
zencoder-py Documentation

Release 0.6.5

Alex Schworer

September 03, 2015

1 Usage	3
2 zencoder	5
3 Indices and tables	9

Contents:

Usage

Create an instance of *Zencoder*:

```
from zencoder import Zencoder
zen = Zencoder('abc123') # enter your api key
```

Submit a job to *Zencoder*:

```
# creates an encoding job with the defaults
job = zen.job.create('http://input-file/movie.avi')
print job.code
print job.body
print job.body['id']
```

Return output progress:

```
# get the transcode progress of the first output
progress = zen.output.progress(job.body['outputs'][0]['id'])
print progress.body
```

Create a new job with multiple outputs:

```
# configure your outputs with dictionaries
iphone = {
    'label': 'iPhone',
    'url': 's3://output-bucket/output-file-1.mp4',
    'width': 480,
    'height': 320
}
web = {
    'label': 'web',
    'url': 's3://output-bucket/output-file.vp8',
    'video_codec': 'vp8'
}

# the outputs kwarg requires an iterable
outputs = (iphone, web)
another_job = zen.job.create(input_url, outputs=outputs)
```

zencoder

class `zencoder.core.Zencoder` (*api_key=None, api_version=None, base_url=None, timeout=None, test=False, proxies=None, cert=None, verify=True*)

Bases: `object`

This is the entry point to the Zencoder API. You must have a valid `api_key`.

You can pass in the `api_key` as an argument, or set `ZENCODER_API_KEY` as an environment variable, and it will use that, if `api_key` is unspecified.

Set `api_version='edge'` to get the Zencoder development API. (defaults to 'v2')

`timeout`, `proxies` and `verify` can be set to control the underlying HTTP requests that are made.

class `zencoder.core.Account` (**args, **kwargs*)

Bases: `zencoder.core.HTTPBackend`

Contains all API methods relating to Accounts.

<https://app.zencoder.com/docs/api/inputs>

create (*email, tos=1, options=None*)

Creates an account with Zencoder, no API Key necessary.

<https://app.zencoder.com/docs/api/accounts/create>

details ()

Gets account details.

<https://app.zencoder.com/docs/api/accounts/show>

integration ()

Puts the account into integration mode.

<https://app.zencoder.com/docs/api/accounts/integration>

live ()

Puts the account into live mode.

<https://app.zencoder.com/docs/api/accounts/integration>

class `zencoder.core.Job` (**args, **kwargs*)

Bases: `zencoder.core.HTTPBackend`

Contains all API methods relating to transcoding Jobs.

<https://app.zencoder.com/docs/api/jobs>

cancel (*job_id*)

Cancels the given `job_id`.

<https://app.zencoder.com/docs/api/jobs/cancel>

create (*input=None, live_stream=False, outputs=None, options=None*)

Creates a transcoding job. Here are some examples:

```
job.create('s3://zencodertesting/test.mov')
job.create(live_stream=True)
job.create(input='http://example.com/input.mov',
           outputs=({'label': 'test output'},))
```

<https://app.zencoder.com/docs/api/jobs/create>

delete (*job_id*)

Deletes the given *job_id*.

WARNING: This method is aliased to *Job.cancel* – it is deprecated in API version 2 and greater.

details (*job_id*)

Returns details of the given *job_id*.

<https://app.zencoder.com/docs/api/jobs/show>

finish (*job_id*)

Finishes the live stream for *job_id*.

<https://app.zencoder.com/docs/api/jobs/finish>

list (*page=1, per_page=50*)

Lists Jobs.

<https://app.zencoder.com/docs/api/jobs/list>

progress (*job_id*)

Returns the progress of the given *job_id*.

<https://app.zencoder.com/docs/api/jobs/progress>

resubmit (*job_id*)

Resubmits the given *job_id*.

<https://app.zencoder.com/docs/api/jobs/resubmit>

class `zencoder.core.Output` (**args, **kwargs*)

Bases: `zencoder.core.HTTPBackend`

Contains all API methods relating to Outputs.

<https://app.zencoder.com/docs/api/outputs>

details (*output_id*)

Returns the details of the given *output_id*.

<https://app.zencoder.com/docs/api/outputs/show>

progress (*output_id*)

Returns the progress for the given *output_id*.

<https://app.zencoder.com/docs/api/outputs/progress>

class `zencoder.core.Response` (*code, body, raw_body, raw_response*)

Bases: `object`

The Response object stores the details of an API request.

Response.body contains the loaded JSON response from the API.

```
class zencoder.core.HTTPBackend(base_url, api_key, resource_name=None, timeout=None,  
test=False, version=None, proxies=None, cert=None, verify=True)
```

Bases: object

Abstracts out an HTTP backend. Required argument are `base_url` and `api_key`.

delete (*url, params=None*)

Executes an HTTP DELETE request for the given URL.

`params` should be a dictionary

get (*url, data=None*)

Executes an HTTP GET request for the given URL.

`data` should be a dictionary of url parameters

headers

Returns default headers, by setting the Content-Type, Accepts, User-Agent and API Key headers.

post (*url, body=None*)

Executes an HTTP POST request for the given URL.

process (*response*)

Returns HTTP backend agnostic Response data.

put (*url, data=None, body=None*)

Executes an HTTP PUT request for the given URL.

Introduction:

`zencoder` is a Python module for the [Zencoder API](#).

Official Zencoder API Docs: <https://app.zencoder.com/docs>

`zencoder-py` Github: <http://github.com/zencoder/zencoder-py>

Indices and tables

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

A

Account (class in zencoder.core), 5

C

cancel() (zencoder.core.Job method), 5

create() (zencoder.core.Account method), 5

create() (zencoder.core.Job method), 6

D

delete() (zencoder.core.HTTPBackend method), 7

delete() (zencoder.core.Job method), 6

details() (zencoder.core.Account method), 5

details() (zencoder.core.Job method), 6

details() (zencoder.core.Output method), 6

F

finish() (zencoder.core.Job method), 6

G

get() (zencoder.core.HTTPBackend method), 7

H

headers (zencoder.core.HTTPBackend attribute), 7

HTTPBackend (class in zencoder.core), 6

I

integration() (zencoder.core.Account method), 5

J

Job (class in zencoder.core), 5

L

list() (zencoder.core.Job method), 6

live() (zencoder.core.Account method), 5

O

Output (class in zencoder.core), 6

P

post() (zencoder.core.HTTPBackend method), 7

process() (zencoder.core.HTTPBackend method), 7

progress() (zencoder.core.Job method), 6

progress() (zencoder.core.Output method), 6

put() (zencoder.core.HTTPBackend method), 7

R

Response (class in zencoder.core), 6

resubmit() (zencoder.core.Job method), 6

Z

Zencoder (class in zencoder.core), 5