
hydra Documentation

Release 0.1

sandeep

Feb 18, 2018

Contents

1	hydrus package	1
1.1	Submodules	1
1.2	hydrus.app module	1
1.3	hydrus.utils module	3
1.4	Module contents	3
2	hydrus.data package	5
2.1	Submodules	5
2.2	hydrus.data.crud module	5
2.3	hydrus.data.db_models module	6
2.4	hydrus.data.doc_parse module	8
2.5	hydrus.data.exceptions module	8
2.6	hydrus.data.user module	9
2.7	Module contents	10
3	hydrus.data.legacy package	11
3.1	hydrus.data.legacy.generator module	11
3.2	hydrus.data.legacy.insert_data module	11
3.3	hydrus.data.legacy.insert_classes module	11
4	hydrus.hydraspec package	13
4.1	Submodules	13
4.2	hydrus.hydraspec.doc_maker module	13
4.3	hydrus.hydraspec.doc_writer module	14
4.4	hydrus.hydraspec.doc_writer_sample module	16
4.5	hydrus.hydraspec.doc_writer_sample_output module	16
4.6	Module contents	19
5	hydrus.hydraspec.legacy package	21
5.1	hydrus.hydraspec.legacy.crud_template module	21
5.2	hydrus.hydraspec.legacy.parser module	21
6	Indices and tables	23
	Python Module Index	25

1.1 Submodules

1.2 hydrus.app module

Main route for the application.

```
class hydrus.app.Contexts
    Bases: flask_restful.Resource
    Dynamically generated contexts.
    get (category)
        Return the context for the specified class.
    methods = ['GET']
```

```
class hydrus.app.EntryPoint
    Bases: flask_restful.Resource
    Hydra EntryPoint.
    get ()
        Return application main Entrypoint.
    methods = ['GET']
```

```
class hydrus.app.Index
    Bases: flask_restful.Resource
    Class for the EntryPoint.
    get ()
        Return main entypoint for the api.
    methods = ['GET']
```

class `hydrus.app.Item`

Bases: `flask_restful.Resource`

Handles all operations(GET, POST, PATCH, DELETE) on Items (item can be anything depending upon the vocabulary).

delete (*id_*, *type_*)

Delete object with `id=id_` from database.

get (*id_*, *type_*)

GET object with `id = id_` from the database.

methods = ['DELETE', 'GET', 'POST', 'PUT']

post (*id_*, *type_*)

Update object of type<*type_*> at ID<*id_*> with new **object_** using HTTP POST.

put (*id_*, *type_*)

Add new **object_** optional <*id_*> parameter using HTTP PUT.

class `hydrus.app.ItemCollection`

Bases: `flask_restful.Resource`

Handle operation related to ItemCollection (a collection of items).

delete (*type_*)

Delete a non Collection class item.

get (*type_*)

Retrieve a collection of items from the database.

methods = ['DELETE', 'GET', 'POST', 'PUT']

post (*type_*)

Update Non Collection class item.

put (*type_*)

Add item to ItemCollection.

class `hydrus.app.Vocab`

Bases: `flask_restful.Resource`

Vocabulary for Hydra.

get ()

Return the main hydra vocab.

methods = ['GET']

`hydrus.app.app_factory` (*API_NAME='api'*)

Create an app object.

`hydrus.app.checkClassOp` (*class_type*, *method*)

Check if the Class supports the operation.

`hydrus.app.checkEndpoint` (*method*, *type_*)

Check if endpoint and method is supported in the API.

`hydrus.app.failed_authentication` ()

Return failed authentication object.

`hydrus.app.getType` (*class_type*, *method*)

Return the @type of object allowed for POST/PUT.

`hydrus.app.hydrafy(object_)`
Add hydra context to objects.

`hydrus.app.set_response_headers(resp, ct='application/ld+json', headers=[], status_code=200)`
Set the response headers.

`hydrus.app.validObject(object_)`
Check if the data passed in POST is of valid format or not.

1.3 hydrus.utils module

Pluggable utilities for Hydrus.

`hydrus.utils.get_api_name()`
Get the server API name.

`hydrus.utils.get_authentication()`
Check whether API needs to be authenticated or not.

`hydrus.utils.get_doc()`
Get the server API Documentation.

`hydrus.utils.get_hydrus_server_url()`
Get the server URL.

`hydrus.utils.get_session()`
Get the Database Session for the server.

`hydrus.utils.set_api_name(*args, **kws)`
Set the server name or EntryPoint for the app. Must be of type <str>.

`hydrus.utils.set_authentication(*args, **kws)`
Set the whether API needs to be authenticated or not.

`hydrus.utils.set_doc(*args, **kws)`
Set the API Documentation for the app. Must be of type <hydrus.hydraspec.doc_writer.HydraDoc>.

`hydrus.utils.set_hydrus_server_url(*args, **kws)`
Set the server URL for the app. Must be of type <str>.

