM_VALUE’.

Returns The `lifecycle_state` of this `LoadBalancer`.

Return type str

listeners

Gets the listeners of this `LoadBalancer`.

Returns The listeners of this `LoadBalancer`.

Return type dict(str, Listener)

shape_name

Gets the `shape_name` of this `LoadBalancer`. A template that determines the total pre-provisioned bandwidth (ingress plus egress). To get a list of available shapes, use the `list_shapes()` operation.

Example: *100Mbps*

Returns The `shape_name` of this `LoadBalancer`.

Return type str

subnet_ids

Gets the `subnet_ids` of this `LoadBalancer`. An array of subnet OCIDs.

Returns The `subnet_ids` of this `LoadBalancer`.

Return type list[str]

time_created

Gets the `time_created` of this `LoadBalancer`. The date and time the load balancer was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The `time_created` of this `LoadBalancer`.

Return type datetime

class `oci.load_balancer.models.LoadBalancerHealth`

critical_state_backend_set_names

Gets the `critical_state_backend_set_names` of this `LoadBalancerHealth`. A list of backend sets that are currently in the *CRITICAL* health state. The list identifies each backend set by the friendly name you assigned when you created it.

Example: *My_backend_set*

Returns The `critical_state_backend_set_names` of this `LoadBalancerHealth`.

Return type list[str]

status

Gets the status of this `LoadBalancerHealth`. The overall health status of the load balancer.

- OK:** All backend sets associated with the load balancer return a status of *OK*.
- WARNING:** At least one of the backend sets associated with the load balancer returns a status of *WARNING*,

no backend sets return a status of *CRITICAL*, and the load balancer life cycle state is *ACTIVE*.

- CRITICAL:** One or more of the backend sets associated with the load balancer return a status of *CRITICAL*.
- UNKNOWN:** If any one of the following conditions is true:
 - The load balancer life cycle state is not *ACTIVE*.
 - No backend sets are defined for the load balancer.
 - More than half of the backend sets associated with the load balancer return a status of *UNKNOWN*, none of the backend sets return a status of *WARNING* or *CRITICAL*, and the load balancer life cycle state is *ACTIVE*.
 - The system could not retrieve metrics for any reason.

Allowed values for this property are: “OK”, “WARNING”, “CRITICAL”, “UNKNOWN”, ‘UNKNOWN_ENUM_VALUE’. Any unrecognized values returned by a service will be mapped to ‘UNKNOWN_ENUM_VALUE’.

Returns The status of this LoadBalancerHealth.

Return type str

total_backend_set_count

Gets the `total_backend_set_count` of this LoadBalancerHealth. The total number of backend sets associated with this load balancer.

Example: 4

Returns The `total_backend_set_count` of this LoadBalancerHealth.

Return type int

unknown_state_backend_set_names

Gets the `unknown_state_backend_set_names` of this LoadBalancerHealth. A list of backend sets that are currently in the *UNKNOWN* health state. The list identifies each backend set by the friendly name you assigned when you created it.

Example: *Backend set2*

Returns The `unknown_state_backend_set_names` of this LoadBalancerHealth.

Return type list[str]

warning_state_backend_set_names

Gets the `warning_state_backend_set_names` of this LoadBalancerHealth. A list of backend sets that are currently in the *WARNING* health state. The list identifies each backend set by the friendly name you assigned when you created it.

Example: *Backend set3*

Returns The `warning_state_backend_set_names` of this LoadBalancerHealth.

Return type list[str]

class `oci.load_balancer.models.LoadBalancerHealthSummary`

load_balancer_id

Gets the `load_balancer_id` of this LoadBalancerHealthSummary. The *OCID* of the load balancer the health status is associated with.

Returns The `load_balancer_id` of this LoadBalancerHealthSummary.

Return type str

status

Gets the status of this LoadBalancerHealthSummary. The overall health status of the load balancer.

- OK:** All backend sets associated with the load balancer return a status of *OK*.
- WARNING:** At least one of the backend sets associated with the load balancer returns a status of *WARNING*,

no backend sets return a status of *CRITICAL*, and the load balancer life cycle state is *ACTIVE*.

- CRITICAL:** One or more of the backend sets associated with the load balancer return a status of *CRITICAL*.

- UNKNOWN:** If any one of the following conditions is true:

- The load balancer life cycle state is not *ACTIVE*.
- No backend sets are defined for the load balancer.
- More than half of the backend sets associated with the load balancer return a status of *UNKNOWN*, none of the backend sets return a status of *WARNING* or *CRITICAL*, and the load balancer life cycle state is *ACTIVE*.
- The system could not retrieve metrics for any reason.

Allowed values for this property are: “OK”, “WARNING”, “CRITICAL”, “UNKNOWN”, ‘UNKNOWN_ENUM_VALUE’. Any unrecognized values returned by a service will be mapped to ‘UNKNOWN_ENUM_VALUE’.

Returns The status of this LoadBalancerHealthSummary.

Return type str

```
class oci.load_balancer.models.LoadBalancerPolicy
```

name

Gets the name of this LoadBalancerPolicy. The name of the load balancing policy.

Returns The name of this LoadBalancerPolicy.

Return type str

```
class oci.load_balancer.models.LoadBalancerProtocol
```

name

Gets the name of this LoadBalancerProtocol. The name of the protocol.

Returns The name of this LoadBalancerProtocol.

Return type str

```
class oci.load_balancer.models.LoadBalancerShape
```

name

Gets the name of this LoadBalancerShape. The name of the shape.

Returns The name of this LoadBalancerShape.

Return type str

```
class oci.load_balancer.models.SSLConfiguration
```

certificate_name

Gets the `certificate_name` of this `SSLConfiguration`. A friendly name for the certificate bundle. It must be unique and it cannot be changed. Valid certificate bundle names include only alphanumeric characters, dashes, and underscores. Certificate bundle names cannot contain spaces. Avoid entering confidential information.

Example: *My_certificate_bundle*

Returns The `certificate_name` of this `SSLConfiguration`.

Return type str

verify_depth

Gets the `verify_depth` of this `SSLConfiguration`. The maximum depth for peer certificate chain verification.

Example: 3

Returns The `verify_depth` of this `SSLConfiguration`.

Return type int

verify_peer_certificate

Gets the `verify_peer_certificate` of this `SSLConfiguration`. Whether the load balancer listener should verify peer certificates.

Example: *true*

Returns The `verify_peer_certificate` of this `SSLConfiguration`.

Return type bool

class `oci.load_balancer.models.SSLConfigurationDetails`

certificate_name

Gets the `certificate_name` of this `SSLConfigurationDetails`. A friendly name for the certificate bundle. It must be unique and it cannot be changed. Valid certificate bundle names include only alphanumeric characters, dashes, and underscores. Certificate bundle names cannot contain spaces. Avoid entering confidential information.

Example: *My_certificate_bundle*

Returns The `certificate_name` of this `SSLConfigurationDetails`.

Return type str

verify_depth

Gets the `verify_depth` of this `SSLConfigurationDetails`. The maximum depth for peer certificate chain verification.

Example: 3

Returns The `verify_depth` of this `SSLConfigurationDetails`.

Return type int

verify_peer_certificate

Gets the `verify_peer_certificate` of this `SSLConfigurationDetails`. Whether the load balancer listener should verify peer certificates.

Example: *true*

Returns The `verify_peer_certificate` of this `SSLConfigurationDetails`.

Return type bool

```
class oci.load_balancer.models.SessionPersistenceConfigurationDetails
```

cookie_name

Gets the `cookie_name` of this `SessionPersistenceConfigurationDetails`. The name of the cookie used to detect a session initiated by the backend server. Use `*` to specify that any cookie set by the backend causes the session to persist.

Example: *myCookieName*

Returns The `cookie_name` of this `SessionPersistenceConfigurationDetails`.

Return type str

disable_fallback

Gets the `disable_fallback` of this `SessionPersistenceConfigurationDetails`. Whether the load balancer is prevented from directing traffic from a persistent session client to a different backend server if the original server is unavailable. Defaults to false.

Example: *true*

Returns The `disable_fallback` of this `SessionPersistenceConfigurationDetails`.

Return type bool

```
class oci.load_balancer.models.UpdateBackendDetails
```

backup

Gets the `backup` of this `UpdateBackendDetails`. Whether the load balancer should treat this server as a backup unit. If *true*, the load balancer forwards no ingress traffic to this backend server unless all other backend servers not marked as “backup” fail the health check policy.

Example: *true*

Returns The `backup` of this `UpdateBackendDetails`.

Return type bool

drain

Gets the `drain` of this `UpdateBackendDetails`. Whether the load balancer should drain this server. Servers marked “drain” receive no new incoming traffic.

Example: *true*

Returns The `drain` of this `UpdateBackendDetails`.

Return type bool

offline

Gets the `offline` of this `UpdateBackendDetails`. Whether the load balancer should treat this server as offline. Offline servers receive no incoming traffic.

Example: *true*

Returns The `offline` of this `UpdateBackendDetails`.

Return type bool

weight

Gets the `weight` of this `UpdateBackendDetails`. The load balancing policy weight assigned to the server. Backend servers with a higher weight receive a larger proportion of incoming traffic. For example, a server weighted ‘3’ receives 3 times the number of new connections as a server weighted ‘1’. For more information on load balancing policies, see [How Load Balancing Policies Work](#).

Example: 3

Returns The weight of this UpdateBackendDetails.

Return type int

class `oci.load_balancer.models.UpdateBackendSetDetails`

backends

Gets the backends of this UpdateBackendSetDetails.

Returns The backends of this UpdateBackendSetDetails.

Return type list[BackendDetails]

health_checker

Gets the health_checker of this UpdateBackendSetDetails.

Returns The health_checker of this UpdateBackendSetDetails.

Return type *HealthCheckerDetails*

policy

Gets the policy of this UpdateBackendSetDetails. The load balancer policy for the backend set. The default load balancing policy is 'ROUND_ROBIN' To get a list of available policies, use the `list_policies()` operation.

Example: *LEAST_CONNECTIONS*

Returns The policy of this UpdateBackendSetDetails.

Return type str

session_persistence_configuration

Gets the session_persistence_configuration of this UpdateBackendSetDetails.

Returns The session_persistence_configuration of this UpdateBackendSetDetails.

Return type *SessionPersistenceConfigurationDetails*

ssl_configuration

Gets the ssl_configuration of this UpdateBackendSetDetails.

Returns The ssl_configuration of this UpdateBackendSetDetails.

Return type *SSLConfigurationDetails*

class `oci.load_balancer.models.UpdateHealthCheckerDetails`

interval_in_millis

Gets the interval_in_millis of this UpdateHealthCheckerDetails. The interval between health checks, in milliseconds.

Example: *30000*

Returns The interval_in_millis of this UpdateHealthCheckerDetails.

Return type int

port

Gets the port of this UpdateHealthCheckerDetails. The backend server port against which to run the health check.

Example: *8080*

Returns The port of this UpdateHealthCheckerDetails.

Return type int

protocol

Gets the protocol of this UpdateHealthCheckerDetails. The protocol the health check must use; either HTTP or TCP.

Example: *HTTP*

Returns The protocol of this UpdateHealthCheckerDetails.

Return type str

response_body_regex

Gets the response_body_regex of this UpdateHealthCheckerDetails. A regular expression for parsing the response body from the backend server.

Example: *^(500|40[1348])\$*

Returns The response_body_regex of this UpdateHealthCheckerDetails.

Return type str

retries

Gets the retries of this UpdateHealthCheckerDetails. The number of retries to attempt before a backend server is considered “unhealthy”.

Example: *3*

Returns The retries of this UpdateHealthCheckerDetails.

Return type int

return_code

Gets the return_code of this UpdateHealthCheckerDetails. The status code a healthy backend server should return.

Example: *200*

Returns The return_code of this UpdateHealthCheckerDetails.

Return type int

timeout_in_millis

Gets the timeout_in_millis of this UpdateHealthCheckerDetails. The maximum time, in milliseconds, to wait for a reply to a health check. A health check is successful only if a reply returns within this timeout period.

Example: *6000*

Returns The timeout_in_millis of this UpdateHealthCheckerDetails.

Return type int

url_path

Gets the url_path of this UpdateHealthCheckerDetails. The path against which to run the health check.

Example: */healthcheck*

Returns The url_path of this UpdateHealthCheckerDetails.

Return type str

class `oci.load_balancer.models.UpdateListenerDetails`

default_backend_set_name

Gets the `default_backend_set_name` of this `UpdateListenerDetails`. The name of the associated backend set.

Returns The `default_backend_set_name` of this `UpdateListenerDetails`.

Return type str

port

Gets the port of this `UpdateListenerDetails`. The communication port for the listener.

Example: *80*

Returns The port of this `UpdateListenerDetails`.

Return type int

protocol

Gets the protocol of this `UpdateListenerDetails`. The protocol on which the listener accepts connection requests. To get a list of valid protocols, use the `list_protocols()` operation.

Example: *HTTP*

Returns The protocol of this `UpdateListenerDetails`.

Return type str

ssl_configuration

Gets the `ssl_configuration` of this `UpdateListenerDetails`.

Returns The `ssl_configuration` of this `UpdateListenerDetails`.

Return type *SSLConfigurationDetails*

class `oci.load_balancer.models.UpdateLoadBalancerDetails`

display_name

Gets the `display_name` of this `UpdateLoadBalancerDetails`. The user-friendly display name for the load balancer. It does not have to be unique, and it is changeable. Avoid entering confidential information.

Example: *My load balancer*

Returns The `display_name` of this `UpdateLoadBalancerDetails`.

Return type str

class `oci.load_balancer.models.WorkRequest`

error_details

Gets the `error_details` of this `WorkRequest`.

Returns The `error_details` of this `WorkRequest`.

Return type list[`WorkRequestError`]

id

Gets the id of this `WorkRequest`. The **OCID** of the work request.

Returns The id of this `WorkRequest`.

Return type str

lifecycle_state

Gets the `lifecycle_state` of this `WorkRequest`. Allowed values for this property are: "ACCEPTED",

“IN_PROGRESS”, “FAILED”, “SUCCEEDED”, ‘UNKNOWN_ENUM_VALUE’. Any unrecognized values returned by a service will be mapped to ‘UNKNOWN_ENUM_VALUE’.

Returns The lifecycle_state of this WorkRequest.

Return type str

load_balancer_id

Gets the load_balancer_id of this WorkRequest. The OCID of the load balancer with which the work request is associated.

Returns The load_balancer_id of this WorkRequest.

Return type str

message

Gets the message of this WorkRequest. A collection of data, related to the load balancer provisioning process, that helps with debugging in the event of failure. Possible data elements include:

- workflow name
- event ID
- work request ID
- load balancer ID
- workflow completion message

Returns The message of this WorkRequest.

Return type str

time_accepted

Gets the time_accepted of this WorkRequest. The date and time the work request was created, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The time_accepted of this WorkRequest.

Return type datetime

time_finished

Gets the time_finished of this WorkRequest. The date and time the work request was completed, in the format defined by RFC3339.

Example: *2016-08-25T21:10:29.600Z*

Returns The time_finished of this WorkRequest.

Return type datetime

type

Gets the type of this WorkRequest. The type of action the work request represents.

Returns The type of this WorkRequest.

Return type str

class oci.load_balancer.models.**WorkRequestError**

error_code

Gets the error_code of this WorkRequestError. Allowed values for this property are: “BAD_INPUT”,

“INTERNAL_ERROR”, ‘UNKNOWN_ENUM_VALUE’. Any unrecognized values returned by a service will be mapped to ‘UNKNOWN_ENUM_VALUE’.

Returns The error_code of this WorkRequestError.

Return type str

message

Gets the message of this WorkRequestError. A human-readable error string.

Returns The message of this WorkRequestError.

Return type str

Object Storage

Client

class oci.object_storage.object_storage_client.**ObjectStorageClient** (*config*)

abort_multipart_upload (*namespace_name*, *bucket_name*, *object_name*, *upload_id*, ***kwargs*)

AbortMultipartUpload Aborts an in-progress multipart upload and deletes all parts that have been uploaded.

Parameters

- **namespace_name** (*str*) – (required) The top-level namespace used for the request.
- **bucket_name** (*str*) – (required) The name of the bucket.
Example: *my-new-bucket1*
- **object_name** (*str*) – (required) The name of the object.
Example: *test/object1.log*
- **upload_id** (*str*) – (required) The upload ID for a multipart upload.
- **opc_client_request_id** (*str*) – (optional) The client request ID for tracing.

Returns A *Response* object with data of type None

Return type *Response*

commit_multipart_upload (*namespace_name*, *bucket_name*, *object_name*, *upload_id*, *commit_multipart_upload_details*, ***kwargs*)

CommitMultipartUpload Commits a multipart upload, which involves checking part numbers and ETags of the parts, to create an aggregate object.

Parameters

- **namespace_name** (*str*) – (required) The top-level namespace used for the request.
- **bucket_name** (*str*) – (required) The name of the bucket.
Example: *my-new-bucket1*
- **object_name** (*str*) – (required) The name of the object.
Example: *test/object1.log*
- **upload_id** (*str*) – (required) The upload ID for a multipart upload.

- **commit_multipart_upload_details** (`CommitMultipartUploadDetails`) – (required) The part numbers and ETags for the parts you want to commit.
- **if_match** (`str`) – (optional) The entity tag to match. For creating and committing a multipart upload to an object, this is the entity tag of the target object. For uploading a part, this is the entity tag of the target part.
- **if_none_match** (`str`) – (optional) The entity tag to avoid matching. The only valid value is ‘*’, which indicates that the request should fail if the object already exists. For creating and committing a multipart upload, this is the entity tag of the target object. For uploading a part, this is the entity tag of the target part.
- **opc_client_request_id** (`str`) – (optional) The client request ID for tracing.

Returns A `Response` object with data of type `None`

Return type `Response`

create_bucket (`namespace_name`, `create_bucket_details`, `**kwargs`)

`CreateBucket` Creates a bucket in the given namespace with a bucket name and optional user-defined metadata.