`hydrus.utils.set_session(*args, **kws)`
Set the database session for the app. Must be of type <hydrus.hydraspec.doc_writer.HydraDoc>.

1.4 Module contents

2.1 Submodules

2.2 hydrus.data.crud module

Basic CRUD operations for the server.

`hydrus.data.crud.delete` (*id_*, *type_*, *session*)

Delete an Instance and all its relations from DB given id [DELETE].

`hydrus.data.crud.delete_single` (*type_*, *session*)

Delete instance of classes with single objects.

`hydrus.data.crud.get` (*id_*, *type_*, *api_name*, *session*, *recursive=False*)

Retrieve an Instance with given ID from the database [GET].

`hydrus.data.crud.get_collection` (*API_NAME*, *type_*, *session*)

Retrieve a type of collection from the database.

`hydrus.data.crud.get_single` (*type_*, *api_name*, *session*)

Get instance of classes with single objects.

`hydrus.data.crud.insert` (*object_*, *session*, *id_=None*)

Insert an object to database [POST] and returns the inserted object.

`hydrus.data.crud.insert_single` (*object_*, *session*)

Insert instance of classes with single objects.

`hydrus.data.crud.update` (*id_*, *type_*, *object_*, *session*, *api_name*)

Update an object properties based on the given object [PUT].

`hydrus.data.crud.update_single` (*object_*, *session*, *api_name*)

Update instance of classes with single objects.

2.3 hydrus.data.db_models module

Models for Hydra Classes.

```
class hydrus.data.db_models.AbstractProperty (**kwargs)
    Bases: hydrus.data.db_models.BaseProperty
```

Model for Abstract Properties.

Abstract Properties are properties that are used as predicate between two RDF-OWL classes. >>> prop1 = BaseProperty('hasWeight') >>> prop2 = BaseProperty('hasCost') Example of a triple:

```
RDFClass('A') Property('isSubSystemOf') RDFClass('B')
```

id

name

type_

```
class hydrus.data.db_models.BaseProperty (**kwargs)
    Bases: sqlalchemy.ext.declarative.api.Base
```

Model for Basic Property.

id

name

type_

```
class hydrus.data.db_models.Graph (**kwargs)
    Bases: sqlalchemy.ext.declarative.api.Base
```

Model for a graph that store triples of instance from the other models to map relationships.

id

type

```
class hydrus.data.db_models.GraphCAC (**kwargs)
    Bases: hydrus.data.db_models.Graph
```

Graph model for Class >> AbstractProperty >> Class.

id

object_

predicate

subject

type

```
class hydrus.data.db_models.GraphIAC (**kwargs)
    Bases: hydrus.data.db_models.Graph
```

Graph model for Instance >> AbstractProperty >> Class.

id

object_

predicate

subject

type

class `hydrus.data.db_models.GraphIII` (**kwargs)
 Bases: `hydrus.data.db_models.Graph`

Graph model for Instance >> InstanceProperty >> Instance.

id**object_****predicate****subject****type**

class `hydrus.data.db_models.GraphIIT` (**kwargs)
 Bases: `hydrus.data.db_models.Graph`

Graph model for Instance >> InstanceProperty >> Terminal.

id**object_****predicate****subject****type**

class `hydrus.data.db_models.Instance` (**kwargs)
 Bases: `sqlalchemy.ext.declarative.api.Base`

Model for Object/Resource.

Instances are instances of some kind/classes that are served through the API.

id**type_**

class `hydrus.data.db_models.InstanceProperty` (**kwargs)
 Bases: `hydrus.data.db_models.BaseProperty`

Model for Instance Properties.

Instance Properties are properties that are used as predicate when the subject is an Instance. >>> prop1 = BaseProperty('hasWeight') >>> prop2 = BaseProperty('hasCost')

id**name****type_**

class `hydrus.data.db_models.RDFClass` (**kwargs)
 Bases: `sqlalchemy.ext.declarative.api.Base`

Model for Classes.

Classes are RDF-OWL or RDF-HYDRA classes.

id**name**

```
class hydrus.data.db_models.Terminal (**kwargs)
    Bases: sqlalchemy.ext.declarative.api.Base

    Model for Terminals.
```

Terminals are numbers or string that can be referenced by a Property. They can be only objects in a triple.

```
>>> t = Terminal(value=85, unit='cubic centimeters')
>>> t1 = Terminal(value='Cubesat', unit='standard')
```

```
id
unit
value
```

```
class hydrus.data.db_models.User (**kwargs)
    Bases: sqlalchemy.ext.declarative.api.Base

    Model for a user that stores the ID, paraphrase and a nonce.
```

```
id
paraphrase
```

2.4 hydrus.data.doc_parse module

Parser for Hydra APIDocumentation creates Classes and Properties.

```
hydrus.data.doc_parse.get_all_properties (classes)
    Get all the classes in the APIDocumentation.
```

```
hydrus.data.doc_parse.get_classes (apidoc)
    Get all the classes in the APIDocumentation.
```

```
hydrus.data.doc_parse.insert_classes (classes, session)
    Insert all the classes as defined in the APIDocumentation into DB.
```

```
hydrus.data.doc_parse.insert_properties (properties, session)
    Insert all the properties as defined in the APIDocumentation into DB.
```

2.5 hydrus.data.exceptions module

Exceptions for the crud operations.

```
exception hydrus.data.exceptions.ClassNotFound (type_)
    Bases: exceptions.Exception

    Error when the RDFClass is not found.
```

```
get_HTTP ()
    Return the HTTP response for the Exception.
```

```
exception hydrus.data.exceptions.InstanceExists (type_, id_=None)
    Bases: exceptions.Exception

    Error when the Instance already exists.
```

get_HTTP ()
Return the HTTP response for the Exception.

exception `hydrus.data.exceptions.InstanceNotFound (type_, id_=None)`
Bases: `exceptions.Exception`
Error when the Instance is not found.

get_HTTP ()
Return the HTTP response for the Exception.

exception `hydrus.data.exceptions.NotAbstractProperty (type_)`
Bases: `exceptions.Exception`
Error when the property is not an Abstract property.

get_HTTP ()
Return the HTTP response for the Exception.

exception `hydrus.data.exceptions.NotInstanceProperty (type_)`
Bases: `exceptions.Exception`
Error when the property is not an Instance property.

get_HTTP ()
Return the HTTP response for the Exception.

exception `hydrus.data.exceptions.PropertyNotFound (type_)`
Bases: `exceptions.Exception`
Error when the Property is not found.

get_HTTP ()
Return the HTTP response for the Exception.

exception `hydrus.data.exceptions.UserExists (id_)`
Bases: `exceptions.Exception`
Error when the User already existst.

get_HTTP ()
Return the HTTP response for the Exception.

exception `hydrus.data.exceptions.UserNotFound (id_)`
Bases: `exceptions.Exception`
Error when the User is not found.

get_HTTP ()
Return the HTTP response for the Exception.

2.6 hydrus.data.user module

File operations for User authorization.

`hydrus.data.user.add_user (id_, paraphrase, session)`
Add new users to the database.

`hydrus.data.user.authenticate_user (id_, hashvalue, session)`
Authenticate a user based on the ID and his paraphrase.

`hydrus.data.user.check_authorization (request, session)`
Check if the request object has the correct authorization.

`hydrus.data.user.generate_basic_digest` (*id_*, *paraphrase*)
Create the digest to be added to the HTTP Authorization header.

2.7 Module contents

hydrus.data.legacy package

3.1 hydrus.data.legacy.generator module

3.2 hydrus.data.legacy.insert_data module

3.3 hydrus.data.legacy.insert_classes module

4.1 Submodules

4.2 hydrus.hydraspec.doc_maker module

Constructor to take a Python dict containing an API Documentation and create a HydraDoc object for it.

`hydrus.hydraspec.doc_maker.class_in_endpoint` (*class_*, *entrypoint*)

Check if a given class is in the EntryPoint object as a class.

`hydrus.hydraspec.doc_maker.collection_in_endpoint` (*class_*, *entrypoint*)

Check if a given class is in the EntryPoint object as a collection.

`hydrus.hydraspec.doc_maker.convert_literal` (*literal*)

Convert JSON literals to Python ones.

`hydrus.hydraspec.doc_maker.createClass` (*entrypoint*, *class_dict*)

Create HydraClass objects for classes in the API Documentation.

`hydrus.hydraspec.doc_maker.createDoc` (*doc*, *HYDRUS_SERVER_URL=None*,
API_NAME=None)

Create the HydraDoc object from the API Documentation.

`hydrus.hydraspec.doc_maker.createOperation` (*supported_op*)

Create a HydraClassOp object from the supportedOperation.

`hydrus.hydraspec.doc_maker.createProperty` (*supported_prop*)

Create a HydraClassProp object from the supportedProperty.

`hydrus.hydraspec.doc_maker.createStatus` (*possible_status*)

Create a HydraStatus object from the possibleStatus.

`hydrus.hydraspec.doc_maker.getEntryPoint` (*doc*)

Find and return the endpoint object in the doc.