To use this and other API operations, you must be authorized in an IAM policy. If you’re not authorized, talk to an administrator. If you’re an administrator who needs to write policies to give users access, see [Getting Started with Policies](#).

Parameters

- **namespace_name** (`str`) – (required) The top-level namespace used for the request.
- **create_bucket_details** (`CreateBucketDetails`) – (required) Request object for creating a bucket.
- **opc_client_request_id** (`str`) – (optional) The client request ID for tracing.

Returns A `Response` object with data of type `Bucket`

Return type `Response`

create_multipart_upload (`namespace_name`, `bucket_name`, `create_multipart_upload_details`, `**kwargs`)

`CreateMultipartUpload` Starts a new multipart upload to a specific object in the given bucket in the given namespace.

Parameters

- **namespace_name** (`str`) – (required) The top-level namespace used for the request.
- **bucket_name** (`str`) – (required) The name of the bucket.
Example: `my-new-bucket1`
- **create_multipart_upload_details** (`CreateMultipartUploadDetails`) – (required) Request object for creating a multi-part upload.
- **if_match** (`str`) – (optional) The entity tag to match. For creating and committing a multipart upload to an object, this is the entity tag of the target object. For uploading a part, this is the entity tag of the target part.
- **if_none_match** (`str`) – (optional) The entity tag to avoid matching. The only valid value is ‘*’, which indicates that the request should fail if the object already exists. For creating and committing a multipart upload, this is the entity tag of the target object. For uploading a part, this is the entity tag of the target part.
- **opc_client_request_id** (`str`) – (optional) The client request ID for tracing.

Returns A *Response* object with data of type *MultipartUpload*

Return type *Response*

create_preauthenticated_request (*namespace_name*, *bucket_name*, *create_preauthenticated_request_details*, ***kwargs*)

CreatePreauthenticatedRequest Create a pre-authenticated request specific to the bucket

Parameters

- **namespace_name** (*str*) – (required) The top-level namespace used for the request.
- **bucket_name** (*str*) – (required) The name of the bucket.
Example: *my-new-bucket1*
- **create_preauthenticated_request_details** (*CreatePreauthenticatedRequestDetails*) – (required) details for creating the pre-authenticated request.
- **opc_client_request_id** (*str*) – (optional) The client request ID for tracing.

Returns A *Response* object with data of type *PreauthenticatedRequest*

Return type *Response*

delete_bucket (*namespace_name*, *bucket_name*, ***kwargs*)

DeleteBucket Deletes a bucket if it is already empty. If the bucket is not empty, use `delete_object()` first.

Parameters

- **namespace_name** (*str*) – (required) The top-level namespace used for the request.
- **bucket_name** (*str*) – (required) The name of the bucket.
Example: *my-new-bucket1*
- **if_match** (*str*) – (optional) The entity tag to match. For creating and committing a multipart upload to an object, this is the entity tag of the target object. For uploading a part, this is the entity tag of the target part.
- **opc_client_request_id** (*str*) – (optional) The client request ID for tracing.

Returns A *Response* object with data of type *None*

Return type *Response*

delete_object (*namespace_name*, *bucket_name*, *object_name*, ***kwargs*)

DeleteObject Deletes an object.

Parameters

- **namespace_name** (*str*) – (required) The top-level namespace used for the request.
- **bucket_name** (*str*) – (required) The name of the bucket.
Example: *my-new-bucket1*
- **object_name** (*str*) – (required) The name of the object.
Example: *test/object1.log*
- **if_match** (*str*) – (optional) The entity tag to match. For creating and committing a multipart upload to an object, this is the entity tag of the target object. For uploading a part, this is the entity tag of the target part.
- **opc_client_request_id** (*str*) – (optional) The client request ID for tracing.

Returns A *Response* object with data of type None

Return type *Response*

delete_preauthenticated_request (*namespace_name*, *bucket_name*, *par_id*, ***kwargs*)

DeletePreauthenticatedRequest Deletes the bucket level pre-authenticated request

Parameters

- **namespace_name** (*str*) – (required) The top-level namespace used for the request.
- **bucket_name** (*str*) – (required) The name of the bucket.
Example: *my-new-bucket1*
- **par_id** (*str*) – (required) The unique identifier for the pre-authenticated request (PAR). This can be used to manage the PAR such as GET or DELETE the PAR
- **opc_client_request_id** (*str*) – (optional) The client request ID for tracing.

Returns A *Response* object with data of type None

Return type *Response*

get_bucket (*namespace_name*, *bucket_name*, ***kwargs*)

GetBucket Gets the current representation of the given bucket in the given namespace.

Parameters

- **namespace_name** (*str*) – (required) The top-level namespace used for the request.
- **bucket_name** (*str*) – (required) The name of the bucket.
Example: *my-new-bucket1*
- **if_match** (*str*) – (optional) The entity tag to match. For creating and committing a multipart upload to an object, this is the entity tag of the target object. For uploading a part, this is the entity tag of the target part.
- **if_none_match** (*str*) – (optional) The entity tag to avoid matching. The only valid value is '*', which indicates that the request should fail if the object already exists. For creating and committing a multipart upload, this is the entity tag of the target object. For uploading a part, this is the entity tag of the target part.
- **opc_client_request_id** (*str*) – (optional) The client request ID for tracing.

Returns A *Response* object with data of type *Bucket*

Return type *Response*

get_namespace (***kwargs*)

GetNamespace Gets the name of the namespace for the user making the request. An account name must be unique, must start with a letter, and can have up to 15 lowercase letters and numbers. You cannot use spaces or special characters.

Parameters **opc_client_request_id** (*str*) – (optional) The client request ID for tracing.

Returns A *Response* object with data of type str

Return type *Response*

get_object (*namespace_name*, *bucket_name*, *object_name*, ***kwargs*)

GetObject Gets the metadata and body of an object.

Parameters

- **namespace_name** (*str*) – (required) The top-level namespace used for the request.
- **bucket_name** (*str*) – (required) The name of the bucket.
Example: *my-new-bucket1*
- **object_name** (*str*) – (required) The name of the object.
Example: *test/object1.log*
- **if_match** (*str*) – (optional) The entity tag to match. For creating and committing a multipart upload to an object, this is the entity tag of the target object. For uploading a part, this is the entity tag of the target part.
- **if_none_match** (*str*) – (optional) The entity tag to avoid matching. The only valid value is ‘*’, which indicates that the request should fail if the object already exists. For creating and committing a multipart upload, this is the entity tag of the target object. For uploading a part, this is the entity tag of the target part.
- **opc_client_request_id** (*str*) – (optional) The client request ID for tracing.
- **range** (*str*) – (optional) Optional byte range to fetch, as described in [RFC 7233](#), section 2.1. Note, only a single range of bytes is supported.

Returns A *Response* object with data of type stream

Return type *Response*

get_preauthenticated_request (*namespace_name*, *bucket_name*, *par_id*, ****kwargs**)

GetPreauthenticatedRequest Get the bucket level pre-authenticated request

Parameters

- **namespace_name** (*str*) – (required) The top-level namespace used for the request.
- **bucket_name** (*str*) – (required) The name of the bucket.
Example: *my-new-bucket1*
- **par_id** (*str*) – (required) The unique identifier for the pre-authenticated request (PAR). This can be used to manage the PAR such as GET or DELETE the PAR
- **opc_client_request_id** (*str*) – (optional) The client request ID for tracing.

Returns A *Response* object with data of type *PreauthenticatedRequestSummary*

Return type *Response*

head_bucket (*namespace_name*, *bucket_name*, ****kwargs**)

HeadBucket Efficiently checks if a bucket exists and gets the current ETag for the bucket.

Parameters

- **namespace_name** (*str*) – (required) The top-level namespace used for the request.
- **bucket_name** (*str*) – (required) The name of the bucket.
Example: *my-new-bucket1*
- **if_match** (*str*) – (optional) The entity tag to match. For creating and committing a multipart upload to an object, this is the entity tag of the target object. For uploading a part, this is the entity tag of the target part.
- **if_none_match** (*str*) – (optional) The entity tag to avoid matching. The only valid value is ‘*’, which indicates that the request should fail if the object already exists. For creating and committing a multipart upload, this is the entity tag of the target object. For uploading a part, this is the entity tag of the target part.

- **opc_client_request_id** (*str*) – (optional) The client request ID for tracing.

Returns A *Response* object with data of type None

Return type *Response*

head_object (*namespace_name*, *bucket_name*, *object_name*, ***kwargs*)

HeadObject Gets the user-defined metadata and entity tag for an object.

Parameters

- **namespace_name** (*str*) – (required) The top-level namespace used for the request.
- **bucket_name** (*str*) – (required) The name of the bucket.
Example: *my-new-bucket1*
- **object_name** (*str*) – (required) The name of the object.
Example: *test/object1.log*
- **if_match** (*str*) – (optional) The entity tag to match. For creating and committing a multipart upload to an object, this is the entity tag of the target object. For uploading a part, this is the entity tag of the target part.
- **if_none_match** (*str*) – (optional) The entity tag to avoid matching. The only valid value is '*', which indicates that the request should fail if the object already exists. For creating and committing a multipart upload, this is the entity tag of the target object. For uploading a part, this is the entity tag of the target part.
- **opc_client_request_id** (*str*) – (optional) The client request ID for tracing.

Returns A *Response* object with data of type None

Return type *Response*

list_buckets (*namespace_name*, *compartment_id*, ***kwargs*)

ListBuckets Gets a list of all *BucketSummary*'s in a compartment. A *BucketSummary* contains only summary fields for the bucket and does not contain fields like the user-defined metadata.

To use this and other API operations, you must be authorized in an IAM policy. If you're not authorized, talk to an administrator. If you're an administrator who needs to write policies to give users access, see [Getting Started with Policies](#).

Parameters

- **namespace_name** (*str*) – (required) The top-level namespace used for the request.
- **compartment_id** (*str*) – (required) The ID of the compartment in which to create the bucket.
- **limit** (*int*) – (optional) The maximum number of items to return.
- **page** (*str*) – (optional) The page at which to start retrieving results.
- **opc_client_request_id** (*str*) – (optional) The client request ID for tracing.

Returns A *Response* object with data of type list of *BucketSummary*

Return type *Response*

list_multipart_upload_parts (*namespace_name*, *bucket_name*, *object_name*, *upload_id*, ***kwargs*)

ListMultipartUploadParts Lists the parts of an in-progress multipart upload.

Parameters

- **namespace_name** (*str*) – (required) The top-level namespace used for the request.

- **bucket_name** (*str*) – (required) The name of the bucket.
Example: *my-new-bucket1*
- **object_name** (*str*) – (required) The name of the object.
Example: *test/object1.log*
- **upload_id** (*str*) – (required) The upload ID for a multipart upload.
- **limit** (*int*) – (optional) The maximum number of items to return.
- **page** (*str*) – (optional) The page at which to start retrieving results.
- **opc_client_request_id** (*str*) – (optional) The client request ID for tracing.

Returns A *Response* object with data of type list of *MultipartUploadPartSummary*

Return type *Response*

list_multipart_uploads (*namespace_name*, *bucket_name*, ***kwargs*)

ListMultipartUploads Lists all in-progress multipart uploads for the given bucket in the given namespace.

Parameters

- **namespace_name** (*str*) – (required) The top-level namespace used for the request.
- **bucket_name** (*str*) – (required) The name of the bucket.
Example: *my-new-bucket1*
- **limit** (*int*) – (optional) The maximum number of items to return.
- **page** (*str*) – (optional) The page at which to start retrieving results.
- **opc_client_request_id** (*str*) – (optional) The client request ID for tracing.

Returns A *Response* object with data of type list of *MultipartUpload*

Return type *Response*

list_objects (*namespace_name*, *bucket_name*, ***kwargs*)

ListObjects Lists the objects in a bucket.

To use this and other API operations, you must be authorized in an IAM policy. If you're not authorized, talk to an administrator. If you're an administrator who needs to write policies to give users access, see [Getting Started with Policies](#).

Parameters

- **namespace_name** (*str*) – (required) The top-level namespace used for the request.
- **bucket_name** (*str*) – (required) The name of the bucket.
Example: *my-new-bucket1*
- **prefix** (*str*) – (optional) The string to use for matching against the start of object names in a list query.
- **start** (*str*) – (optional) Object names returned by a list query must be greater or equal to this parameter.
- **end** (*str*) – (optional) Object names returned by a list query must be strictly less than this parameter.
- **limit** (*int*) – (optional) The maximum number of items to return.

- **delimiter** (*str*) – (optional) When this parameter is set, only objects whose names do not contain the delimiter character (after an optionally specified prefix) are returned. Scanned objects whose names contain the delimiter have part of their name up to the last occurrence of the delimiter (after the optional prefix) returned as a set of prefixes. Note that only `/` is a supported delimiter character at this time.
- **fields** (*str*) – (optional) Object summary in list of objects includes the ‘name’ field. This parameter can also include ‘size’ (object size in bytes), ‘md5’, and ‘timeCreated’ (object creation date and time) fields. Value of this parameter should be a comma-separated, case-insensitive list of those field names. For example ‘name,timeCreated,md5’.
- **opc_client_request_id** (*str*) – (optional) The client request ID for tracing.

Returns A *Response* object with data of type *ListObjects*

Return type *Response*

list_preauthenticated_requests (*namespace_name*, *bucket_name*, ***kwargs*)

ListPreauthenticatedRequests List pre-authenticated requests for the bucket

Parameters

- **namespace_name** (*str*) – (required) The top-level namespace used for the request.
- **bucket_name** (*str*) – (required) The name of the bucket.
Example: *my-new-bucket1*
- **object_name_prefix** (*str*) – (optional) Pre-authenticated requests returned by the list must have object names starting with prefix
- **limit** (*int*) – (optional) The maximum number of items to return.
- **page** (*str*) – (optional) The page at which to start retrieving results.
- **opc_client_request_id** (*str*) – (optional) The client request ID for tracing.

Returns A *Response* object with data of type list of *PreauthenticatedRequestSummary*

Return type *Response*

put_object (*namespace_name*, *bucket_name*, *object_name*, *put_object_body*, ***kwargs*)

PutObject Creates a new object or overwrites an existing one.

To use this and other API operations, you must be authorized in an IAM policy. If you’re not authorized, talk to an administrator. If you’re an administrator who needs to write policies to give users access, see [Getting Started with Policies](#).

Parameters

- **namespace_name** (*str*) – (required) The top-level namespace used for the request.
- **bucket_name** (*str*) – (required) The name of the bucket.
Example: *my-new-bucket1*
- **object_name** (*str*) – (required) The name of the object.
Example: *test/object1.log*
- **put_object_body** (*stream*) – (required) The object to upload to the object store.
- **content_length** (*int*) – (optional) The content length of the body.

- **if_match** (*str*) – (optional) The entity tag to match. For creating and committing a multipart upload to an object, this is the entity tag of the target object. For uploading a part, this is the entity tag of the target part.
- **if_none_match** (*str*) – (optional) The entity tag to avoid matching. The only valid value is ‘*’, which indicates that the request should fail if the object already exists. For creating and committing a multipart upload, this is the entity tag of the target object. For uploading a part, this is the entity tag of the target part.
- **opc_client_request_id** (*str*) – (optional) The client request ID for tracing.
- **expect** (*str*) – (optional) 100-continue
- **content_md5** (*str*) – (optional) The base-64 encoded MD5 hash of the body.
- **content_type** (*str*) – (optional) The content type of the object. Defaults to ‘application/octet-stream’ if not overridden during the PutObject call.
- **content_language** (*str*) – (optional) The content language of the object.
- **content_encoding** (*str*) – (optional) The content encoding of the object.
- **str** **opc_meta** (*dict* (*str*,)) – (optional) Optional user-defined metadata key and value.

Returns A *Response* object with data of type None

Return type *Response*

update_bucket (*namespace_name*, *bucket_name*, *update_bucket_details*, ***kwargs*)

UpdateBucket Performs a partial or full update of a bucket’s user-defined metadata.

Parameters

- **namespace_name** (*str*) – (required) The top-level namespace used for the request.
- **bucket_name** (*str*) – (required) The name of the bucket.
Example: *my-new-bucket1*
- **update_bucket_details** (*UpdateBucketDetails*) – (required) Request object for updating a bucket.
- **if_match** (*str*) – (optional) The entity tag to match. For creating and committing a multipart upload to an object, this is the entity tag of the target object. For uploading a part, this is the entity tag of the target part.
- **opc_client_request_id** (*str*) – (optional) The client request ID for tracing.

Returns A *Response* object with data of type *Bucket*

Return type *Response*

upload_part (*namespace_name*, *bucket_name*, *object_name*, *upload_id*, *upload_part_num*, *upload_part_body*, ***kwargs*)

UploadPart Uploads a single part of a multipart upload.

Parameters

- **namespace_name** (*str*) – (required) The top-level namespace used for the request.
- **bucket_name** (*str*) – (required) The name of the bucket.
Example: *my-new-bucket1*
- **object_name** (*str*) – (required) The name of the object.
Example: *test/object1.log*

- **upload_id** (*str*) – (required) The upload ID for a multipart upload.
- **upload_part_num** (*int*) – (required) The part number that identifies the object part currently being uploaded.
- **upload_part_body** (*stream*) – (required) The part being uploaded to the Object Storage Service.
- **content_length** (*int*) – (optional) The content length of the body.
- **opc_client_request_id** (*str*) – (optional) The client request ID for tracing.
- **if_match** (*str*) – (optional) The entity tag to match. For creating and committing a multipart upload to an object, this is the entity tag of the target object. For uploading a part, this is the entity tag of the target part.
- **if_none_match** (*str*) – (optional) The entity tag to avoid matching. The only valid value is '*', which indicates that the request should fail if the object already exists. For creating and committing a multipart upload, this is the entity tag of the target object. For uploading a part, this is the entity tag of the target part.
- **expect** (*str*) – (optional) 100-continue
- **content_md5** (*str*) – (optional) The base-64 encoded MD5 hash of the body.