4.3 hydrus.hydraspec.doc_writer module

API Doc templates generator.

class hydrus.hydraspec.doc_writer.**Context** (*address, adders={}, class_=None, collection=None, endpoint=None*)

Class for JSON-LD context.

add (*key, value*)

Add entry to context.

createContext (*object_*)

Create the context for the given object.

generate ()

Get as a python dict.

class hydrus.hydraspec.doc_writer.**EntryPointClass** (*class_*)

Class for a Operation Entry to the EntryPoint object.

generate ()

Get as Python Dict.

class hydrus.hydraspec.doc_writer.**EntryPointCollection** (*collection*)

Class for a Collection Entry to the EntryPoint object.

generate ()

Get as a python dict.

class hydrus.hydraspec.doc_writer.**EntryPointOp** (*id_, method, desc, expects, returns, statusCodes=[], type_=None, label=""*)

supportedOperation for EntryPoint.

generate ()

Get as Python Dict.

get_type (*method*)

Return @type for op based on method type.

class hydrus.hydraspec.doc_writer.**HydraClass** (*id_, title, desc, endpoint=False, subclassof=None*)

Template for a new class.

add_supported_op (*op*)

Add a new supportedOperation.

add_supported_prop (*prop*)

Add a new supportedProperty.

generate ()

Get the Hydra class as a python dict.

class hydrus.hydraspec.doc_writer.**HydraClassOp** (*title, method, expects, returns, status*)

Template for a new supportedOperation.

generate ()

Get the Hydra op as a python dict.

get_type (*method*)

Return @type for op based on method type.

class hydrus.hydraspec.doc_writer.**HydraClassProp** (*prop, title, read, write, required, desc=""*)

Template for a new property.

```

generate ()
    Get the Hydra prop as a python dict.

class hydrus.hydraspec.doc_writer.HydraCollection (class_, get=True, post=True)
    Class for Hydra Collection.

generate ()
    Get as a python dict.

class hydrus.hydraspec.doc_writer.HydraCollectionOp (id_, type_, method, desc, expects,
                                                    returns, status=[])
    Operation class for Collection operations.

generate ()
    Get as a Python dict.

class hydrus.hydraspec.doc_writer.HydraDoc (API, title, desc, entripoint, base_url)
    Class for an API Doc.

add_baseCollection ()
    Add Collection class to the API Doc.

add_baseResource ()
    Add Resource class to the API Doc.

add_possible_status (status)
    Add a new possibleStatus.

add_supported_class (class_, collection=False, collectionGet=True, collectionPost=True)
    Add a new supportedClass.

add_to_context (key, value)
    Add entries to the vocabs context.

gen_EntryPoint ()
    Generate the EntryPoint for the Hydra Doc.

generate ()
    Get the Hydra API Doc as a python dict.

class hydrus.hydraspec.doc_writer.HydraEntryPoint (base_url, entripoint)
    Template for a new entripoint.

add_Class (class_)
    Add supportedProperty to the EntryPoint.

add_Collection (collection)
    Add supportedProperty to the EntryPoint.

generate ()
    Get as a Python dict.

get ()
    Create the EntryPoint object to be returned for the get function.

class hydrus.hydraspec.doc_writer.HydraStatus (code, title, desc)
    Class for possibleStatus in Hydra Doc.

generate ()
    Get as Python dict.

```

4.4 hydrus.hydraspec.doc_writer_sample module

Sample to create Hydra APIDocumentation using doc_writer.

```
hydrus.hydraspec.doc_writer_sample.api_doc = <hydrus.hydraspec.doc_writer.HydraDoc instance>
    Creating classes for the API
hydrus.hydraspec.doc_writer_sample.class_ = <hydrus.hydraspec.doc_writer.HydraClass instance>
    Create new properties for the class
hydrus.hydraspec.doc_writer_sample.dummyProp2 = <hydrus.hydraspec.doc_writer.HydraClassProperty instance>
    Create operations for the class
hydrus.hydraspec.doc_writer_sample.op1 = <hydrus.hydraspec.doc_writer.HydraClassOperation instance>
    Add the operation to the Class
```

4.5 hydrus.hydraspec.doc_writer_sample_output module

Generated API Documentation sample using doc_writer_sample.py.

```
doc = {
  "@context": { "ApiDocumentation": "hydra:ApiDocumentation", "description": "hydra:description", "domain": {
    "@id": "rdfs:domain", "@type": "@id"
  }, "expects": {
    "@id": "hydra:expects", "@type": "@id"
  }, "hydra": "http://www.w3.org/ns/hydra/core#", "label": "rdfs:label", "method": "hydra:method", "possibleStatus": "hydra:possibleStatus", "property": {
    "@id": "hydra:property", "@type": "@id"
  }, "range": {
    "@id": "rdfs:range", "@type": "@id"
  }, "rdf": "http://www.w3.org/1999/02/22-rdf-syntax-ns#", "rdfs": "http://www.w3.org/2000/01/rdf-schema#", "readonly": "hydra:readonly", "required": "hydra:required", "returns": {
    "@id": "hydra:returns", "@type": "@id"
  }, "statusCode": "hydra:statusCode", "statusCodes": "hydra:statusCodes", "subclassOf": {
    "@id": "rdfs:subclassOf", "@type": "@id"
  }, "supportedClass": "hydra:supportedClass", "supportedOperation": "hydra:supportedOperation", "supportedProperty": "hydra:supportedProperty", "title": "hydra:title", "vocab": "https://hydrus.com/demoapi/vocab#", "writeonly": "hydra:writeonly"
}, "@id": "https://hydrus.com/demoapi/vocab", "@type": "ApiDocumentation", "description": "Description for the API Documentation", "possibleStatus": [], "supportedClass": [
  { "@id": "http://hydrus.com/dummyClass", "@type": "hydra:Class", "description": "A dummy-Class for demo", "supportedOperation": [
    { "@type": "http://schema.org/UpdateAction", "expects": "vocab:dummyClass", "method": "POST", "possibleStatus": [
```

```

        { "description": "dummyClass updated", "statusCode": 200
        }
    ], "returns": "null", "title": "UpdateClass"
}
], "supportedProperty": [
    { "@type": "SupportedProperty", "property": "http://hydrus.com/prop1", "readonly":
      "false", "required": "false", "title": "Prop1", "writeonly": "true"
    }, {
      "@type": "SupportedProperty", "property": "http://hydrus.com/prop1",
      "readonly": "false", "required": "false", "title": "Prop2", "writeonly": "true"
    }
  ], "title": "dummyClass"
}, {
  "@id": "http://www.w3.org/ns/hydra/core#Resource", "@type": "hydra:Class", "de-
  scription": "null", "supportedOperation": [], "supportedProperty": [], "title": "Resource"
}, {
  "@id": "http://www.w3.org/ns/hydra/core#Collection", "@type": "hydra:Class", "de-
  scription": "null", "supportedOperation": [], "supportedProperty": [
    { "@type": "SupportedProperty", "property": "http://www.w3.org/ns/hydra/
      core#member", "readonly": "false", "required": "null", "title": "members",
      "writeonly": "false"
    }
  ], "title": "Collection"
}, {
  "@id": "vocab:dummyClassCollection", "@type": "hydra:Class", "description": "A col-
  lection of dummyclass", "subClassOf": "http://www.w3.org/ns/hydra/core#Collection",
  "supportedOperation": [
    { "@id": "_:dummyclass_collection_retrieve", "@type": "hydra:Operation",
      "description": "Retrieves all dummyClass entities", "expects": "null",
      "method": "GET", "returns": "vocab:dummyClassCollection", "status-
      Codes": []
    }
  ], {
    "@id": "_:dummyclass_create", "@type": "http://schema.org/
    AddAction", "description": "Create new dummyClass entity",
    "expects": "http://hydrus.com/dummyClass", "method": "PUT",
    "returns": "http://hydrus.com/dummyClass", "statusCodes": [
      { "description": "If the dummyClass entity was created success-
        fully.", "statusCode": 201
      }
    ]
  ]
}
}

```

```

    ], "supportedProperty": [
      { "@type": "SupportedProperty", "description": "The dummyclass", "property": "http://www.w3.org/ns/hydra/core#member", "readonly": "false", "required": "false", "title": "members", "writeonly": "false"
      }
    ], "title": "dummyClassCollection"
  }, {
    "@id": "vocab:EntryPoint", "@type": "hydra:Class", "description": "The main entry point or homepage of the API.", "supportedOperation": [
      { "@id": "._:entry_point", "@type": "hydra:Operation", "description": "The APIs main entry point.", "expects": "null", "method": "GET", "returns": "null", "statusCodes": "vocab:EntryPoint"
      }
    ], "supportedProperty": [
      { "hydra:description": "The dummyClassCollection collection", "hydra:title": "dummyclasscollection", "property": {
        "@id": "vocab:EntryPoint/dummyClassCollection", "@type": "hydra:Link", "description": "The dummyClassCollection collection", "domain": "vocab:EntryPoint", "label": "dummyClassCollection", "range": "vocab:dummyClassCollection", "supportedOperation": [
          { "@id": "._:dummyclass_collection_retrieve", "@type": "hydra:Operation", "description": "Retrieves all dummyClass entities", "expects": "null", "method": "GET", "returns": "vocab:dummyClassCollection", "statusCodes": []
          }, {
            "@id": "._:dummyclass_create", "@type": "http://schema.org/AddAction", "description": "Create new dummyClass entity", "expects": "http://hydrus.com/dummyClass", "method": "PUT", "returns": "http://hydrus.com/dummyClass", "statusCodes": [
              { "description": "If the dummyClass entity was created successfully.", "statusCode": 201
              }
            ]
          }
        ]
      }, "readonly": "true", "required": "null", "writeonly": "false"
      }
    ], "title": "EntryPoint"
  }
], "title": "Title for the API Documentation"

```

}

4.6 Module contents

5.1 `hydrus.hydraspec.legacy.crud_template` module

5.2 `hydrus.hydraspec.legacy.parser` module

This is an interface to Python's internal parser.

`parser.ASTType`
alias of `st`

exception `parser.ParserError`
Bases: `exceptions.Exception`

`parser.STType`
alias of `st`

`parser.ast2list` ()
Creates a list-tree representation of an ST.