Returns A *Response* object with data of type None

Return type *Response*

Models

class `oci.object_storage.models.Bucket`

compartment_id

Gets the `compartment_id` of this `Bucket`. The compartment ID in which the bucket is authorized.

Returns The `compartment_id` of this `Bucket`.

Return type `str`

created_by

Gets the `created_by` of this `Bucket`. The OCID of the user who created the bucket.

Returns The `created_by` of this `Bucket`.

Return type `str`

etag

Gets the `etag` of this `Bucket`. The entity tag for the bucket.

Returns The `etag` of this `Bucket`.

Return type `str`

metadata

Gets the metadata of this `Bucket`. Arbitrary string keys and values for user-defined metadata.

Returns The metadata of this `Bucket`.

Return type `dict(str, str)`

name

Gets the `name` of this `Bucket`. The name of the bucket.

Returns The name of this Bucket.

Return type str

namespace

Gets the namespace of this Bucket. The namespace in which the bucket lives.

Returns The namespace of this Bucket.

Return type str

public_access_type

Gets the public_access_type of this Bucket. The type of public access available on this bucket. Allows authenticated caller to access the bucket or contents of this bucket. By default a bucket is set to NoPublicAccess. It is treated as NoPublicAccess when this value is not specified. When the type is NoPublicAccess the bucket does not allow any public access. When the type is ObjectRead the bucket allows public access to the GetObject, HeadObject, ListObjects.

Allowed values for this property are: “NoPublicAccess”, “ObjectRead”, ‘UNKNOWN_ENUM_VALUE’. Any unrecognized values returned by a service will be mapped to ‘UNKNOWN_ENUM_VALUE’.

Returns The public_access_type of this Bucket.

Return type str

time_created

Gets the time_created of this Bucket. The date and time at which the bucket was created.

Returns The time_created of this Bucket.

Return type datetime

class oci.object_storage.models.**BucketSummary**

compartment_id

Gets the compartment_id of this BucketSummary. The compartment ID in which the bucket is authorized.

Returns The compartment_id of this BucketSummary.

Return type str

created_by

Gets the created_by of this BucketSummary. The OCID of the user who created the bucket.

Returns The created_by of this BucketSummary.

Return type str

etag

Gets the etag of this BucketSummary. The entity tag for the bucket.

Returns The etag of this BucketSummary.

Return type str

name

Gets the name of this BucketSummary. The name of the bucket.

Returns The name of this BucketSummary.

Return type str

namespace

Gets the namespace of this BucketSummary. The namespace in which the bucket lives.

Returns The namespace of this BucketSummary.

Return type str

time_created

Gets the time_created of this BucketSummary. The date and time at which the bucket was created.

Returns The time_created of this BucketSummary.

Return type datetime

class oci.object_storage.models.**CommitMultipartUploadDetails**

parts_to_commit

Gets the parts_to_commit of this CommitMultipartUploadDetails. The part numbers and ETags for the parts to be committed.

Returns The parts_to_commit of this CommitMultipartUploadDetails.

Return type list[CommitMultipartUploadPartDetails]

parts_to_exclude

Gets the parts_to_exclude of this CommitMultipartUploadDetails. The part numbers for the parts to be excluded from the completed object. Each part created for this upload must be in either partsToExclude or partsToCommit, but cannot be in both.

Returns The parts_to_exclude of this CommitMultipartUploadDetails.

Return type list[int]

class oci.object_storage.models.**CommitMultipartUploadPartDetails**

etag

Gets the etag of this CommitMultipartUploadPartDetails. The ETag returned when this part was uploaded.

Returns The etag of this CommitMultipartUploadPartDetails.

Return type str

part_num

Gets the part_num of this CommitMultipartUploadPartDetails. The part number for this part.

Returns The part_num of this CommitMultipartUploadPartDetails.

Return type int

class oci.object_storage.models.**CreateBucketDetails**

compartment_id

Gets the compartment_id of this CreateBucketDetails. The ID of the compartment in which to create the bucket.

Returns The compartment_id of this CreateBucketDetails.

Return type str

metadata

Gets the metadata of this CreateBucketDetails. Arbitrary string, up to 4KB, of keys and values for user-defined metadata.

Returns The metadata of this CreateBucketDetails.

Return type dict(str, str)

name

Gets the name of this CreateBucketDetails. The name of the bucket. Valid characters are uppercase or lowercase letters, numbers, and dashes. Bucket names must be unique within the namespace.

Returns The name of this CreateBucketDetails.

Return type str

public_access_type

Gets the public_access_type of this CreateBucketDetails. The type of public access available on this bucket. Allows authenticated caller to access the bucket or contents of this bucket. By default a bucket is set to NoPublicAccess. It is treated as NoPublicAccess when this value is not specified. When the type is NoPublicAccess the bucket does not allow any public access. When the type is ObjectRead the bucket allows public access to the GetObject, HeadObject, ListObjects.

Allowed values for this property are: “NoPublicAccess”, “ObjectRead”

Returns The public_access_type of this CreateBucketDetails.

Return type str

class oci.object_storage.models.**CreateMultipartUploadDetails**

content_encoding

Gets the content_encoding of this CreateMultipartUploadDetails. the content encoding of the object to upload.

Returns The content_encoding of this CreateMultipartUploadDetails.

Return type str

content_language

Gets the content_language of this CreateMultipartUploadDetails. the content language of the object to upload.

Returns The content_language of this CreateMultipartUploadDetails.

Return type str

content_type

Gets the content_type of this CreateMultipartUploadDetails. the content type of the object to upload.

Returns The content_type of this CreateMultipartUploadDetails.

Return type str

metadata

Gets the metadata of this CreateMultipartUploadDetails. Arbitrary string keys and values for the user-defined metadata for the object. Keys must be in “opc-meta-***” format.

Returns The metadata of this CreateMultipartUploadDetails.

Return type dict(str, str)

object

Gets the object of this CreateMultipartUploadDetails. the name of the object to which this multi-part upload is targetted.

Returns The object of this CreateMultipartUploadDetails.

Return type str

class oci.object_storage.models.**CreatePreauthenticatedRequestDetails**

access_type

Gets the `access_type` of this `CreatePreauthenticatedRequestDetails`. the operation that can be performed on this resource e.g PUT or GET.

Allowed values for this property are: “ObjectRead”, “ObjectWrite”, “ObjectReadWrite”, “AnyObjectWrite”

Returns The `access_type` of this `CreatePreauthenticatedRequestDetails`.

Return type str

name

Gets the name of this `CreatePreauthenticatedRequestDetails`. user specified name for pre-authenticated request. Helpful for management purposes.

Returns The name of this `CreatePreauthenticatedRequestDetails`.

Return type str

object_name

Gets the `object_name` of this `CreatePreauthenticatedRequestDetails`. Name of object that is being granted access to by the pre-authenticated request. This can be null and that would mean that the pre-authenticated request is granting access to the entire bucket

Returns The `object_name` of this `CreatePreauthenticatedRequestDetails`.

Return type str

time_expires

Gets the `time_expires` of this `CreatePreauthenticatedRequestDetails`. The expiration date after which the pre-authenticated request will no longer be valid per spec [RFC 3339](#)

Returns The `time_expires` of this `CreatePreauthenticatedRequestDetails`.

Return type datetime

class `oci.object_storage.models.ListObjects`

next_start_with

Gets the `next_start_with` of this `ListObjects`. The name of the object to use in the ‘startWith’ parameter to obtain the next page of a truncated `ListObjects` response.

Returns The `next_start_with` of this `ListObjects`.

Return type str

objects

Gets the objects of this `ListObjects`. An array of object summaries.

Returns The objects of this `ListObjects`.

Return type list[ObjectSummary]

prefixes

Gets the prefixes of this `ListObjects`. Prefixes that are common to the results returned by the request if the request specified a delimiter.

Returns The prefixes of this `ListObjects`.

Return type list[str]

class `oci.object_storage.models.MultipartUpload`

bucket

Gets the bucket of this MultipartUpload. The bucket in which the in-progress multipart upload is stored.

Returns The bucket of this MultipartUpload.

Return type str

namespace

Gets the namespace of this MultipartUpload. The namespace in which the in-progress multipart upload is stored.

Returns The namespace of this MultipartUpload.

Return type str

object

Gets the object of this MultipartUpload. The object name of the in-progress multipart upload.

Returns The object of this MultipartUpload.

Return type str

time_created

Gets the time_created of this MultipartUpload. The date and time when the upload was created.

Returns The time_created of this MultipartUpload.

Return type datetime

upload_id

Gets the upload_id of this MultipartUpload. The unique identifier for the in-progress multipart upload.

Returns The upload_id of this MultipartUpload.

Return type str

class `oci.object_storage.models.MultipartUploadPartSummary`

etag

Gets the etag of this MultipartUploadPartSummary. the current entity tag for the part.

Returns The etag of this MultipartUploadPartSummary.

Return type str

md5

Gets the md5 of this MultipartUploadPartSummary. the MD5 hash of the bytes of the part.

Returns The md5 of this MultipartUploadPartSummary.

Return type str

part_number

Gets the part_number of this MultipartUploadPartSummary. the part number for this part.

Returns The part_number of this MultipartUploadPartSummary.

Return type int

size

Gets the size of this MultipartUploadPartSummary. the size of the part in bytes.

Returns The size of this MultipartUploadPartSummary.

Return type int

class `oci.object_storage.models.ObjectSummary`

md5

Gets the md5 of this ObjectSummary. Base64-encoded MD5 hash of the object data.

Returns The md5 of this ObjectSummary.

Return type str

name

Gets the name of this ObjectSummary. The name of the object.

Returns The name of this ObjectSummary.

Return type str

size

Gets the size of this ObjectSummary. Size of the object in bytes.

Returns The size of this ObjectSummary.

Return type int

time_created

Gets the time_created of this ObjectSummary. Date and time of object creation.

Returns The time_created of this ObjectSummary.

Return type datetime

class `oci.object_storage.models.PreauthenticatedRequest`

access_type

Gets the access_type of this PreauthenticatedRequest. the operation that can be performed on this resource e.g PUT or GET.

Allowed values for this property are: “ObjectRead”, “ObjectWrite”, “ObjectReadWrite”, “AnyObjectWrite”, ‘UNKNOWN_ENUM_VALUE’. Any unrecognized values returned by a service will be mapped to ‘UNKNOWN_ENUM_VALUE’.

Returns The access_type of this PreauthenticatedRequest.

Return type str

access_uri

Gets the access_uri of this PreauthenticatedRequest. the uri to embed in the url when using the pre-authenticated request.

Returns The access_uri of this PreauthenticatedRequest.

Return type str

id

Gets the id of this PreauthenticatedRequest. the unique identifier to use when directly addressing the pre-authenticated request

Returns The id of this PreauthenticatedRequest.

Return type str

name

Gets the name of this PreauthenticatedRequest. the user supplied name of the pre-authenticated request.

Returns The name of this PreauthenticatedRequest.

Return type str

object_name

Gets the object_name of this PreauthenticatedRequest. Name of object that is being granted access to by the pre-authenticated request. This can be null and that would mean that the pre-authenticated request is granting access to the entire bucket

Returns The object_name of this PreauthenticatedRequest.

Return type str

time_created

Gets the time_created of this PreauthenticatedRequest. the date when the pre-authenticated request was created as per spec [RFC 3339](#)

Returns The time_created of this PreauthenticatedRequest.

Return type datetime

time_expires

Gets the time_expires of this PreauthenticatedRequest. the expiration date after which the pre authenticated request will no longer be valid as per spec [RFC 3339](#)

Returns The time_expires of this PreauthenticatedRequest.

Return type datetime

class oci.object_storage.models.**PreauthenticatedRequestSummary**

access_type

Gets the access_type of this PreauthenticatedRequestSummary. the operation that can be performed on this resource e.g PUT or GET.

Allowed values for this property are: "ObjectRead", "ObjectWrite", "ObjectReadWrite", "AnyObjectWrite", 'UNKNOWN_ENUM_VALUE'. Any unrecognized values returned by a service will be mapped to 'UNKNOWN_ENUM_VALUE'.

Returns The access_type of this PreauthenticatedRequestSummary.

Return type str

id

Gets the id of this PreauthenticatedRequestSummary. the unique identifier to use when directly addressing the pre-authenticated request

Returns The id of this PreauthenticatedRequestSummary.

Return type str

name

Gets the name of this PreauthenticatedRequestSummary. the user supplied name of the pre-authenticated request

Returns The name of this PreauthenticatedRequestSummary.

Return type str

object_name

Gets the object_name of this PreauthenticatedRequestSummary. Name of object that is being granted access to by the pre-authenticated request. This can be null and that would mean that the pre-authenticated request is granting access to the entire bucket

Returns The object_name of this PreauthenticatedRequestSummary.

Return type str

time_created

Gets the time_created of this PreauthenticatedRequestSummary. the date when the pre-authenticated request was created as per spec [RFC 3339](#)

Returns The time_created of this PreauthenticatedRequestSummary.

Return type datetime

time_expires

Gets the time_expires of this PreauthenticatedRequestSummary. the expiration date after which the pre authenticated request will no longer be valid as per spec [RFC 3339](#)

Returns The time_expires of this PreauthenticatedRequestSummary.

Return type datetime

class `oci.object_storage.models.UpdateBucketDetails`

metadata

Gets the metadata of this UpdateBucketDetails. Arbitrary string, up to 4KB, of keys and values for user-defined metadata.

Returns The metadata of this UpdateBucketDetails.

Return type dict(str, str)

name

Gets the name of this UpdateBucketDetails. The name of the bucket.

Returns The name of this UpdateBucketDetails.

Return type str

namespace

Gets the namespace of this UpdateBucketDetails. The namespace in which the bucket lives.

Returns The namespace of this UpdateBucketDetails.

Return type str

public_access_type

Gets the public_access_type of this UpdateBucketDetails. The type of public access available on this bucket. Allows authenticated caller to access the bucket or contents of this bucket. By default a bucket is set to NoPublicAccess. It is treated as NoPublicAccess when this value is not specified. When the type is NoPublicAccess the bucket does not allow any public access. When the type is ObjectRead the bucket allows public access to the GetObject, HeadObject, ListObjects.

Allowed values for this property are: “NoPublicAccess”, “ObjectRead”

Returns The public_access_type of this UpdateBucketDetails.

Return type str

Base Client

class `oci.base_client.BaseClient` (*service, config, signer, type_mapping*)

call_api (*resource_path*, *method*, *path_params=None*, *query_params=None*, *header_params=None*, *body=None*, *response_type=None*, *enforce_content_headers=True*)
Makes the HTTP request and return the deserialized data.

Parameters

- **resource_path** – Path to the resource (e.g. /instance)
- **method** – HTTP method
- **path_params** – (optional) Path parameters in the url.
- **query_params** – (optional) Query parameters in the url.
- **header_params** – (optional) Request header params.
- **body** – (optional) Request body.
- **response_type** – (optional) Response data type.
- **enforce_content_headers** – (optional) Whether content headers should be added for PUT and POST requests when not present. Defaults to True.

Returns A Response object, or throw in the case of an error.

Config

oci.config.from_file (*file_location='~/oci/config'*, *profile_name='DEFAULT'*)
Create a config dict from a file.

Parameters

- **file_location** – Path to the config file. Defaults to ~/oci/config and with a fallback to ~/.oraclebmc/config.
- **profile_name** – The profile to load from the config file. Defaults to “DEFAULT”

Returns A config dict that can be used to create clients.

oci.config.validate_config (*config*)
Raises ValueError if required fields are missing or malformed.

oci.regions.is_region (*region_name*)

oci.regions.endpoint_for (*service*, *region=None*, *endpoint=None*)
Returns the base URI for a service, either in the given region or at the specified endpoint.

If endpoint and region are provided, endpoint is used.

Exceptions

exception **oci.exceptions.ClientError**
A client-side error occurred..

exception **oci.exceptions.ConfigFileNotFound**
Config file not be found.

exception **oci.exceptions.InvalidConfig** (*errors*)
The config object is missing required keys or contains malformed values.

For example:

```
raise InvalidConfig({
    "region": "missing",
    "key_id": "malformed"
})
```

exception `oci.exceptions.InvalidPrivateKey`

The provided key is not a private key, or the provided passphrase is incorrect.

exception `oci.exceptions.MaximumWaitTimeExceeded`

Maximum wait time has been exceeded.

exception `oci.exceptions.MissingPrivateKeyPassphrase`

The provided key requires a passphrase.

exception `oci.exceptions.ProfileNotFound`

The specified profile was not found in the config file.

exception `oci.exceptions.ServiceError` (*status, code, headers, message*)

The service returned an error response.

exception `oci.exceptions.WaitUntilNotSupported`

`wait_until` is not supported by this response.

Signing

`oci.signer.load_private_key_from_file` (*filename, pass_phrase=None*)

`oci.signer.load_private_key` (*secret, pass_phrase*)

Loads a private key that may use a `pass_phrase`.

Tries to correct or diagnose common errors:

- provided `pass_phrase` but didn't need one
- provided a public key

class `oci.signer.Signer` (*tenancy, user, fingerprint, private_key_file_location, pass_phrase=None*)

A requests auth instance that can be reused across requests.

You can manually sign calls by creating an instance of the signer, and providing it as the `auth` argument to Requests functions:

```
import requests
from oci import Signer

auth = Signer(...)
resp = requests.get("https://...", auth=auth)
```

Utilities

`oci.util.to_dict` (*obj*)

Helper to flatten models into dicts for rendering.

The following conversions are applied:

- `datetime.date`, `datetime.datetime`, `datetime.time` are converted into ISO8601 UTC strings

class `oci.util.Sentinel` (*name, truthy=True*)

Named singletons for clear docstrings. Also used to differentiate an explicit param of None from a lack of argument.

```
>>> missing = Sentinel("Missing", False)
>>> also_missing = Sentinel("Missing", False)
>>> assert missing is also_missing
>>> repr(missing)
<Missing>
>>> assert bool(missing) is False
```

Request

class `oci.request.Request` (*method, url, query_params=None, header_params=None, body=None, response_type=None, enforce_content_headers=True*)

Response

class `oci.response.Response` (*status, headers, data, request*)

has_next_page

Gets a value representing whether or not there is a next page of results in a list Response.

Return type bool

Contributions

Got a fix for a bug, or a new feature you'd like to contribute? The SDK is open source and accepting pull requests on [GitHub](#).

Notifications

To be notified when a new version of the Python SDK is released, subscribe to the [Atom feed](#).

License

Copyright (c) 2016, 2017, Oracle and/or its affiliates. All rights reserved.

This software is dual-licensed to you under the Universal Permissive License (UPL) and Apache License 2.0. See below for license terms. You may choose either license, or both.

UPL

The Universal Permissive License (UPL), Version 1.0 Copyright (c) 2016, 2017, Oracle and/or its affiliates. All rights reserved.

Subject to the condition set forth below, permission is hereby granted to any person obtaining a copy of this software, associated documentation and/or data (collectively the “Software”), free of charge and under any and all copyright rights in the Software, and any and all patent rights owned or freely licensable by each licensor hereunder covering either (i) the unmodified Software as contributed to or provided by such licensor, or (ii) the Larger Works (as defined below), to deal in both

1. the Software, and
2. any piece of software and/or hardware listed in the lgrwrks.txt file if one is included with the Software (each a “Larger Work” to which the Software is contributed by such licensors),

without restriction, including without limitation the rights to copy, create derivative works of, display, perform, and distribute the Software and make, use, sell, offer for sale, import, export, have made, and have sold the Software and the Larger Work(s), and to sublicense the foregoing rights on either these or other terms.

This license is subject to the following condition:

The above copyright notice and either this complete permission notice or at a minimum a reference to the UPL must be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache

The Apache Software License, Version 2.0

Copyright (c) 2016, 2017, Oracle and/or its affiliates. All rights reserved.

Licensed under the Apache License, Version 2.0 (the “License”); You may not use this product except in compliance with the License. You may obtain a copy of the License at <http://www.apache.org/licenses/LICENSE-2.0>. A copy of the license is also reproduced below. Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache License Version 2.0, January 2004 <http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

- “License” shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
- “Licensor” shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
- “Legal Entity” shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, “control” means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
- “You” (or “Your”) shall mean an individual or Legal Entity exercising permissions granted by this License.
- “Source” form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
- “Object” form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
- “Work” shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
- “Derivative Works” shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
- “Contribution” shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, “submitted” means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as “Not a Contribution.”

- “Contributor” shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. **Grant of Copyright License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
 3. **Grant of Patent License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
 4. **Redistribution.** You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - You must cause any modified files to carry prominent notices stating that You changed the files; and
 - You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - If the Work includes a “NOTICE” text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.
 - You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
 5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
 6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
 7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE.

You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets “[]” replaced with your own identifying information. (Don’t include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same “printed page” as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Questions or Feedback

Ways to get in touch:

- [GitHub](#): To file bugs and feature requests only
- [Stack Overflow](#): Please use the `oracle-cloud-infrastructure` and `oci-python-sdk` tags in your post
- [Developer Tools](#) section of the Oracle Cloud forums
- [My Oracle Support](#)

O

`oci.base_client`, 217
`oci.config`, 218
`oci.core.models`, 61
`oci.exceptions`, 218
`oci.identity.models`, 140
`oci.load_balancer.models`, 173
`oci.object_storage.models`, 209
`oci.regions`, 218
`oci.request`, 220
`oci.response`, 220
`oci.signer`, 219
`oci.util`, 219

A

- abort_multipart_upload()
 - (oci.object_storage.object_storage_client.ObjectStorageClient method), 200
- access_type (oci.object_storage.models.CreatePreauthenticatedRequestDetails attribute), 212
- access_type (oci.object_storage.models.PreauthenticatedRequest attribute), 215
- access_type (oci.object_storage.models.PreauthenticatedRequestSummary attribute), 216
- access_uri (oci.object_storage.models.PreauthenticatedRequest attribute), 215
- add_user_to_group() (oci.identity.identity_client.IdentityClient method), 124
- AddUserToGroupDetails (class in oci.identity.models), 140
- ApiKey (class in oci.identity.models), 140
- assign_public_ip (oci.core.models.CreateVnicDetails attribute), 74
- attach_vnic() (oci.core.compute_client.ComputeClient method), 26
- attach_volume() (oci.core.compute_client.ComputeClient method), 26
- AttachIScsiVolumeDetails (class in oci.core.models), 61
- attachment_type (oci.core.models.VolumeAttachment attribute), 122
- AttachVnicDetails (class in oci.core.models), 61
- AttachVolumeDetails (class in oci.core.models), 62
- availability_domain (oci.core.models.ConsoleHistory attribute), 62
- availability_domain (oci.core.models.CreateSubnetDetails attribute), 71
- availability_domain (oci.core.models.CreateVolumeDetails attribute), 76
- availability_domain (oci.core.models.Instance attribute), 92
- availability_domain (oci.core.models.LaunchInstanceDetails attribute), 96
- availability_domain (oci.core.models.PrivateIp attribute), 100
- availability_domain (oci.core.models.Subnet attribute), 104
- availability_domain (oci.core.models.Vnic attribute), 118
- availability_domain (oci.core.models.VnicAttachment attribute), 119
- availability_domain (oci.core.models.Volume attribute), 121
- availability_domain (oci.core.models.VolumeAttachment attribute), 122
- AvailabilityDomain (class in oci.identity.models), 141

B

- Backend (class in oci.load_balancer.models), 173
- backend_sets (oci.load_balancer.models.CreateLoadBalancerDetails attribute), 183
- backend_sets (oci.load_balancer.models.LoadBalancer attribute), 189
- BackendDetails (class in oci.load_balancer.models), 174
- BackendHealth (class in oci.load_balancer.models), 175
- backends (oci.load_balancer.models.BackendSet attribute), 175
- backends (oci.load_balancer.models.BackendSetDetails attribute), 176
- backends (oci.load_balancer.models.CreateBackendSetDetails attribute), 181
- backends (oci.load_balancer.models.UpdateBackendSetDetails attribute), 196
- BackendSet (class in oci.load_balancer.models), 175
- BackendSetDetails (class in oci.load_balancer.models), 176
- BackendSetHealth (class in oci.load_balancer.models), 177
- backup (oci.load_balancer.models.Backend attribute), 173
- backup (oci.load_balancer.models.BackendDetails attribute), 174
- backup (oci.load_balancer.models.CreateBackendDetails attribute), 180
- backup (oci.load_balancer.models.UpdateBackendDetails attribute), 195

bandwidth_in_mbps (oci.core.models.VirtualCircuitBandwidthShape attribute), 117

bandwidth_shape_name (oci.core.models.CreateVirtualCircuitDetails attribute), 73

bandwidth_shape_name (oci.core.models.UpdateVirtualCircuitDetails attribute), 112

bandwidth_shape_name (oci.core.models.VirtualCircuit attribute), 115

base_image_id (oci.core.models.Image attribute), 89

BaseClient (class in oci.base_client), 217

bgp_md5_auth_key (oci.core.models.CrossConnectMappingDetails attribute), 78

bgp_session_state (oci.core.models.VirtualCircuit attribute), 115

blocked (oci.identity.models.UpdateStateDetails attribute), 158

BlockstorageClient (class in oci.core.blockstorage_client), 23

Bucket (class in oci.object_storage.models), 209

bucket (oci.object_storage.models.MultipartUpload attribute), 213

bucket_name (oci.core.models.ExportImageViaObjectStorageDetails attribute), 85

bucket_name (oci.core.models.ImageSourceViaObjectStorageDetails attribute), 91

BucketSummary (class in oci.object_storage.models), 210

C

ca_certificate (oci.load_balancer.models.Certificate attribute), 178

ca_certificate (oci.load_balancer.models.CertificateDetails attribute), 179

ca_certificate (oci.load_balancer.models.CreateCertificateDetails attribute), 182

call_api() (oci.base_client.BaseClient method), 217

capture_console_history() (oci.core.compute_client.ComputeClient method), 26

CaptureConsoleHistoryDetails (class in oci.core.models), 62

Certificate (class in oci.load_balancer.models), 178

certificate_name (oci.load_balancer.models.Certificate attribute), 178

certificate_name (oci.load_balancer.models.CertificateDetails attribute), 179

certificate_name (oci.load_balancer.models.CreateCertificateDetails attribute), 182

certificate_name (oci.load_balancer.models.SSLConfigurationDetails attribute), 193

certificate_name (oci.load_balancer.models.SSLConfigurationDetails attribute), 194

CertificateDetails (class in oci.load_balancer.models), 179

certificates (oci.load_balancer.models.CreateLoadBalancerDetails attribute), 184

cidr_block (oci.load_balancer.models.LoadBalancer attribute), 190

cidr_block (oci.core.models.IScsiVolumeAttachment attribute), 88

cidr_block (oci.core.models.IScsiVolumeAttachment attribute), 89

cidr_block (oci.core.models.CreateSubnetDetails attribute), 71

cidr_block (oci.core.models.CreateVcnDetails attribute), 72

cidr_block (oci.core.models.RouteRule attribute), 102

cidr_block (oci.core.models.Subnet attribute), 104

cidr_block (oci.core.models.Vcn attribute), 114

circuit_type (oci.core.models.LetterOfAuthority attribute), 99

ClientError, 218

code (oci.core.models.IcmpOptions attribute), 89

commit_multipart_upload() (oci.object_storage.object_storage_client.ObjectStorageClient method), 200

CommitMultipartUploadDetails (class in oci.object_storage.models), 211

CommitMultipartUploadPartDetails (class in oci.object_storage.models), 211

Compartment (class in oci.identity.models), 141

compartment_id (oci.core.models.ConsoleHistory attribute), 62

compartment_id (oci.core.models.Cpe attribute), 63

compartment_id (oci.core.models.CreateCpeDetails attribute), 64

compartment_id (oci.core.models.CreateCrossConnectDetails attribute), 64

compartment_id (oci.core.models.CreateCrossConnectGroupDetails attribute), 65

compartment_id (oci.core.models.CreateDhcpDetails attribute), 66

compartment_id (oci.core.models.CreateDrgDetails attribute), 67

compartment_id (oci.core.models.CreateImageDetails attribute), 68

compartment_id (oci.core.models.CreateInternetGatewayDetails attribute), 68

compartment_id (oci.core.models.CreateIPSecConnectionDetails attribute), 67

compartment_id (oci.core.models.CreateRouteTableDetails attribute), 70

compartment_id (oci.core.models.CreateSecurityListDetails attribute), 70

compartment_id (oci.core.models.CreateSubnetDetails attribute), 71

compartment_id (oci.core.models.CreateVcnDetails attribute), 72

- compartment_id (oci.core.models.CreateVirtualCircuitDetails attribute), 73
 compartment_id (oci.core.models.CreateVolumeDetails attribute), 76
 compartment_id (oci.core.models.CrossConnect attribute), 76
 compartment_id (oci.core.models.CrossConnectGroup attribute), 77
 compartment_id (oci.core.models.DhcpOptions attribute), 81
 compartment_id (oci.core.models.Drg attribute), 82
 compartment_id (oci.core.models.DrgAttachment attribute), 83
 compartment_id (oci.core.models.Image attribute), 89
 compartment_id (oci.core.models.Instance attribute), 93
 compartment_id (oci.core.models.InstanceConsoleConnection attribute), 94
 compartment_id (oci.core.models.InternetGateway attribute), 95
 compartment_id (oci.core.models.IPSecConnection attribute), 86
 compartment_id (oci.core.models.IPSecConnectionDeviceConfig attribute), 87
 compartment_id (oci.core.models.IPSecConnectionDeviceStatus attribute), 88
 compartment_id (oci.core.models.LaunchInstanceDetails attribute), 97
 compartment_id (oci.core.models.PrivateIp attribute), 100
 compartment_id (oci.core.models.RouteTable attribute), 102
 compartment_id (oci.core.models.SecurityList attribute), 103
 compartment_id (oci.core.models.Subnet attribute), 104
 compartment_id (oci.core.models.Vcn attribute), 114
 compartment_id (oci.core.models.VirtualCircuit attribute), 115
 compartment_id (oci.core.models.Vnic attribute), 118
 compartment_id (oci.core.models.VnicAttachment attribute), 120
 compartment_id (oci.core.models.Volume attribute), 121
 compartment_id (oci.core.models.VolumeAttachment attribute), 122
 compartment_id (oci.core.models.VolumeBackup attribute), 123
 compartment_id (oci.identity.models.AvailabilityDomain attribute), 141
 compartment_id (oci.identity.models.Compartment attribute), 141
 compartment_id (oci.identity.models.CreateCompartmentDetails attribute), 142
 compartment_id (oci.identity.models.CreateGroupDetails attribute), 143
 compartment_id (oci.identity.models.CreateIdentityProviderDetails attribute), 143
 compartment_id (oci.identity.models.CreatePolicyDetails attribute), 144
 compartment_id (oci.identity.models.CreateUserDetails attribute), 146
 compartment_id (oci.identity.models.Group attribute), 148
 compartment_id (oci.identity.models.IdentityProvider attribute), 149
 compartment_id (oci.identity.models.IdpGroupMapping attribute), 150
 compartment_id (oci.identity.models.Policy attribute), 151
 compartment_id (oci.identity.models.User attribute), 159
 compartment_id (oci.identity.models.UserGroupMembership attribute), 160
 compartment_id (oci.load_balancer.models.CreateLoadBalancerDetails attribute), 184
 compartment_id (oci.load_balancer.models.LoadBalancer attribute), 190
 compartment_id (oci.object_storage.models.Bucket attribute), 209
 compartment_id (oci.object_storage.models.BucketSummary attribute), 210
 compartment_id (oci.object_storage.models.CreateBucketDetails attribute), 211
 ComputeClient (class in oci.core.compute_client), 26
 ConfigFileNotFound, 218
 connection_string (oci.core.models.InstanceConsoleConnection attribute), 95
 ConsoleHistory (class in oci.core.models), 62
 content_encoding (oci.object_storage.models.CreateMultipartUploadDetails attribute), 212
 content_language (oci.object_storage.models.CreateMultipartUploadDetails attribute), 212
 content_type (oci.object_storage.models.CreateMultipartUploadDetails attribute), 212
 cookie_name (oci.load_balancer.models.SessionPersistenceConfigurationDetails attribute), 195
 Cpe (class in oci.core.models), 63
 cpe_id (oci.core.models.CreateIPSecConnectionDetails attribute), 67
 cpe_id (oci.core.models.IPSecConnection attribute), 86
 create_backend() (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 161
 create_backend_set() (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 161
 create_bucket() (oci.object_storage.object_storage_client.ObjectStorageClient method), 201
 create_certificate() (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 162
 create_compartment() (oci.identity.identity_client.IdentityClient method), 124
 create_scope() (oci.core.virtual_network_client.VirtualNetworkClient

method), 36

create_cross_connect() (oci.core.virtual_network_client.VirtualNetworkClient method), 37

create_cross_connect_group() (oci.core.virtual_network_client.VirtualNetworkClient method), 37

create_customer_secret_key() (oci.identity.identity_client.IdentityClient method), 125

create_dhcp_options() (oci.core.virtual_network_client.VirtualNetworkClient method), 38

create_drg() (oci.core.virtual_network_client.VirtualNetworkClient method), 38

create_drg_attachment() (oci.core.virtual_network_client.VirtualNetworkClient method), 39

create_group() (oci.identity.identity_client.IdentityClient method), 125

create_identity_provider() (oci.identity.identity_client.IdentityClient method), 126

create_idp_group_mapping() (oci.identity.identity_client.IdentityClient method), 127

create_image() (oci.core.compute_client.ComputeClient method), 27

create_image_allowed (oci.core.models.Image attribute), 90

create_instance_console_connection() (oci.core.compute_client.ComputeClient method), 28

create_internet_gateway() (oci.core.virtual_network_client.VirtualNetworkClient method), 39

create_ip_sec_connection() (oci.core.virtual_network_client.VirtualNetworkClient method), 40

create_listener() (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 162

create_load_balancer() (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 162

create_multipart_upload() (oci.object_storage.object_storage_client.ObjectStorageClient method), 201

create_or_reset_ui_password() (oci.identity.identity_client.IdentityClient method), 127

create_policy() (oci.identity.identity_client.IdentityClient method), 127

create_preauthenticated_request() (oci.object_storage.object_storage_client.ObjectStorageClient method), 202

create_private_ip() (oci.core.virtual_network_client.VirtualNetworkClient method), 40

create_region_subscription() (oci.identity.identity_client.IdentityClient method), 128

create_route_table() (oci.core.virtual_network_client.VirtualNetworkClient method), 41

create_security_list() (oci.core.virtual_network_client.VirtualNetworkClient method), 41

create_subnet() (oci.core.virtual_network_client.VirtualNetworkClient method), 42

create_swift_password() (oci.identity.identity_client.IdentityClient method), 128

create_user() (oci.identity.identity_client.IdentityClient method), 129

create_vcn() (oci.core.virtual_network_client.VirtualNetworkClient method), 42

create_virtual_circuit() (oci.core.virtual_network_client.VirtualNetworkClient method), 43

create_vnic_details (oci.core.models.AttachVnicDetails attribute), 61

create_vnic_details (oci.core.models.LaunchInstanceDetails attribute), 97

create_volume() (oci.core.blockstorage_client.BlockstorageClient method), 23

create_volume_backup() (oci.core.blockstorage_client.BlockstorageClient method), 23