`parser.ast2tuple` ()
Creates a tuple-tree representation of an ST.

`parser.compileast` ()
Compiles an ST object into a code object.

`parser.compilest` ()
Compiles an ST object into a code object.

`parser.expr` ()
Creates an ST object from an expression.

`parser.isexpr` ()
Determines if an ST object was created from an expression.

`parser.issuite` ()
Determines if an ST object was created from a suite.

`parser.sequence2ast ()`
Creates an ST object from a tree representation.

`parser.sequence2st ()`
Creates an ST object from a tree representation.

`parser.st2list ()`
Creates a list-tree representation of an ST.

`parser.st2tuple ()`
Creates a tuple-tree representation of an ST.

`parser.suite ()`
Creates an ST object from a suite.

`parser.tuple2ast ()`
Creates an ST object from a tree representation.

`parser.tuple2st ()`
Creates an ST object from a tree representation.

CHAPTER 6

Indices and tables

- `genindex`
- `modindex`
- `search`

h

hydrus, 3
hydrus.app, 1
hydrus.data, 10
hydrus.data.crud, 5
hydrus.data.db_models, 6
hydrus.data.doc_parse, 8
hydrus.data.exceptions, 8
hydrus.data.user, 9
hydrus.hydraspec, 19
hydrus.hydraspec.doc_maker, 13
hydrus.hydraspec.doc_writer, 14
hydrus.hydraspec.doc_writer_sample, 16
hydrus.hydraspec.doc_writer_sample_output,
16
hydrus.utils, 3

p

parser, 21

A

AbstractProperty (class in `hydrus.data.db_models`), 6
 add() (`hydrus.hydraspec.doc_writer.Context` method), 14
 add_baseCollection() (`hydrus.hydraspec.doc_writer.HydraDoc` method), 15
 add_baseResource() (`hydrus.hydraspec.doc_writer.HydraDoc` method), 15
 add_Class() (`hydrus.hydraspec.doc_writer.HydraEntryPointcollection_in_endpoint` method), 15
 add_Collection() (`hydrus.hydraspec.doc_writer.HydraEntryPointcollection_in_endpoint` method), 15
 add_possible_status() (`hydrus.hydraspec.doc_writer.HydraDoc` method), 15
 add_supported_class() (`hydrus.hydraspec.doc_writer.HydraDoc` method), 15
 add_supported_op() (`hydrus.hydraspec.doc_writer.HydraClass` method), 14
 add_supported_prop() (`hydrus.hydraspec.doc_writer.HydraClass` method), 14
 add_to_context() (`hydrus.hydraspec.doc_writer.HydraDoc` method), 15
 add_user() (in module `hydrus.data.user`), 9
 api_doc (in module `hydrus.hydraspec.doc_writer_sample`), 16
 app_factory() (in module `hydrus.app`), 2
 ast2list() (in module `parser`), 21
 ast2tuple() (in module `parser`), 21
 ASTType (in module `parser`), 21
 authenticate_user() (in module `hydrus.data.user`), 9

B

BaseProperty (class in `hydrus.data.db_models`), 6

C

check_authorization() (in module `hydrus.data.user`), 9
 checkClassOp() (in module `hydrus.app`), 2
 checkEndpoint() (in module `hydrus.app`), 2
 class_ (in module `hydrus.hydraspec.doc_writer_sample`), 16
 class_in_endpoint() (in module `hydrus.hydraspec.doc_maker`), 13
 ClassNotFound, 8
 collection_in_endpoint() (in module `hydrus.hydraspec.doc_maker`), 13
 compileast() (in module `parser`), 21
 compilest() (in module `parser`), 21
 Context (class in `hydrus.hydraspec.doc_writer`), 14
 Contexts (class in `hydrus.app`), 1
 convert_literal() (in module `hydrus.hydraspec.doc_maker`), 13
 createClass() (in module `hydrus.hydraspec.doc_maker`), 13
 createContext() (`hydrus.hydraspec.doc_writer.Context` method), 14
 createDoc() (in module `hydrus.hydraspec.doc_maker`), 13
 createOperation() (in module `hydrus.hydraspec.doc_maker`), 13
 createProperty() (in module `hydrus.hydraspec.doc_maker`), 13
 createStatus() (in module `hydrus.hydraspec.doc_maker`), 13

D

delete() (`hydrus.app.Item` method), 2
 delete() (`hydrus.app.ItemCollection` method), 2
 delete() (in module `hydrus.data.crud`), 5
 delete_single() (in module `hydrus.data.crud`), 5
 dummyProp2 (in module `hydrus.hydraspec.doc_writer_sample`), 16

E

Entrypoint (class in `hydrus.app`), 1

EntryPointClass (class in hydrus.hydraspec.doc_writer), 14

EntryPointCollection (class in hydrus.hydraspec.doc_writer), 14

EntryPointOp (class in hydrus.hydraspec.doc_writer), 14
 expr() (in module parser), 21