CreateApiKeyDetails (class in oci.identity.models), 142

CreateBackendDetails (class in oci.load_balancer.models), 180

CreateBackendSetDetails (class in oci.load_balancer.models), 181

CreateBucketDetails (class in oci.object_storage.models), 211

CreateCertificateDetails (class in oci.load_balancer.models), 182

CreateCompartmentDetails (class in oci.identity.models), 142

CreateCpeDetails (class in oci.core.models), 64

CreateCrossConnectDetails (class in oci.core.models), 64

CreateCrossConnectGroupDetails (class in oci.load_balancer.models), 65

CreateCustomerSecretKeyDetails (class in oci.identity.models), 143

created_by (oci.object_storage.models.Bucket attribute), 209

created_by (oci.object_storage.models.BucketSummary attribute), 210

CreateDhcpDetails (class in oci.core.models), 66

CreateDrgAttachmentDetails (class in oci.core.models), 66

CreateDrgDetails (class in oci.core.models), 67

CreateGroupDetails (class in oci.identity.models), 143

CreateIdentityProviderDetails (class in oci.identity.models), 143

CreateIdpGroupMappingDetails (class in oci.identity.models), 144

- CreateImageDetails (class in oci.core.models), 67
- CreateInstanceConsoleConnectionDetails (class in oci.core.models), 68
- CreateInternetGatewayDetails (class in oci.core.models), 68
- CreateIPSecConnectionDetails (class in oci.core.models), 67
- CreateListenerDetails (class in oci.load_balancer.models), 183
- CreateLoadBalancerDetails (class in oci.load_balancer.models), 183
- CreateMultipartUploadDetails (class in oci.object_storage.models), 212
- CreatePolicyDetails (class in oci.identity.models), 144
- CreatePreauthenticatedRequestDetails (class in oci.object_storage.models), 212
- CreatePrivateIpDetails (class in oci.core.models), 69
- CreateRegionSubscriptionDetails (class in oci.identity.models), 145
- CreateRouteTableDetails (class in oci.core.models), 69
- CreateSaml2IdentityProviderDetails (class in oci.identity.models), 145
- CreateSecurityListDetails (class in oci.core.models), 70
- CreateSubnetDetails (class in oci.core.models), 71
- CreateSwiftPasswordDetails (class in oci.identity.models), 145
- CreateUserDetails (class in oci.identity.models), 146
- CreateVcnDetails (class in oci.core.models), 72
- CreateVirtualCircuitDetails (class in oci.core.models), 73
- CreateVnicDetails (class in oci.core.models), 74
- CreateVolumeBackupDetails (class in oci.core.models), 75
- CreateVolumeDetails (class in oci.core.models), 76
- critical_state_backend_names (oci.load_balancer.models.BackendSetHealth attribute), 177
- critical_state_backend_set_names (oci.load_balancer.models.LoadBalancerHealth attribute), 191
- cross_connect_group_id (oci.core.models.CreateCrossConnectDetails attribute), 64
- cross_connect_group_id (oci.core.models.CrossConnect attribute), 76
- cross_connect_id (oci.core.models.CrossConnectStatus attribute), 80
- cross_connect_id (oci.core.models.LetterOfAuthority attribute), 99
- cross_connect_mappings (oci.core.models.CreateVirtualCircuitDetails attribute), 73
- cross_connect_mappings (oci.core.models.UpdateVirtualCircuitDetails attribute), 112
- cross_connect_mappings (oci.core.models.VirtualCircuit attribute), 115
- cross_connect_or_cross_connect_group_id (oci.core.models.CrossConnectMapping attribute), 79
- CrossConnect (class in oci.core.models), 76
- CrossConnectGroup (class in oci.core.models), 77
- CrossConnectLocation (class in oci.core.models), 78
- CrossConnectMapping (class in oci.core.models), 78
- CrossConnectPortSpeedShape (class in oci.core.models), 79
- CrossConnectStatus (class in oci.core.models), 80
- custom_dns_servers (oci.core.models.DhcpDnsOption attribute), 80
- customer_bgp_asn (oci.core.models.CreateVirtualCircuitDetails attribute), 73
- customer_bgp_asn (oci.core.models.UpdateVirtualCircuitDetails attribute), 112
- customer_bgp_asn (oci.core.models.VirtualCircuit attribute), 116
- customer_bgp_peering_ip (oci.core.models.CrossConnectMapping attribute), 79
- CustomerSecretKey (class in oci.identity.models), 146
- CustomerSecretKeySummary (class in oci.identity.models), 147
- ## D
- default_backend_set_name (oci.load_balancer.models.CreateListenerDetails attribute), 183
- default_backend_set_name (oci.load_balancer.models.Listener attribute), 188
- default_backend_set_name (oci.load_balancer.models.ListenerDetails attribute), 189
- default_backend_set_name (oci.load_balancer.models.UpdateListenerDetails attribute), 197
- default_dhcp_options_id (oci.core.models.Vcn attribute), 114
- default_route_table_id (oci.core.models.Vcn attribute), 114
- default_security_list_id (oci.core.models.Vcn attribute), 114
- delete_api_key() (oci.identity.identity_client.IdentityClient method), 129
- delete_backend() (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 163
- delete_backend_set() (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 164
- delete_bucket() (oci.object_storage.object_storage_client.ObjectStorageClient method), 202

delete_certificate() (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 164

delete_console_history() (oci.core.compute_client.ComputeClient method), 28

delete_cpe() (oci.core.virtual_network_client.VirtualNetworkClient method), 43

delete_cross_connect() (oci.core.virtual_network_client.VirtualNetworkClient method), 44

delete_cross_connect_group() (oci.core.virtual_network_client.VirtualNetworkClient method), 44

delete_customer_secret_key() (oci.identity.identity_client.IdentityClient method), 130

delete_dhcp_options() (oci.core.virtual_network_client.VirtualNetworkClient method), 44

delete_drg() (oci.core.virtual_network_client.VirtualNetworkClient method), 45

delete_drg_attachment() (oci.core.virtual_network_client.VirtualNetworkClient method), 45

delete_group() (oci.identity.identity_client.IdentityClient method), 130

delete_identity_provider() (oci.identity.identity_client.IdentityClient method), 130

delete_idp_group_mapping() (oci.identity.identity_client.IdentityClient method), 130

delete_image() (oci.core.compute_client.ComputeClient method), 28

delete_instance_console_connection() (oci.core.compute_client.ComputeClient method), 28

delete_internet_gateway() (oci.core.virtual_network_client.VirtualNetworkClient method), 45

delete_ip_sec_connection() (oci.core.virtual_network_client.VirtualNetworkClient method), 45

delete_listener() (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 164

delete_load_balancer() (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 165

delete_object() (oci.object_storage.object_storage_client.ObjectStorageClient method), 202

delete_policy() (oci.identity.identity_client.IdentityClient method), 131

delete_preauthenticated_request() (oci.object_storage.object_storage_client.ObjectStorageClient method), 203

delete_private_ip() (oci.core.virtual_network_client.VirtualNetworkClient method), 46

delete_route_table() (oci.core.virtual_network_client.VirtualNetworkClient method), 46

delete_subnet() (oci.core.virtual_network_client.VirtualNetworkClient method), 47

delete_swift_password() (oci.identity.identity_client.IdentityClient method), 131

delete_vcen() (oci.core.virtual_network_client.VirtualNetworkClient method), 47

delete_virtual_circuit() (oci.core.virtual_network_client.VirtualNetworkClient method), 47

delete_volume() (oci.core.blockstorage_client.BlockstorageClient method), 24

delete_volume_backup() (oci.core.blockstorage_client.BlockstorageClient method), 24

description (oci.core.models.CrossConnectLocation attribute), 78

description (oci.core.models.FastConnectProviderService attribute), 86

description (oci.identity.models.Compartment attribute), 141

description (oci.identity.models.CreateCompartmentDetails attribute), 142

description (oci.identity.models.CreateGroupDetails attribute), 143

description (oci.identity.models.CreateIdentityProviderDetails attribute), 143

description (oci.identity.models.CreatePolicyDetails attribute), 144

description (oci.identity.models.CreateSwiftPasswordDetails attribute), 145

description (oci.identity.models.CreateUserDetails attribute), 146

description (oci.identity.models.Group attribute), 148

description (oci.identity.models.IdentityProvider attribute), 149

description (oci.identity.models.Policy attribute), 152

description (oci.identity.models.SwiftPassword attribute), 155

description (oci.identity.models.Tenancy attribute), 156

description (oci.identity.models.UpdateCompartmentDetails attribute), 157

description (oci.identity.models.UpdateGroupDetails attribute), 157

description (oci.identity.models.UpdateIdentityProviderDetails attribute), 157

description (oci.identity.models.UpdatePolicyDetails attribute), 158

description (oci.identity.models.UpdateSwiftPasswordDetails attribute), 158

description (oci.identity.models.UpdateUserDetails attribute), 159

description (oci.identity.models.User attribute), 159

- destination (oci.core.models.EgressSecurityRule attribute), 84
- destination_port_range (oci.core.models.TcpOptions attribute), 106
- destination_port_range (oci.core.models.UdpOptions attribute), 108
- destination_type (oci.core.models.ExportImageDetails attribute), 85
- destination_uri (oci.core.models.ExportImageViaObjectStorageDetails attribute), 86
- detach_vnic() (oci.core.compute_client.ComputeClient method), 29
- detach_volume() (oci.core.compute_client.ComputeClient method), 29
- dhcp_options_id (oci.core.models.CreateSubnetDetails attribute), 71
- dhcp_options_id (oci.core.models.Subnet attribute), 104
- DhcpDnsOption (class in oci.core.models), 80
- DhcpOption (class in oci.core.models), 81
- DhcpOptions (class in oci.core.models), 81
- DhcpSearchDomainOption (class in oci.core.models), 82
- disable_fallback (oci.load_balancer.models.SessionPersistenceConfigurationDetails attribute), 195
- display_name (oci.core.models.AttachVnicDetails attribute), 61
- display_name (oci.core.models.AttachVolumeDetails attribute), 62
- display_name (oci.core.models.ConsoleHistory attribute), 63
- display_name (oci.core.models.Cpe attribute), 63
- display_name (oci.core.models.CreateCpeDetails attribute), 64
- display_name (oci.core.models.CreateCrossConnectDetails attribute), 65
- display_name (oci.core.models.CreateCrossConnectGroupDetails attribute), 66
- display_name (oci.core.models.CreateDhcpDetails attribute), 66
- display_name (oci.core.models.CreateDrgAttachmentDetails attribute), 66
- display_name (oci.core.models.CreateDrgDetails attribute), 67
- display_name (oci.core.models.CreateImageDetails attribute), 68
- display_name (oci.core.models.CreateInternetGatewayDetails attribute), 68
- display_name (oci.core.models.CreateIPSecConnectionDetails attribute), 67
- display_name (oci.core.models.CreatePrivateIpDetails attribute), 69
- display_name (oci.core.models.CreateRouteTableDetails attribute), 70
- display_name (oci.core.models.CreateSecurityListDetails attribute), 70
- display_name (oci.core.models.CreateSubnetDetails attribute), 71
- display_name (oci.core.models.CreateVcnDetails attribute), 72
- display_name (oci.core.models.CreateVirtualCircuitDetails attribute), 73
- display_name (oci.core.models.CreateVnicDetails attribute), 74
- display_name (oci.core.models.CreateVolumeBackupDetails attribute), 75
- display_name (oci.core.models.CreateVolumeDetails attribute), 76
- display_name (oci.core.models.CrossConnect attribute), 76
- display_name (oci.core.models.CrossConnectGroup attribute), 78
- display_name (oci.core.models.DhcpOptions attribute), 81
- display_name (oci.core.models.Drg attribute), 83
- display_name (oci.core.models.DrgAttachment attribute), 83
- display_name (oci.core.models.DrgAttachmentDetails attribute), 90
- display_name (oci.core.models.Instance attribute), 93
- display_name (oci.core.models.InternetGateway attribute), 96
- display_name (oci.core.models.IPSecConnection attribute), 87
- display_name (oci.core.models.LaunchInstanceDetails attribute), 97
- display_name (oci.core.models.PrivateIp attribute), 100
- display_name (oci.core.models.RouteTable attribute), 102
- display_name (oci.core.models.SecurityList attribute), 103
- display_name (oci.core.models.Subnet attribute), 104
- display_name (oci.core.models.UpdateCpeDetails attribute), 108
- display_name (oci.core.models.UpdateCrossConnectDetails attribute), 108
- display_name (oci.core.models.UpdateCrossConnectGroupDetails attribute), 108
- display_name (oci.core.models.UpdateDhcpDetails attribute), 109
- display_name (oci.core.models.UpdateDrgAttachmentDetails attribute), 109
- display_name (oci.core.models.UpdateDrgDetails attribute), 109
- display_name (oci.core.models.UpdateImageDetails attribute), 109
- display_name (oci.core.models.UpdateInstanceDetails attribute), 110
- display_name (oci.core.models.UpdateInternetGatewayDetails attribute), 110
- display_name (oci.core.models.UpdateIPSecConnectionDetails attribute), 110

- attribute), 109
 - display_name (oci.core.models.UpdatePrivateIpDetails attribute), 110
 - display_name (oci.core.models.UpdateRouteTableDetails attribute), 111
 - display_name (oci.core.models.UpdateSecurityListDetails attribute), 111
 - display_name (oci.core.models.UpdateSubnetDetails attribute), 111
 - display_name (oci.core.models.UpdateVcnDetails attribute), 111
 - display_name (oci.core.models.UpdateVirtualCircuitDetails attribute), 112
 - display_name (oci.core.models.UpdateVnicDetails attribute), 113
 - display_name (oci.core.models.UpdateVolumeBackupDetails attribute), 113
 - display_name (oci.core.models.UpdateVolumeDetails attribute), 113
 - display_name (oci.core.models.Vcn attribute), 114
 - display_name (oci.core.models.VirtualCircuit attribute), 116
 - display_name (oci.core.models.Vnic attribute), 118
 - display_name (oci.core.models.VnicAttachment attribute), 120
 - display_name (oci.core.models.Volume attribute), 121
 - display_name (oci.core.models.VolumeAttachment attribute), 122
 - display_name (oci.core.models.VolumeBackup attribute), 123
 - display_name (oci.identity.models.CreateCustomerSecretKeyDetails attribute), 143
 - display_name (oci.identity.models.CustomerSecretKey attribute), 146
 - display_name (oci.identity.models.CustomerSecretKeySummary attribute), 147
 - display_name (oci.identity.models.UpdateCustomerSecretKeyDetails attribute), 157
 - display_name (oci.load_balancer.models.CreateLoadBalancerDetails attribute), 184
 - display_name (oci.load_balancer.models.LoadBalancer attribute), 190
 - display_name (oci.load_balancer.models.UpdateLoadBalancerDetails attribute), 198
 - dns_label (oci.core.models.CreateSubnetDetails attribute), 71
 - dns_label (oci.core.models.CreateVcnDetails attribute), 72
 - dns_label (oci.core.models.Subnet attribute), 105
 - dns_label (oci.core.models.Vcn attribute), 114
 - drain (oci.load_balancer.models.Backend attribute), 173
 - drain (oci.load_balancer.models.BackendDetails attribute), 174
 - drain (oci.load_balancer.models.CreateBackendDetails attribute), 180
 - drain (oci.load_balancer.models.UpdateBackendDetails attribute), 195
 - Drg (class in oci.core.models), 82
 - drg_id (oci.core.models.CreateDrgAttachmentDetails attribute), 66
 - drg_id (oci.core.models.CreateIPSecConnectionDetails attribute), 67
 - drg_id (oci.core.models.DrgAttachment attribute), 83
 - drg_id (oci.core.models.IPSecConnection attribute), 87
 - DrgAttachment (class in oci.core.models), 83
- ## E
- egress_security_rules (oci.core.models.CreateSecurityListDetails attribute), 70
 - egress_security_rules (oci.core.models.SecurityList attribute), 103
 - egress_security_rules (oci.core.models.UpdateSecurityListDetails attribute), 111
 - EgressSecurityRule (class in oci.core.models), 84
 - endpoint_for() (in module oci.regions), 218
 - error_code (oci.load_balancer.models.WorkRequestError attribute), 199
 - error_details (oci.load_balancer.models.WorkRequest attribute), 198
 - etag (oci.object_storage.models.Bucket attribute), 209
 - etag (oci.object_storage.models.BucketSummary attribute), 210
 - etag (oci.object_storage.models.CommitMultipartUploadPartDetails attribute), 211
 - etag (oci.object_storage.models.MultipartUploadPartSummary attribute), 214
 - expires_on (oci.identity.models.SwiftPassword attribute), 154
 - export_image() (oci.core.compute_client.ComputeClient method), 29
 - ExportImageDetails (class in oci.core.models), 85
 - ExportImageViaObjectStorageTupleDetails (class in oci.core.models), 85
 - ExportImageViaObjectStorageUriDetails (class in oci.core.models), 86
 - extended_metadata (oci.core.models.Instance attribute), 93
 - extended_metadata (oci.core.models.LaunchInstanceDetails attribute), 97
- ## F
- facility_location (oci.core.models.LetterOfAuthority attribute), 99
 - far_cross_connect_or_cross_connect_group_id (oci.core.models.CreateCrossConnectDetails attribute), 65
 - FastConnectProviderService (class in oci.core.models), 86

- fingerprint (oci.core.models.InstanceConsoleConnection attribute), 95
- fingerprint (oci.identity.models.ApiKey attribute), 140
- from_file() (in module oci.config), 218
- ## G
- gateway_id (oci.core.models.CreateVirtualCircuitDetails attribute), 73
- gateway_id (oci.core.models.UpdateVirtualCircuitDetails attribute), 112
- gateway_id (oci.core.models.VirtualCircuit attribute), 116
- get_backend() (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 165
- get_backend_health() (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 165
- get_backend_set() (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 166
- get_backend_set_health() (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 166
- get_bucket() (oci.object_storage.object_storage_client.ObjectStorageClient method), 203
- get_compartment() (oci.identity.identity_client.IdentityClient method), 131
- get_console_history() (oci.core.compute_client.ComputeClient method), 30
- get_console_history_content() (oci.core.compute_client.ComputeClient method), 30
- get_cpe() (oci.core.virtual_network_client.VirtualNetworkClient method), 48
- get_cross_connect() (oci.core.virtual_network_client.VirtualNetworkClient method), 48
- get_cross_connect_group() (oci.core.virtual_network_client.VirtualNetworkClient method), 48
- get_cross_connect_letter_of_authority() (oci.core.virtual_network_client.VirtualNetworkClient method), 48
- get_cross_connect_status() (oci.core.virtual_network_client.VirtualNetworkClient method), 48
- get_dhcp_options() (oci.core.virtual_network_client.VirtualNetworkClient method), 48
- get_drg() (oci.core.virtual_network_client.VirtualNetworkClient method), 48
- get_drg_attachment() (oci.core.virtual_network_client.VirtualNetworkClient method), 49
- get_group() (oci.identity.identity_client.IdentityClient method), 132
- get_health_checker() (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 166
- get_identity_provider() (oci.identity.identity_client.IdentityClient method), 132
- get_idp_group_mapping() (oci.identity.identity_client.IdentityClient method), 132
- get_image() (oci.core.compute_client.ComputeClient method), 30
- get_instance() (oci.core.compute_client.ComputeClient method), 30
- get_instance_console_connection() (oci.core.compute_client.ComputeClient method), 30
- get_internet_gateway() (oci.core.virtual_network_client.VirtualNetworkClient method), 49
- get_ipsec_connection() (oci.core.virtual_network_client.VirtualNetworkClient method), 49
- get_ipsec_connection_device_config() (oci.core.virtual_network_client.VirtualNetworkClient method), 49
- get_ipsec_connection_device_status() (oci.core.virtual_network_client.VirtualNetworkClient method), 49
- get_ipsec_connection_health() (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 167
- get_ipsec_connection_health() (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 167
- get_namespace() (oci.object_storage.object_storage_client.ObjectStorageClient method), 203
- get_object() (oci.object_storage.object_storage_client.ObjectStorageClient method), 203
- get_policy() (oci.identity.identity_client.IdentityClient method), 132
- get_preauthenticated_request() (oci.object_storage.object_storage_client.ObjectStorageClient method), 204
- get_private_ip() (oci.core.virtual_network_client.VirtualNetworkClient method), 49
- get_route_table() (oci.core.virtual_network_client.VirtualNetworkClient method), 49
- get_security_list() (oci.core.virtual_network_client.VirtualNetworkClient method), 50
- get_subnet() (oci.core.virtual_network_client.VirtualNetworkClient method), 50
- get_subtype() (oci.core.models.AttachVolumeDetails static method), 62
- get_subtype() (oci.core.models.DhcpOption static method), 81
- get_subtype() (oci.core.models.ExportImageDetails static method), 85
- get_subtype() (oci.core.models.ImageSourceDetails static method), 91
- get_subtype() (oci.core.models.VolumeAttachment static method), 122

get_subtype() (oci.identity.models.CreateIdentityProviderDetails static method), 143

get_subtype() (oci.identity.models.IdentityProvider static method), 149

get_subtype() (oci.identity.models.UpdateIdentityProviderDetails static method), 157

get_tenancy() (oci.identity.identity_client.IdentityClient method), 132

get_user() (oci.identity.identity_client.IdentityClient method), 132

get_user_group_membership() (oci.identity.identity_client.IdentityClient method), 133

get_vcn() (oci.core.virtual_network_client.VirtualNetworkClient method), 50

get_virtual_circuit() (oci.core.virtual_network_client.VirtualNetworkClient method), 50

get_vnic() (oci.core.virtual_network_client.VirtualNetworkClient method), 50

get_vnic_attachment() (oci.core.compute_client.ComputeClient method), 31

get_volume() (oci.core.blockstorage_client.BlockstorageClient method), 24

get_volume_attachment() (oci.core.compute_client.ComputeClient method), 31

get_volume_backup() (oci.core.blockstorage_client.BlockstorageClient method), 24

get_windows_instance_initial_credentials() (oci.core.compute_client.ComputeClient method), 31

get_work_request() (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 167

Group (class in oci.identity.models), 148

group_id (oci.identity.models.AddUserToGroupDetails attribute), 140

group_id (oci.identity.models.CreateIdpGroupMappingDetails attribute), 144

group_id (oci.identity.models.IdpGroupMapping attribute), 151

group_id (oci.identity.models.UpdateIdpGroupMappingDetails attribute), 157

group_id (oci.identity.models.UserGroupMembership attribute), 160

H

has_next_page (oci.response.Response attribute), 220

head_bucket() (oci.object_storage.object_storage_client.ObjectStorageClient method), 204

head_object() (oci.object_storage.object_storage_client.ObjectStorageClient method), 205

health_check_results (oci.load_balancer.models.BackendSet attribute), 175

health_check_status (oci.load_balancer.models.HealthCheckResult attribute), 185

health_checker (oci.load_balancer.models.BackendSet attribute), 176

health_checker (oci.load_balancer.models.BackendSetDetails attribute), 176

health_checker (oci.load_balancer.models.CreateBackendSetDetails attribute), 181

health_checker (oci.load_balancer.models.UpdateBackendSetDetails attribute), 196

HealthChecker (class in oci.load_balancer.models), 185

HealthCheckerDetails (class in oci.load_balancer.models), 187

HealthCheckResult (class in oci.load_balancer.models), 185

hostname_label (oci.identity.models.Tenancy attribute), 155

hostname_label (oci.core.models.CreatePrivateIpDetails attribute), 69

hostname_label (oci.core.models.CreateVnicDetails attribute), 75

hostname_label (oci.core.models.LaunchInstanceDetails attribute), 97

hostname_label (oci.core.models.PrivateIp attribute), 101

hostname_label (oci.core.models.UpdatePrivateIpDetails attribute), 110

hostname_label (oci.core.models.UpdateVnicDetails attribute), 113

hostname_label (oci.core.models.Vnic attribute), 118

I

icmp_options (oci.core.models.EgressSecurityRule attribute), 84

icmp_options (oci.core.models.IngressSecurityRule attribute), 92

IcmpOptions (class in oci.core.models), 89

id (oci.core.models.ConsoleHistory attribute), 63

id (oci.core.models.Cpe attribute), 64

id (oci.core.models.CrossConnect attribute), 77

id (oci.core.models.CrossConnectGroup attribute), 78

id (oci.core.models.DhcpOptions attribute), 81

id (oci.core.models.Drg attribute), 83

id (oci.core.models.DrgAttachment attribute), 84

id (oci.core.models.Image attribute), 90

id (oci.core.models.Instance attribute), 93

id (oci.core.models.InstanceConsoleConnection attribute), 95

id (oci.core.models.InternetGateway attribute), 96

id (oci.core.models.IPSecConnection attribute), 87

id (oci.core.models.IPSecConnectionDeviceConfig attribute), 87

id (oci.core.models.IPSecConnectionDeviceStatus attribute), 88

id (oci.core.models.PrivateIp attribute), 101

- id (oci.core.models.RouteTable attribute), 102
- id (oci.core.models.SecurityList attribute), 103
- id (oci.core.models.Subnet attribute), 105
- id (oci.core.models.Vcn attribute), 114
- id (oci.core.models.VirtualCircuit attribute), 116
- id (oci.core.models.Vnic attribute), 118
- id (oci.core.models.VnicAttachment attribute), 120
- id (oci.core.models.Volume attribute), 121
- id (oci.core.models.VolumeAttachment attribute), 122
- id (oci.core.models.VolumeBackup attribute), 123
- id (oci.identity.models.Compartment attribute), 141
- id (oci.identity.models.CustomerSecretKey attribute), 146
- id (oci.identity.models.CustomerSecretKeySummary attribute), 147
- id (oci.identity.models.Group attribute), 148
- id (oci.identity.models.IdentityProvider attribute), 149
- id (oci.identity.models.IdpGroupMapping attribute), 151
- id (oci.identity.models.Policy attribute), 152
- id (oci.identity.models.SwiftPassword attribute), 154
- id (oci.identity.models.Tenancy attribute), 155
- id (oci.identity.models.User attribute), 159
- id (oci.identity.models.UserGroupMembership attribute), 160
- id (oci.load_balancer.models.LoadBalancer attribute), 190
- id (oci.load_balancer.models.WorkRequest attribute), 198
- id (oci.object_storage.models.PreauthenticatedRequest attribute), 215
- id (oci.object_storage.models.PreauthenticatedRequestSummary attribute), 216
- IdentityClient (class in oci.identity.identity_client), 124
- IdentityProvider (class in oci.identity.models), 149
- idp_group_name (oci.identity.models.CreateIdpGroupMappingDetails attribute), 144
- idp_group_name (oci.identity.models.IdpGroupMapping attribute), 151
- idp_group_name (oci.identity.models.UpdateIdpGroupMappingDetails attribute), 157
- idp_id (oci.identity.models.IdpGroupMapping attribute), 151
- IdpGroupMapping (class in oci.identity.models), 150
- Image (class in oci.core.models), 89
- image_id (oci.core.models.Instance attribute), 93
- image_id (oci.core.models.LaunchInstanceDetails attribute), 97
- image_source_details (oci.core.models.CreateImageDetails attribute), 68
- ImageSourceDetails (class in oci.core.models), 91
- ImageSourceViaObjectStorageTupleDetails (class in oci.core.models), 91
- ImageSourceViaObjectStorageUriDetails (class in oci.core.models), 91
- inactive_status (oci.identity.models.ApiKey attribute), 140
- inactive_status (oci.identity.models.Compartment attribute), 141
- inactive_status (oci.identity.models.CustomerSecretKey attribute), 146
- inactive_status (oci.identity.models.CustomerSecretKeySummary attribute), 147
- inactive_status (oci.identity.models.Group attribute), 148
- inactive_status (oci.identity.models.IdentityProvider attribute), 149
- inactive_status (oci.identity.models.IdpGroupMapping attribute), 151
- inactive_status (oci.identity.models.Policy attribute), 152
- inactive_status (oci.identity.models.SwiftPassword attribute), 154
- inactive_status (oci.identity.models.UIPassword attribute), 156
- inactive_status (oci.identity.models.User attribute), 159
- inactive_status (oci.identity.models.UserGroupMembership attribute), 160
- ingress_security_rules (oci.core.models.CreateSecurityListDetails attribute), 70
- ingress_security_rules (oci.core.models.SecurityList attribute), 103
- ingress_security_rules (oci.core.models.UpdateSecurityListDetails attribute), 111
- IngressSecurityRule (class in oci.core.models), 91
- Instance (class in oci.core.models), 92
- instance_action() (oci.core.compute_client.ComputeClient method), 31
- instance_id (oci.core.models.AttachVnicDetails attribute), 62
- instance_id (oci.core.models.AttachVolumeDetails attribute), 62
- instance_id (oci.core.models.CaptureConsoleHistoryDetails attribute), 62
- instance_id (oci.core.models.ConsoleHistory attribute), 62
- instance_id (oci.core.models.CreateImageDetails attribute), 68
- instance_id (oci.core.models.CreateInstanceConsoleConnectionDetails attribute), 68
- instance_id (oci.core.models.InstanceConsoleConnection attribute), 95
- instance_id (oci.core.models.VnicAttachment attribute), 120
- instance_id (oci.core.models.VolumeAttachment attribute), 122
- InstanceConsoleConnection (class in oci.core.models), 94
- InstanceCredentials (class in oci.core.models), 95
- interface_state (oci.core.models.CrossConnectStatus attribute), 80
- InternetGateway (class in oci.core.models), 95
- interval_in_millis (oci.load_balancer.models.HealthChecker attribute), 185

- interval_in_millis (oci.load_balancer.models.HealthCheckerDetails attribute), 187
 - interval_in_millis (oci.load_balancer.models.UpdateHealthCheckerDetails attribute), 196
 - InvalidConfig, 218
 - InvalidPrivateKey, 219
 - ip_address (oci.core.models.Cpe attribute), 64
 - ip_address (oci.core.models.CreateCpeDetails attribute), 64
 - ip_address (oci.core.models.CreatePrivateIpDetails attribute), 69
 - ip_address (oci.core.models.PrivateIp attribute), 101
 - ip_address (oci.core.models.TunnelConfig attribute), 107
 - ip_address (oci.core.models.TunnelStatus attribute), 107
 - ip_address (oci.load_balancer.models.Backend attribute), 173
 - ip_address (oci.load_balancer.models.BackendDetails attribute), 174
 - ip_address (oci.load_balancer.models.CreateBackendDetails attribute), 180
 - ip_address (oci.load_balancer.models.IpAddress attribute), 188
 - ip_addresses (oci.load_balancer.models.LoadBalancer attribute), 190
 - IpAddress (class in oci.load_balancer.models), 188
 - IPSecConnection (class in oci.core.models), 86
 - IPSecConnectionDeviceConfig (class in oci.core.models), 87
 - IPSecConnectionDeviceStatus (class in oci.core.models), 88
 - ipv4 (oci.core.models.IScsiVolumeAttachment attribute), 89
 - ipxe_script (oci.core.models.Instance attribute), 93
 - ipxe_script (oci.core.models.LaunchInstanceDetails attribute), 97
 - iqn (oci.core.models.IScsiVolumeAttachment attribute), 89
 - is_active (oci.core.models.UpdateCrossConnectDetails attribute), 108
 - is_enabled (oci.core.models.CreateInternetGatewayDetails attribute), 69
 - is_enabled (oci.core.models.InternetGateway attribute), 96
 - is_enabled (oci.core.models.UpdateInternetGatewayDetails attribute), 110
 - is_home_region (oci.identity.models.RegionSubscription attribute), 153
 - is_primary (oci.core.models.PrivateIp attribute), 101
 - is_primary (oci.core.models.Vnic attribute), 118
 - is_private (oci.load_balancer.models.CreateLoadBalancerDetails attribute), 184
 - is_private (oci.load_balancer.models.LoadBalancer attribute), 190
 - is_public (oci.load_balancer.models.IpAddress attribute), 188
 - is_region() (in module oci.regions), 218
 - is_stateless (oci.core.models.IngressSecurityRule attribute), 92
 - IScsiVolumeAttachment (class in oci.core.models), 88
- ## K
- key (oci.identity.models.CreateApiKeyDetails attribute), 142
 - key (oci.identity.models.CustomerSecretKey attribute), 146
 - key (oci.identity.models.Region attribute), 153
 - key_id (oci.identity.models.ApiKey attribute), 140
 - key_value (oci.identity.models.ApiKey attribute), 140
- ## L
- launch_instance() (oci.core.compute_client.ComputeClient method), 32
 - LaunchInstanceDetails (class in oci.core.models), 96
 - LetterOfAuthority (class in oci.core.models), 99
 - lifecycle_state (oci.core.models.ConsoleHistory attribute), 63
 - lifecycle_state (oci.core.models.CrossConnect attribute), 77
 - lifecycle_state (oci.core.models.CrossConnectGroup attribute), 78
 - lifecycle_state (oci.core.models.DhcpOptions attribute), 82
 - lifecycle_state (oci.core.models.Drg attribute), 83
 - lifecycle_state (oci.core.models.DrgAttachment attribute), 84
 - lifecycle_state (oci.core.models.Image attribute), 90
 - lifecycle_state (oci.core.models.Instance attribute), 94
 - lifecycle_state (oci.core.models.InstanceConsoleConnection attribute), 95
 - lifecycle_state (oci.core.models.InternetGateway attribute), 96
 - lifecycle_state (oci.core.models.IPSecConnection attribute), 87
 - lifecycle_state (oci.core.models.RouteTable attribute), 102
 - lifecycle_state (oci.core.models.SecurityList attribute), 103
 - lifecycle_state (oci.core.models.Subnet attribute), 105
 - lifecycle_state (oci.core.models.TunnelStatus attribute), 107
 - lifecycle_state (oci.core.models.Vcn attribute), 115
 - lifecycle_state (oci.core.models.VirtualCircuit attribute), 116
 - lifecycle_state (oci.core.models.Vnic attribute), 119
 - lifecycle_state (oci.core.models.VnicAttachment attribute), 120

- method), 134
- list_policies() (oci.load_balancer.load_balancer_client.LoadBalancerClient), 191
 - method), 169
- list_preauthenticated_requests()
 - (oci.object_storage.object_storage_client.ObjectStorageClient), 192
 - method), 207
- list_private_ips() (oci.core.virtual_network_client.VirtualNetworkClient), 199
 - method), 54
- list_protocols() (oci.load_balancer.load_balancer_client.LoadBalancerClient), 169
 - method), 169
- list_region_subscriptions()
 - (oci.identity.identity_client.IdentityClient), 135
 - method), 135
- list_regions() (oci.identity.identity_client.IdentityClient), 135
 - method), 135
- list_route_tables() (oci.core.virtual_network_client.VirtualNetworkClient), 54
 - method), 54
- list_security_lists() (oci.core.virtual_network_client.VirtualNetworkClient), 55
 - method), 55
- list_shapes() (oci.core.compute_client.ComputeClient), 34
 - method), 34
- list_shapes() (oci.load_balancer.load_balancer_client.LoadBalancerClient), 170
 - method), 170
- list_subnets() (oci.core.virtual_network_client.VirtualNetworkClient), 55
 - method), 55
- list_swift_passwords() (oci.identity.identity_client.IdentityClient), 135
 - method), 135
- list_user_group_memberships()
 - (oci.identity.identity_client.IdentityClient), 135
 - method), 135
- list_users() (oci.identity.identity_client.IdentityClient), 136
 - method), 136
- list_vcns() (oci.core.virtual_network_client.VirtualNetworkClient), 55
 - method), 55
- list_virtual_circuit_bandwidth_shapes()
 - (oci.core.virtual_network_client.VirtualNetworkClient), 55
 - method), 55
- list_virtual_circuits() (oci.core.virtual_network_client.VirtualNetworkClient), 56
 - method), 56
- list_vnic_attachments() (oci.core.compute_client.ComputeClient), 34
 - method), 34
- list_volume_attachments()
 - (oci.core.compute_client.ComputeClient), 35
 - method), 35
- list_volume_backups() (oci.core.blockstorage_client.BlockstorageClient), 25
 - method), 25
- list_volumes() (oci.core.blockstorage_client.BlockstorageClient), 25
 - method), 25
- list_work_requests() (oci.load_balancer.load_balancer_client.LoadBalancerClient), 170
 - method), 170
- Listener (class in oci.load_balancer.models), 188
- ListenerDetails (class in oci.load_balancer.models), 189
- listeners (oci.load_balancer.models.CreateLoadBalancerDetails attribute), 184
 - listeners (oci.load_balancer.models.LoadBalancer attribute), 191
 - ListObjects (class in oci.object_storage.models), 213
 - load_balancer_id (oci.load_balancer.models.LoadBalancerHealthSummary attribute), 192
 - load_balancer_id (oci.load_balancer.models.WorkRequest attribute), 199
 - load_private_key() (in module oci.signer), 219
 - load_private_key_from_file() (in module oci.signer), 219
 - LoadBalancer (class in oci.load_balancer.models), 189
 - LoadBalancerClient (class in oci.load_balancer.load_balancer_client), 161
 - LoadBalancerHealth (class in oci.load_balancer.models), 191
 - LoadBalancerHealthSummary (class in oci.load_balancer.models), 192
 - LoadBalancerPolicy (class in oci.load_balancer.models), 193
 - LoadBalancerProtocol (class in oci.load_balancer.models), 193
 - LoadBalancerShape (class in oci.load_balancer.models), 193
 - location_name (oci.core.models.CreateCrossConnectDetails attribute), 65
 - location_name (oci.core.models.CrossConnect attribute), 77