F

failed_authentication() (in module hydrus.app), 2

G

gen_EntryPoint() (hydrus.hydraspec.doc_writer.HydraDoc method), 15

generate() (hydrus.hydraspec.doc_writer.Context method), 14

generate() (hydrus.hydraspec.doc_writer.EntryPointClass method), 14

generate() (hydrus.hydraspec.doc_writer.EntryPointCollection method), 14

generate() (hydrus.hydraspec.doc_writer.EntryPointOp method), 14

generate() (hydrus.hydraspec.doc_writer.HydraClass method), 14

generate() (hydrus.hydraspec.doc_writer.HydraClassOp method), 14

generate() (hydrus.hydraspec.doc_writer.HydraClassProp method), 14

generate() (hydrus.hydraspec.doc_writer.HydraCollection method), 15

generate() (hydrus.hydraspec.doc_writer.HydraCollectionOp method), 15

generate() (hydrus.hydraspec.doc_writer.HydraDoc method), 15

generate() (hydrus.hydraspec.doc_writer.HydraEntryPoint method), 15

generate() (hydrus.hydraspec.doc_writer.HydraStatus method), 15

generate_basic_digest() (in module hydrus.data.user), 9

get() (hydrus.app.Contexts method), 1

get() (hydrus.app.Entrypoint method), 1

get() (hydrus.app.Index method), 1

get() (hydrus.app.Item method), 2

get() (hydrus.app.ItemCollection method), 2

get() (hydrus.app.Vocab method), 2

get() (hydrus.hydraspec.doc_writer.HydraEntryPoint method), 15

get() (in module hydrus.data.crud), 5

get_all_properties() (in module hydrus.data.doc_parse), 8

get_api_name() (in module hydrus.utils), 3

get_authentication() (in module hydrus.utils), 3

get_classes() (in module hydrus.data.doc_parse), 8

get_collection() (in module hydrus.data.crud), 5

get_doc() (in module hydrus.utils), 3

get_HTTP() (hydrus.data.exceptions.ClassNotFound method), 8

get_HTTP() (hydrus.data.exceptions.InstanceExists method), 8

get_HTTP() (hydrus.data.exceptions.InstanceNotFound method), 9

get_HTTP() (hydrus.data.exceptions.NotAbstractProperty method), 9

get_HTTP() (hydrus.data.exceptions.NotInstanceProperty method), 9

get_HTTP() (hydrus.data.exceptions.PropertyNotFound method), 9

get_HTTP() (hydrus.data.exceptions.UserExists method), 9

get_HTTP() (hydrus.data.exceptions.UserNotFound method), 9

get_hydrus_server_url() (in module hydrus.utils), 3

get_session() (in module hydrus.utils), 3

get_single() (in module hydrus.data.crud), 5

get_type() (hydrus.hydraspec.doc_writer.EntryPointOp method), 14

get_type() (hydrus.hydraspec.doc_writer.HydraClassOp method), 14

getEntrypoint() (in module hydrus.hydraspec.doc_maker), 13

getType() (in module hydrus.app), 2

Graph (class in hydrus.data.db_models), 6

GraphCAC (class in hydrus.data.db_models), 6

GraphIAC (class in hydrus.data.db_models), 6

GraphIII (class in hydrus.data.db_models), 7

GraphIIT (class in hydrus.data.db_models), 7

H

HydraClass (class in hydrus.hydraspec.doc_writer), 14

HydraClassOp (class in hydrus.hydraspec.doc_writer), 14

HydraClassProp (class in hydrus.hydraspec.doc_writer), 14

HydraCollection (class in hydrus.hydraspec.doc_writer), 15

HydraCollectionOp (class in hydrus.hydraspec.doc_writer), 15

HydraDoc (class in hydrus.hydraspec.doc_writer), 15

HydraEntryPoint (class in hydrus.hydraspec.doc_writer), 15

hydrify() (in module hydrus.app), 2

HydraStatus (class in hydrus.hydraspec.doc_writer), 15

hydrus (module), 3

hydrus.app (module), 1

hydrus.data (module), 10

hydrus.data.crud (module), 5

hydrus.data.db_models (module), 6

hydrus.data.doc_parse (module), 8

hydrus.data.exceptions (module), 8

hydrus.data.user (module), 9

hydrus.hydraspec (module), 19
 hydrus.hydraspec.doc_maker (module), 13
 hydrus.hydraspec.doc_writer (module), 14
 hydrus.hydraspec.doc_writer_sample (module), 16
 hydrus.hydraspec.doc_writer_sample_output (module), 16
 hydrus.utils (module), 3

I

id (hydrus.data.db_models.AbstractProperty attribute), 6
 id (hydrus.data.db_models.BaseProperty attribute), 6
 id (hydrus.data.db_models.Graph attribute), 6
 id (hydrus.data.db_models.GraphCAC attribute), 6
 id (hydrus.data.db_models.GraphIAC attribute), 6
 id (hydrus.data.db_models.GraphIII attribute), 7
 id (hydrus.data.db_models.GraphIIT attribute), 7
 id (hydrus.data.db_models.Instance attribute), 7
 id (hydrus.data.db_models.InstanceProperty attribute), 7
 id (hydrus.data.db_models.RDFClass attribute), 7
 id (hydrus.data.db_models.Terminal attribute), 8
 id (hydrus.data.db_models.User attribute), 8
 Index (class in hydrus.app), 1
 insert() (in module hydrus.data.crud), 5
 insert_classes() (in module hydrus.data.doc_parse), 8
 insert_properties() (in module hydrus.data.doc_parse), 8
 insert_single() (in module hydrus.data.crud), 5
 Instance (class in hydrus.data.db_models), 7
 InstanceExists, 8
 InstanceNotFound, 9
 InstanceProperty (class in hydrus.data.db_models), 7
 isexpr() (in module parser), 21
 issuite() (in module parser), 21
 Item (class in hydrus.app), 1
 ItemCollection (class in hydrus.app), 2

M

methods (hydrus.app.Contexts attribute), 1
 methods (hydrus.app.Entrypoint attribute), 1
 methods (hydrus.app.Index attribute), 1
 methods (hydrus.app.Item attribute), 2
 methods (hydrus.app.ItemCollection attribute), 2
 methods (hydrus.app.Vocab attribute), 2

N

name (hydrus.data.db_models.AbstractProperty attribute), 6
 name (hydrus.data.db_models.BaseProperty attribute), 6
 name (hydrus.data.db_models.InstanceProperty attribute), 7
 name (hydrus.data.db_models.RDFClass attribute), 7
 NotAbstractProperty, 9
 NotInstanceProperty, 9

O

object_ (hydrus.data.db_models.GraphCAC attribute), 6
 object_ (hydrus.data.db_models.GraphIAC attribute), 6
 object_ (hydrus.data.db_models.GraphIII attribute), 7
 object_ (hydrus.data.db_models.GraphIIT attribute), 7
 op1 (in module hydrus.hydraspec.doc_writer_sample), 16

P

paraphrase (hydrus.data.db_models.User attribute), 8
 parser (module), 21
 ParserError, 21
 post() (hydrus.app.Item method), 2
 post() (hydrus.app.ItemCollection method), 2
 predicate (hydrus.data.db_models.GraphCAC attribute), 6
 predicate (hydrus.data.db_models.GraphIAC attribute), 6
 predicate (hydrus.data.db_models.GraphIII attribute), 7
 predicate (hydrus.data.db_models.GraphIIT attribute), 7
 PropertyNotFound, 9
 put() (hydrus.app.Item method), 2
 put() (hydrus.app.ItemCollection method), 2

R

RDFClass (class in hydrus.data.db_models), 7

S

sequence2ast() (in module parser), 21
 sequence2st() (in module parser), 22
 set_api_name() (in module hydrus.utils), 3
 set_authentication() (in module hydrus.utils), 3
 set_doc() (in module hydrus.utils), 3
 set_hydrus_server_url() (in module hydrus.utils), 3
 set_response_headers() (in module hydrus.app), 3
 set_session() (in module hydrus.utils), 3
 st2list() (in module parser), 22
 st2tuple() (in module parser), 22
 STType (in module parser), 21
 subject (hydrus.data.db_models.GraphCAC attribute), 6
 subject (hydrus.data.db_models.GraphIAC attribute), 6
 subject (hydrus.data.db_models.GraphIII attribute), 7
 subject (hydrus.data.db_models.GraphIIT attribute), 7
 suite() (in module parser), 22

T

Terminal (class in hydrus.data.db_models), 7
 tuple2ast() (in module parser), 22
 tuple2st() (in module parser), 22
 type (hydrus.data.db_models.Graph attribute), 6
 type (hydrus.data.db_models.GraphCAC attribute), 6
 type (hydrus.data.db_models.GraphIAC attribute), 6
 type (hydrus.data.db_models.GraphIII attribute), 7
 type (hydrus.data.db_models.GraphIIT attribute), 7

type_ (hydrus.data.db_models.AbstractProperty attribute), 6
type_ (hydrus.data.db_models.BaseProperty attribute), 6
type_ (hydrus.data.db_models.Instance attribute), 7
type_ (hydrus.data.db_models.InstanceProperty attribute), 7

U

unit (hydrus.data.db_models.Terminal attribute), 8
update() (in module hydrus.data.crud), 5
update_single() (in module hydrus.data.crud), 5
User (class in hydrus.data.db_models), 8
UserExists, 9
UserNotFound, 9

V

validObject() (in module hydrus.app), 3
value (hydrus.data.db_models.Terminal attribute), 8
Vocab (class in hydrus.app), 2