M

- mac_address (oci.core.models.Vnic attribute), 119
- max (oci.core.models.PortRange attribute), 100
- MaximumWaitTimeExceeded, 219
- multipart_upload (oci.object_storage.models.MultipartUploadPartSummary attribute), 214
- md5 (oci.object_storage.models.ObjectSummary attribute), 215
- message (oci.load_balancer.models.WorkRequest attribute), 199
- message (oci.load_balancer.models.WorkRequestError attribute), 200
- metadata (oci.core.models.Instance attribute), 94
- metadata (oci.core.models.LaunchInstanceDetails attribute), 98
- metadata (oci.identity.models.CreateSaml2IdentityProviderDetails attribute), 145
- metadata (oci.identity.models.UpdateSaml2IdentityProviderDetails attribute), 158
- metadata (oci.object_storage.models.Bucket attribute), 211
- metadata (oci.object_storage.models.CreateBucketDetails attribute), 211
- metadata (oci.object_storage.models.CreateMultipartUploadDetails attribute), 212

- metadata (oci.object_storage.models.UpdateBucketDetails attribute), 217
- metadata_url (oci.identity.models.CreateSaml2IdentityProviderDetails attribute), 145
- metadata_url (oci.identity.models.Saml2IdentityProvider attribute), 154
- metadata_url (oci.identity.models.UpdateSaml2IdentityProviderDetails attribute), 158
- min (oci.core.models.PortRange attribute), 100
- MissingPrivateKeyPassphrase, 219
- MultipartUpload (class in oci.object_storage.models), 213
- MultipartUploadPartSummary (class in oci.object_storage.models), 214
- ## N
- name (oci.core.models.CrossConnectLocation attribute), 78
- name (oci.core.models.CrossConnectPortSpeedShape attribute), 79
- name (oci.core.models.VirtualCircuitBandwidthShape attribute), 118
- name (oci.identity.models.AvailabilityDomain attribute), 141
- name (oci.identity.models.Compartment attribute), 142
- name (oci.identity.models.CreateCompartmentDetails attribute), 142
- name (oci.identity.models.CreateGroupDetails attribute), 143
- name (oci.identity.models.CreateIdentityProviderDetails attribute), 143
- name (oci.identity.models.CreatePolicyDetails attribute), 145
- name (oci.identity.models.CreateUserDetails attribute), 146
- name (oci.identity.models.Group attribute), 149
- name (oci.identity.models.IdentityProvider attribute), 150
- name (oci.identity.models.Policy attribute), 152
- name (oci.identity.models.Region attribute), 153
- name (oci.identity.models.Tenancy attribute), 155
- name (oci.identity.models.UpdateCompartmentDetails attribute), 156
- name (oci.identity.models.User attribute), 160
- name (oci.load_balancer.models.Backend attribute), 173
- name (oci.load_balancer.models.BackendSet attribute), 176
- name (oci.load_balancer.models.CreateBackendSetDetails attribute), 181
- name (oci.load_balancer.models.CreateListenerDetails attribute), 183
- name (oci.load_balancer.models.Listener attribute), 188
- name (oci.load_balancer.models.LoadBalancerPolicy attribute), 193
- name (oci.load_balancer.models.LoadBalancerProtocol attribute), 193
- name (oci.load_balancer.models.LoadBalancerShape attribute), 193
- name (oci.object_storage.models.Bucket attribute), 209
- name (oci.object_storage.models.BucketSummary attribute), 210
- name (oci.object_storage.models.CreateBucketDetails attribute), 211
- name (oci.object_storage.models.CreatePreauthenticatedRequestDetails attribute), 213
- name (oci.object_storage.models.ObjectSummary attribute), 215
- name (oci.object_storage.models.PreauthenticatedRequest attribute), 215
- name (oci.object_storage.models.PreauthenticatedRequestSummary attribute), 216
- name (oci.object_storage.models.UpdateBucketDetails attribute), 217
- namespace (oci.object_storage.models.Bucket attribute), 210
- namespace (oci.object_storage.models.BucketSummary attribute), 210
- namespace (oci.object_storage.models.MultipartUpload attribute), 214
- namespace (oci.object_storage.models.UpdateBucketDetails attribute), 217
- namespace_name (oci.core.models.ExportImageViaObjectStorageTupleDetails attribute), 85
- namespace_name (oci.core.models.ImageSourceViaObjectStorageTupleDetails attribute), 91
- near_cross_connect_or_cross_connect_group_id (oci.core.models.CreateCrossConnectDetails attribute), 65
- network_entity_id (oci.core.models.RouteRule attribute), 102
- next_start_with (oci.object_storage.models.ListObjects attribute), 213
- ## O
- object (oci.object_storage.models.CreateMultipartUploadDetails attribute), 212
- object (oci.object_storage.models.MultipartUpload attribute), 214
- object_name (oci.core.models.ExportImageViaObjectStorageTupleDetails attribute), 86
- object_name (oci.core.models.ImageSourceViaObjectStorageTupleDetails attribute), 91
- object_name (oci.object_storage.models.CreatePreauthenticatedRequestDetails attribute), 213
- object_name (oci.object_storage.models.PreauthenticatedRequest attribute), 216
- object_name (oci.object_storage.models.PreauthenticatedRequestSummary attribute), 216

- objects (oci.object_storage.models.ListObjects attribute), 213
- ObjectStorageClient (class in oci.object_storage.object_storage_client), 200
- ObjectSummary (class in oci.object_storage.models), 214
- oci.base_client (module), 217
- oci.config (module), 218
- oci.core.models (module), 61
- oci.exceptions (module), 218
- oci.identity.models (module), 140
- oci.load_balancer.models (module), 173
- oci.object_storage.models (module), 209
- oci.regions (module), 218
- oci.request (module), 220
- oci.response (module), 220
- oci.signer (module), 219
- oci.util (module), 219
- offline (oci.load_balancer.models.Backend attribute), 173
- offline (oci.load_balancer.models.BackendDetails attribute), 174
- offline (oci.load_balancer.models.CreateBackendDetails attribute), 180
- offline (oci.load_balancer.models.UpdateBackendDetails attribute), 195
- operating_system (oci.core.models.Image attribute), 90
- operating_system_version (oci.core.models.Image attribute), 90
- options (oci.core.models.CreateDhcpDetails attribute), 66
- options (oci.core.models.DhcpOptions attribute), 82
- options (oci.core.models.UpdateDhcpDetails attribute), 109
- oracle_bgp_asn (oci.core.models.VirtualCircuit attribute), 116
- oracle_bgp_peering_ip (oci.core.models.CrossConnectMapping attribute), 79
- password (oci.identity.models.UIPassword attribute), 156
- Policy (class in oci.identity.models), 151
- policy (oci.load_balancer.models.BackendSet attribute), 176
- policy (oci.load_balancer.models.BackendSetDetails attribute), 176
- policy (oci.load_balancer.models.CreateBackendSetDetails attribute), 181
- policy (oci.load_balancer.models.UpdateBackendSetDetails attribute), 196
- port (oci.core.models.IScsiVolumeAttachment attribute), 89
- port (oci.load_balancer.models.Backend attribute), 174
- port (oci.load_balancer.models.BackendDetails attribute), 175
- port (oci.load_balancer.models.CreateBackendDetails attribute), 180
- port (oci.load_balancer.models.CreateListenerDetails attribute), 183
- port (oci.load_balancer.models.HealthChecker attribute), 186
- port (oci.load_balancer.models.HealthCheckerDetails attribute), 187
- port (oci.load_balancer.models.Listener attribute), 189
- port (oci.load_balancer.models.ListenerDetails attribute), 189
- port (oci.load_balancer.models.UpdateHealthCheckerDetails attribute), 196
- port (oci.load_balancer.models.UpdateListenerDetails attribute), 198
- port_name (oci.core.models.CrossConnect attribute), 77
- port_name (oci.core.models.LetterOfAuthority attribute), 99
- port_speed_in_gbps (oci.core.models.CrossConnectPortSpeedShape attribute), 80
- port_speed_shape_name (oci.core.models.CreateCrossConnectDetails attribute), 65
- port_speed_shape_name (oci.core.models.CrossConnect attribute), 77
- PortRange (class in oci.core.models), 100
- PrauthenticatedRequest (class in oci.object_storage.models), 215
- PrauthenticatedRequestSummary (class in oci.object_storage.models), 216
- prefixes (oci.object_storage.models.ListObjects attribute), 213
- private_ip (oci.core.models.CreateVnicDetails attribute), 75
- private_ip (oci.core.models.Vnic attribute), 119
- private_key (oci.load_balancer.models.CertificateDetails attribute), 179
- private_key (oci.load_balancer.models.CreateCertificateDetails attribute), 182
- PrivateIp (class in oci.core.models), 100

- product_type (oci.identity.models.CreateIdentityProviderDetails attribute), 144
- product_type (oci.identity.models.IdentityProvider attribute), 150
- ProfileNotFound, 219
- prohibit_public_ip_on_vnic (oci.core.models.CreateSubnetDetails attribute), 71
- prohibit_public_ip_on_vnic (oci.core.models.Subnet attribute), 105
- protocol (oci.core.models.EgressSecurityRule attribute), 85
- protocol (oci.core.models.IngressSecurityRule attribute), 92
- protocol (oci.identity.models.CreateIdentityProviderDetails attribute), 144
- protocol (oci.identity.models.IdentityProvider attribute), 150
- protocol (oci.identity.models.UpdateIdentityProviderDetails attribute), 157
- protocol (oci.load_balancer.models.CreateListenerDetails attribute), 183
- protocol (oci.load_balancer.models.HealthChecker attribute), 186
- protocol (oci.load_balancer.models.HealthCheckerDetails attribute), 187
- protocol (oci.load_balancer.models.Listener attribute), 189
- protocol (oci.load_balancer.models.ListenerDetails attribute), 189
- protocol (oci.load_balancer.models.UpdateHealthCheckerDetails attribute), 197
- protocol (oci.load_balancer.models.UpdateListenerDetails attribute), 198
- provider_name (oci.core.models.CreateVirtualCircuitDetails attribute), 74
- provider_name (oci.core.models.FastConnectProviderService attribute), 86
- provider_name (oci.core.models.VirtualCircuit attribute), 116
- provider_service_name (oci.core.models.CreateVirtualCircuitDetails attribute), 74
- provider_service_name (oci.core.models.FastConnectProviderService attribute), 86
- provider_service_name (oci.core.models.VirtualCircuit attribute), 117
- provider_state (oci.core.models.UpdateVirtualCircuitDetails attribute), 112
- provider_state (oci.core.models.VirtualCircuit attribute), 117
- public_access_type (oci.object_storage.models.Bucket attribute), 210
- public_access_type (oci.object_storage.models.CreateBucketDetails attribute), 212
- public_access_type (oci.object_storage.models.UpdateBucketDetails attribute), 217
- public_certificate (oci.load_balancer.models.Certificate attribute), 178
- public_certificate (oci.load_balancer.models.CertificateDetails attribute), 180
- public_certificate (oci.load_balancer.models.CreateCertificateDetails attribute), 183
- public_ip (oci.core.models.Vnic attribute), 119
- public_key (oci.core.models.CreateInstanceConsoleConnectionDetails attribute), 68
- put_object() (oci.object_storage.object_storage_client.ObjectStorageClient method), 207
- ## R
- redirect_url (oci.identity.models.Saml2IdentityProvider attribute), 154
- reference_comment (oci.core.models.UpdateVirtualCircuitDetails attribute), 113
- reference_comment (oci.core.models.VirtualCircuit attribute), 117
- Region (class in oci.identity.models), 153
- region (oci.core.models.CreateVirtualCircuitDetails attribute), 74
- region (oci.core.models.Instance attribute), 94
- region (oci.core.models.VirtualCircuit attribute), 117
- region_key (oci.identity.models.CreateRegionSubscriptionDetails attribute), 145
- region_key (oci.identity.models.RegionSubscription attribute), 153
- region_name (oci.identity.models.RegionSubscription attribute), 153
- RegionSubscription (class in oci.identity.models), 153
- remove_user_from_group() (oci.identity.identity_client.IdentityClient method), 136
- Request (class in oci.request), 220
- Response (class in oci.response), 220
- response_body_regex (oci.load_balancer.models.HealthChecker attribute), 186
- response_body_regex (oci.load_balancer.models.HealthCheckerDetails attribute), 187
- response_body_regex (oci.load_balancer.models.UpdateHealthCheckerDetails attribute), 197
- retries (oci.load_balancer.models.HealthChecker attribute), 186
- retries (oci.load_balancer.models.HealthCheckerDetails attribute), 187
- retries (oci.load_balancer.models.UpdateHealthCheckerDetails attribute), 197
- return_code (oci.load_balancer.models.HealthChecker attribute), 186
- return_code (oci.load_balancer.models.HealthCheckerDetails attribute), 187

- return_code (oci.load_balancer.models.UpdateHealthCheckDetails attribute), 197
 - route_rules (oci.core.models.CreateRouteTableDetails attribute), 70
 - route_rules (oci.core.models.RouteTable attribute), 102
 - route_rules (oci.core.models.UpdateRouteTableDetails attribute), 111
 - route_table_id (oci.core.models.CreateSubnetDetails attribute), 72
 - route_table_id (oci.core.models.Subnet attribute), 105
 - RouteRule (class in oci.core.models), 102
 - RouteTable (class in oci.core.models), 102
- ## S
- Saml2IdentityProvider (class in oci.identity.models), 154
 - search_domain_names (oci.core.models.DhcpSearchDomainOptions attribute), 82
 - security_list_ids (oci.core.models.CreateSubnetDetails attribute), 72
 - security_list_ids (oci.core.models.Subnet attribute), 105
 - SecurityList (class in oci.core.models), 103
 - Sentinel (class in oci.util), 219
 - server_type (oci.core.models.DhcpDnsOption attribute), 81
 - ServiceError, 219
 - session_persistence_configuration (oci.load_balancer.models.BackendSet attribute), 176
 - session_persistence_configuration (oci.load_balancer.models.BackendSetDetails attribute), 177
 - session_persistence_configuration (oci.load_balancer.models.CreateBackendSetDetails attribute), 181
 - session_persistence_configuration (oci.load_balancer.models.UpdateBackendSetDetails attribute), 196
 - SessionPersistenceConfigurationDetails (class in oci.load_balancer.models), 194
 - Shape (class in oci.core.models), 104
 - shape (oci.core.models.Instance attribute), 94
 - shape (oci.core.models.LaunchInstanceDetails attribute), 99
 - shape (oci.core.models.Shape attribute), 104
 - shape_name (oci.load_balancer.models.CreateLoadBalancerDetails attribute), 184
 - shape_name (oci.load_balancer.models.LoadBalancer attribute), 191
 - shared_secret (oci.core.models.TunnelConfig attribute), 107
 - Signer (class in oci.signer), 219
 - signing_certificate (oci.identity.models.Saml2IdentityProvider attribute), 154
 - size (oci.object_storage.models.ObjectSummary attribute), 214
 - size_in_mbs (oci.core.models.CreateVolumeDetails attribute), 76
 - size_in_mbs (oci.core.models.Volume attribute), 121
 - size_in_mbs (oci.core.models.VolumeBackup attribute), 123
 - source (oci.core.models.IngressSecurityRule attribute), 92
 - source_ip_address (oci.load_balancer.models.HealthCheckResult attribute), 185
 - source_port_range (oci.core.models.TcpOptions attribute), 106
 - source_port_range (oci.core.models.UdpOptions attribute), 108
 - source_type (oci.core.models.ImageSourceDetails attribute), 91
 - source_uri (oci.core.models.ImageSourceViaObjectStorageUriDetails attribute), 91
 - ssl_configuration (oci.load_balancer.models.BackendSet attribute), 176
 - ssl_configuration (oci.load_balancer.models.BackendSetDetails attribute), 177
 - ssl_configuration (oci.load_balancer.models.CreateBackendSetDetails attribute), 181
 - ssl_configuration (oci.load_balancer.models.CreateListenerDetails attribute), 183
 - ssl_configuration (oci.load_balancer.models.Listener attribute), 189
 - ssl_configuration (oci.load_balancer.models.ListenerDetails attribute), 189
 - ssl_configuration (oci.load_balancer.models.UpdateBackendSetDetails attribute), 196
 - ssl_configuration (oci.load_balancer.models.UpdateListenerDetails attribute), 198
 - SSLConfiguration (class in oci.load_balancer.models), 193
 - SSLConfigurationDetails (class in oci.load_balancer.models), 194
 - statements (oci.identity.models.CreatePolicyDetails attribute), 145
 - statements (oci.identity.models.Policy attribute), 152
 - statements (oci.identity.models.UpdatePolicyDetails attribute), 158
 - static_routes (oci.core.models.CreateIPSecConnectionDetails attribute), 67
 - static_routes (oci.core.models.IPSecConnection attribute), 87
 - status (oci.identity.models.RegionSubscription attribute), 153
 - status (oci.load_balancer.models.BackendHealth attribute), 175

- status (oci.load_balancer.models.BackendSetHealth attribute), 177
- status (oci.load_balancer.models.LoadBalancerHealth attribute), 191
- status (oci.load_balancer.models.LoadBalancerHealthSummary attribute), 192
- Subnet (class in oci.core.models), 104
- subnet_domain_name (oci.core.models.Subnet attribute), 105
- subnet_id (oci.core.models.CreateVnicDetails attribute), 75
- subnet_id (oci.core.models.LaunchInstanceDetails attribute), 99
- subnet_id (oci.core.models.PrivateIp attribute), 101
- subnet_id (oci.core.models.Vnic attribute), 119
- subnet_id (oci.core.models.VnicAttachment attribute), 120
- subnet_id (oci.load_balancer.models.HealthCheckResult attribute), 185
- subnet_ids (oci.load_balancer.models.CreateLoadBalancerDetails attribute), 185
- subnet_ids (oci.load_balancer.models.LoadBalancer attribute), 191
- SwiftPassword (class in oci.identity.models), 154
- ## T
- tcp_options (oci.core.models.EgressSecurityRule attribute), 85
- tcp_options (oci.core.models.IngressSecurityRule attribute), 92
- TcpOptions (class in oci.core.models), 106
- Tenancy (class in oci.identity.models), 155
- terminate_instance() (oci.core.compute_client.ComputeClient method), 35
- time_accepted (oci.load_balancer.models.WorkRequest attribute), 199
- time_created (oci.core.models.ConsoleHistory attribute), 63
- time_created (oci.core.models.Cpe attribute), 64
- time_created (oci.core.models.CrossConnect attribute), 77
- time_created (oci.core.models.CrossConnectGroup attribute), 78
- time_created (oci.core.models.DhcpOptions attribute), 82
- time_created (oci.core.models.Drg attribute), 83
- time_created (oci.core.models.DrgAttachment attribute), 84
- time_created (oci.core.models.Image attribute), 90
- time_created (oci.core.models.Instance attribute), 94
- time_created (oci.core.models.InternetGateway attribute), 96
- time_created (oci.core.models.IPSecConnection attribute), 87
- time_created (oci.core.models.IPSecConnectionDeviceConfig attribute), 88
- time_created (oci.core.models.IPSecConnectionDeviceStatus attribute), 88
- time_created (oci.core.models.PrivateIp attribute), 101
- time_created (oci.core.models.RouteTable attribute), 102
- time_created (oci.core.models.SecurityList attribute), 103
- time_created (oci.core.models.Subnet attribute), 106
- time_created (oci.core.models.TunnelConfig attribute), 107
- time_created (oci.core.models.TunnelStatus attribute), 107
- time_created (oci.core.models.Vcn attribute), 115
- time_created (oci.core.models.VirtualCircuit attribute), 117
- time_created (oci.core.models.Vnic attribute), 119
- time_created (oci.core.models.VnicAttachment attribute), 120
- time_created (oci.core.models.Volume attribute), 121
- time_created (oci.core.models.VolumeAttachment attribute), 123
- time_created (oci.core.models.VolumeBackup attribute), 123
- time_created (oci.identity.models.ApiKey attribute), 141
- time_created (oci.identity.models.Compartment attribute), 142
- time_created (oci.identity.models.CustomerSecretKey attribute), 147
- time_created (oci.identity.models.CustomerSecretKeySummary attribute), 148
- time_created (oci.identity.models.Group attribute), 149
- time_created (oci.identity.models.IdentityProvider attribute), 150
- time_created (oci.identity.models.IdpGroupMapping attribute), 151
- time_created (oci.identity.models.Policy attribute), 152
- time_created (oci.identity.models.SwiftPassword attribute), 155
- time_created (oci.identity.models.UIPassword attribute), 156
- time_created (oci.identity.models.User attribute), 160
- time_created (oci.identity.models.UserGroupMembership attribute), 160
- time_created (oci.load_balancer.models.LoadBalancer attribute), 191
- time_created (oci.object_storage.models.Bucket attribute), 210
- time_created (oci.object_storage.models.BucketSummary attribute), 211
- time_created (oci.object_storage.models.MultipartUpload attribute), 214
- time_created (oci.object_storage.models.ObjectSummary attribute), 215
- time_created (oci.object_storage.models.PreauthenticatedRequest

- attribute), 216
 - time_created (oci.object_storage.models.PreauthenticatedRequestSummary attribute), 217
 - time_expires (oci.core.models.LetterOfAuthority attribute), 100
 - time_expires (oci.identity.models.CustomerSecretKey attribute), 147
 - time_expires (oci.identity.models.CustomerSecretKeySummary attribute), 148
 - time_expires (oci.object_storage.models.CreatePreauthenticatedRequestDetails attribute), 213
 - time_expires (oci.object_storage.models.PreauthenticatedRequest attribute), 216
 - time_expires (oci.object_storage.models.PreauthenticatedRequestSummary attribute), 217
 - time_finished (oci.load_balancer.models.WorkRequest attribute), 199
 - time_issued (oci.core.models.LetterOfAuthority attribute), 100
 - time_request_received (oci.core.models.VolumeBackup attribute), 124
 - time_state_modified (oci.core.models.TunnelStatus attribute), 107
 - timeout_in_millis (oci.load_balancer.models.HealthChecker attribute), 186
 - timeout_in_millis (oci.load_balancer.models.HealthCheckerDetails attribute), 188
 - timeout_in_millis (oci.load_balancer.models.UpdateHealthCheckerDetails attribute), 197
 - timestamp (oci.load_balancer.models.HealthCheckResult attribute), 185
 - to_dict() (in module oci.util), 219
 - total_backend_count (oci.load_balancer.models.BackendSetHealth attribute), 177
 - total_backend_set_count (oci.load_balancer.models.LoadBalancerHealth attribute), 192
 - TunnelConfig (class in oci.core.models), 106
 - tunnels (oci.core.models.IPSecConnectionDeviceConfig attribute), 88
 - tunnels (oci.core.models.IPSecConnectionDeviceStatus attribute), 88
 - TunnelStatus (class in oci.core.models), 107
 - type (oci.core.models.AttachVolumeDetails attribute), 62
 - type (oci.core.models.CreateVirtualCircuitDetails attribute), 74
 - type (oci.core.models.DhcpOption attribute), 81
 - type (oci.core.models.IcmpOptions attribute), 89
 - type (oci.core.models.VirtualCircuit attribute), 117
 - type (oci.load_balancer.models.WorkRequest attribute), 199
- U**
- udp_options (oci.core.models.EgressSecurityRule attribute), 85
 - udp_options (oci.core.models.IngressSecurityRule attribute), 92
 - UdpOptions (class in oci.core.models), 108
 - UIPassword (class in oci.identity.models), 156
 - unique_size_in_mbs (oci.core.models.VolumeBackup attribute), 124
 - unknown_state_backend_names (oci.load_balancer.models.BackendSetHealth attribute), 178
 - update_backend_set_names (oci.load_balancer.models.LoadBalancerHealth attribute), 192
 - update_backend() (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 170
 - update_backend_set() (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 171
 - update_bucket() (oci.object_storage.object_storage_client.ObjectStorageClient method), 208
 - update_compartment() (oci.identity.identity_client.IdentityClient method), 136
 - update_cpe() (oci.core.virtual_network_client.VirtualNetworkClient method), 56
 - update_cross_connect() (oci.core.virtual_network_client.VirtualNetworkClient method), 56
 - update_cross_connect_group() (oci.core.virtual_network_client.VirtualNetworkClient method), 57
 - update_customer_secret_key() (oci.identity.identity_client.IdentityClient method), 137
 - update_dhcp_options() (oci.core.virtual_network_client.VirtualNetworkClient method), 57
 - update_health_drg() (oci.core.virtual_network_client.VirtualNetworkClient method), 57
 - update_health_attachment() (oci.core.virtual_network_client.VirtualNetworkClient method), 58
 - update_group() (oci.identity.identity_client.IdentityClient method), 137
 - update_health_checker() (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 171
 - update_identity_provider() (oci.identity.identity_client.IdentityClient method), 137
 - update_idp_group_mapping() (oci.identity.identity_client.IdentityClient method), 137
 - update_image() (oci.core.compute_client.ComputeClient method), 35
 - update_instance() (oci.core.compute_client.ComputeClient method), 36
 - update_internet_gateway() (oci.core.virtual_network_client.VirtualNetworkClient method), 58

[update_ip_sec_connection\(\)](#) (oci.core.virtual_network_client.VirtualNetworkClient method), 58

[update_listener\(\)](#) (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 172

[update_load_balancer\(\)](#) (oci.load_balancer.load_balancer_client.LoadBalancerClient method), 172

[update_policy\(\)](#) (oci.identity.identity_client.IdentityClient method), 138

[update_private_ip\(\)](#) (oci.core.virtual_network_client.VirtualNetworkClient method), 59

[update_route_table\(\)](#) (oci.core.virtual_network_client.VirtualNetworkClient method), 59

[update_security_list\(\)](#) (oci.core.virtual_network_client.VirtualNetworkClient method), 59

[update_subnet\(\)](#) (oci.core.virtual_network_client.VirtualNetworkClient method), 60

[update_swift_password\(\)](#) (oci.identity.identity_client.IdentityClient method), 138

[update_user\(\)](#) (oci.identity.identity_client.IdentityClient method), 138

[update_user_state\(\)](#) (oci.identity.identity_client.IdentityClient method), 139

[update_vcn\(\)](#) (oci.core.virtual_network_client.VirtualNetworkClient method), 60

[update_virtual_circuit\(\)](#) (oci.core.virtual_network_client.VirtualNetworkClient method), 60

[update_vnic\(\)](#) (oci.core.virtual_network_client.VirtualNetworkClient method), 61

[update_volume\(\)](#) (oci.core.blockstorage_client.BlockstorageClient method), 25

[update_volume_backup\(\)](#) (oci.core.blockstorage_client.BlockstorageClient method), 26

[UpdateBackendDetails](#) (class in oci.load_balancer.models), 195

[UpdateBackendSetDetails](#) (class in oci.load_balancer.models), 196

[UpdateBucketDetails](#) (class in oci.object_storage.models), 217

[UpdateCompartmentDetails](#) (class in oci.identity.models), 156

[UpdateCpeDetails](#) (class in oci.core.models), 108

[UpdateCrossConnectDetails](#) (class in oci.core.models), 108

[UpdateCrossConnectGroupDetails](#) (class in oci.core.models), 108

[UpdateCustomerSecretKeyDetails](#) (class in oci.identity.models), 156

[UpdateDhcpDetails](#) (class in oci.core.models), 109

[UpdateDrgAttachmentDetails](#) (class in oci.core.models), 109

[UpdateDrgDetails](#) (class in oci.core.models), 109

[UpdateGroupDetails](#) (class in oci.identity.models), 157

[UpdateHealthCheckerDetails](#) (class in oci.load_balancer.models), 196

[UpdateIdentityProviderDetails](#) (class in oci.identity.models), 157

[UpdateIpsecVpnGatewayDetails](#) (class in oci.identity.models), 157

[UpdateImageDetails](#) (class in oci.core.models), 109

[UpdateInstanceDetails](#) (class in oci.core.models), 110

[UpdateInternetGatewayDetails](#) (class in oci.core.models), 110

[UpdateIPSecConnectionDetails](#) (class in oci.core.models), 109

[UpdateListenerDetails](#) (class in oci.load_balancer.models), 197

[UpdateLoadBalancerDetails](#) (class in oci.load_balancer.models), 198

[UpdatePolicyDetails](#) (class in oci.identity.models), 157

[UpdatePrivateIpDetails](#) (class in oci.core.models), 110

[UpdateRouteTableDetails](#) (class in oci.core.models), 110

[UpdateSaml2IdentityProviderDetails](#) (class in oci.identity.models), 158

[UpdateSecurityListDetails](#) (class in oci.core.models), 111

[UpdateStateDetails](#) (class in oci.identity.models), 158

[UpdateSubnetDetails](#) (class in oci.core.models), 111

[UpdateSwiftPasswordDetails](#) (class in oci.identity.models), 158

[UpdateUserDetails](#) (class in oci.identity.models), 159

[UpdateVcnDetails](#) (class in oci.core.models), 111

[UpdateVirtualCircuitDetails](#) (class in oci.core.models), 111

[UpdateVnicDetails](#) (class in oci.core.models), 113

[UpdateVolumeBackupDetails](#) (class in oci.core.models), 113

[UpdateVolumeDetails](#) (class in oci.core.models), 113

[upload_api_key\(\)](#) (oci.identity.identity_client.IdentityClient method), 139

[upload_id](#) (oci.object_storage.models.MultipartUpload attribute), 214

[upload_part\(\)](#) (oci.object_storage.object_storage_client.ObjectStorageClient method), 208

[url_path](#) (oci.load_balancer.models.HealthChecker attribute), 187

[url_path](#) (oci.load_balancer.models.HealthCheckerDetails attribute), 188

[url_path](#) (oci.load_balancer.models.UpdateHealthCheckerDetails attribute), 197

[use_chap](#) (oci.core.models.AttachIScsiVolumeDetails attribute), 61

[User](#) (class in oci.identity.models), 159

[user_id](#) (oci.identity.models.AddUserToGroupDetails attribute), 140

[user_id](#) (oci.identity.models.ApiKey attribute), 141

[user_id](#) (oci.identity.models.CustomerSecretKey attribute), 141

tribute), 147
 user_id (oci.identity.models.CustomerSecretKeySummary attribute), 148
 user_id (oci.identity.models.SwiftPassword attribute), 155
 user_id (oci.identity.models.UIPassword attribute), 156
 user_id (oci.identity.models.UserGroupMembership attribute), 161
 UserGroupMembership (class in oci.identity.models), 160
 username (oci.core.models.InstanceCredentials attribute), 95

V

validate_config() (in module oci.config), 218
 Vcn (class in oci.core.models), 113
 vcn_domain_name (oci.core.models.Vcn attribute), 115
 vcn_id (oci.core.models.CreateDhcpDetails attribute), 66
 vcn_id (oci.core.models.CreateDrgAttachmentDetails attribute), 66
 vcn_id (oci.core.models.CreateInternetGatewayDetails attribute), 69
 vcn_id (oci.core.models.CreateRouteTableDetails attribute), 70
 vcn_id (oci.core.models.CreateSecurityListDetails attribute), 70
 vcn_id (oci.core.models.CreateSubnetDetails attribute), 72
 vcn_id (oci.core.models.DhcpOptions attribute), 82
 vcn_id (oci.core.models.DrgAttachment attribute), 84
 vcn_id (oci.core.models.InternetGateway attribute), 96
 vcn_id (oci.core.models.RouteTable attribute), 103
 vcn_id (oci.core.models.SecurityList attribute), 104
 vcn_id (oci.core.models.Subnet attribute), 106
 verify_depth (oci.load_balancer.models.SSLConfiguration attribute), 194
 verify_depth (oci.load_balancer.models.SSLConfigurationDetails attribute), 194
 verify_peer_certificate (oci.load_balancer.models.SSLConfiguration attribute), 194
 verify_peer_certificate (oci.load_balancer.models.SSLConfigurationDetails attribute), 194
 version_date (oci.identity.models.CreatePolicyDetails attribute), 145
 version_date (oci.identity.models.Policy attribute), 152
 version_date (oci.identity.models.UpdatePolicyDetails attribute), 158
 virtual_router_ip (oci.core.models.Subnet attribute), 106
 virtual_router_mac (oci.core.models.Subnet attribute), 106
 VirtualCircuit (class in oci.core.models), 115
 VirtualCircuitBandwidthShape (class in oci.core.models), 117

VirtualNetworkClient (class in oci.core.virtual_network_client), 36
 vlan (oci.core.models.CrossConnectMapping attribute), 79
 vlan_tag (oci.core.models.VnicAttachment attribute), 120
 Vnic (class in oci.core.models), 118
 vnic_id (oci.core.models.CreatePrivateIpDetails attribute), 69
 vnic_id (oci.core.models.PrivateIp attribute), 101
 vnic_id (oci.core.models.UpdatePrivateIpDetails attribute), 110
 vnic_id (oci.core.models.VnicAttachment attribute), 121
 VnicAttachment (class in oci.core.models), 119
 Volume (class in oci.core.models), 121
 volume_backup_id (oci.core.models.CreateVolumeDetails attribute), 76
 volume_id (oci.core.models.AttachVolumeDetails attribute), 62
 volume_id (oci.core.models.CreateVolumeBackupDetails attribute), 75
 volume_id (oci.core.models.VolumeAttachment attribute), 123
 volume_id (oci.core.models.VolumeBackup attribute), 124
 VolumeAttachment (class in oci.core.models), 122
 VolumeBackup (class in oci.core.models), 123

W

WaitUntilNotSupported, 219
 warning_state_backend_names (oci.load_balancer.models.BackendSetHealth attribute), 178
 warning_state_backend_set_names (oci.load_balancer.models.LoadBalancerHealth attribute), 192
 weight (oci.load_balancer.models.Backend attribute), 174
 weight (oci.load_balancer.models.BackendDetails attribute), 175
 weight (oci.load_balancer.models.CreateBackendDetails attribute), 181
 weight (oci.load_balancer.models.UpdateBackendDetails attribute), 195
 WorkRequest (class in oci.load_balancer.models), 198
 WorkRequestError (class in oci.load_balancer.models), 199