PyOffers Documentation

Release 0.6.9

Dmitry Dygalo

Jun 28, 2018

Contents

1 Usage **3**
- 1.1 Basic . . . 3
- 1.2 Models . . . 4

2 Contributing to PyOffers **7**
- 2.1 Quickstart . . . 7
- 2.2 Syntax and conventions . . . 7
- 2.3 Running the tests . . . 7

3 Changelog **9**
- 3.1 Unreleased . . . 9
- 3.2 **'0.6.9'_** - 2018-06-19 . . . 9
- 3.3 **'0.6.8'_** - 2018-06-15 . . . 9
- 3.4 **'0.6.7'_** - 2018-05-11 . . . 10
- 3.5 **'0.6.6'_** - 2018-04-24 . . . 10
- 3.6 **'0.6.5'_** - 2018-04-16 . . . 10
- 3.7 0.6.4 - 2018-03-28 . . . 10
- 3.8 0.6.3 - 2018-03-15 . . . 10
- 3.9 0.6.2 - 2018-03-12 . . . 10
- 3.10 0.6.1 - 2018-02-28 . . . 11
- 3.11 0.6.0 - 2018-02-20 . . . 11
- 3.12 0.5.0 - 2016-09-20 . . . 11
- 3.13 0.4.4 - 2016-09-09 . . . 11
- 3.14 0.4.3 - 2016-09-09 . . . 12
- 3.15 0.4.2 - 2016-09-01 . . . 12
- 3.16 0.4.1 - 2016-09-01 . . . 12
- 3.17 0.4.0 - 2016-08-31 . . . 12
- 3.18 0.3.2 - 2016-08-30 . . . 13
- 3.19 0.3.1 - 2016-08-30 . . . 13
- 3.20 0.3.0 - 2016-08-30 . . . 13
- 3.21 0.2.2 - 2016-08-29 . . . 14
- 3.22 0.2.1 - 2016-08-29 . . . 14
- 3.23 0.2.0 - 2016-08-29 . . . 14
- 3.24 0.1.2 - 2016-08-29 . . . 14
- 3.25 0.1.1 - 2016-08-28 . . . 14
- 3.26 0.1.0 - 2016-08-28 . . . 15

4 Indices and tables 17

Documentation:

CHAPTER 1

Usage

1.1 Basic

PyOffers provide `HasOffersAPI` class to interact with HasOffers API.

```
from pyoffers.api import HasOffersAPI


hasoffers = HasOffersAPI(
    endpoint='https://api.hasoffers.com/Apiv3/json',
    network_token='<your_network_token>',
    network_id='<your_network_id>',
    verify=True,
    retries=4,
    retry_timeout=4,
    verbosity=3
)
```

Possible options:

endpoint URL, which points to HasOffers server. It may contain your subdomain as well - `https://<your_subdomain>.api.hasoffers.com/Apiv3/json`

network_token Your NetworkToken.

network_id Your NetworkId.

verify Whether to verify SSL certificate or not. HasOffers SSL cert doesn't cover subdomains like mentioned above. Only `*.hasoffers.com` and `hasoffers.com`.

retries Number of retries to make a request in case of exceeding HasOffers API rate limit.

retry_timeout Seconds to wait between retries.

verbosity Controls library's verbosity level. With **3** the request params and raw response will be printed to console.

With this class instance you can perform requests to the API.

1.2 Models

The core idea of interaction with model is based on managers. To work with HasOffers data model you should use appropriate manager. For now only very limited part of models are presented.

1.2.1 Core managers methods

create (*self*, ***kwargs*)

Just pass model data as `kwargs` to create new instance of model. The new instance will be returned.

update (*self*, *id*, ***kwargs*)

Takes `id` of existing model instance and new data as `kwargs`. Updated instance will be returned.

find_by_id (*self*, *id*, *contain=None*)

If you pass related models names to `contain` argument, then if related model exists it will be attached to the returned object as its attribute.

```
>>> offer = hasoffers.offers.find_by_id(id=62, contain=['Country'])
>>> print(offer.country)
<Country: 724>
```

find_all (*self*, *sort=()*, *limit=None*, *page=None*, *contain=None*, ***kwargs*)

`sort` can be a single field name or list of them. To sort in descending order put − before the field name.

```
>>> hasoffers.offers.find_all(sort='-id')
[<Offer: 9>,
 <Offer: 7>,
 <Offer: 5>,
 <Offer: 3>,
 <Offer: 1>]
```

There are some extra features to make advanced queries. To use different operators you should append its name to the end of the field_name. Example. To get all offers with `id` greater than 100:

```
>>> hasoffers.offers.find_all(id__gt=100, sort='id')
[<Offer: 102>,
 <Offer: 104>,
 <Offer: 106>,
 <Offer: 108>,
 <Offer: 110>,
 <Offer: 112>]
```

PyOffers supports the following operators:

- `ne` - `NOT_EQUAL_TO`
- `lt` - `LESS_THAN`
- `lte` - `LESS_THAN_OR_EQUAL_TO`
- `gt` - `GREATER_THAN`
- `gte` - `GREATER_THAN_OR_EQUAL_TO`

- `like` - LIKE
- `not_like` - NOT_LIKE
- `null` - NULL
- `not_null` - NOT_NULL
- `true` - TRUE
- `false` - FALSE

To perform `OR` queries pass `connector='OR'` as filter. To get all offers with **active** status OR with **USD** currency:

```
>>> hasoffers.offers.find_all(status='active', currency='USD', connector='OR')
```

To skip models instantiation you can pass `raw=True` to underlying `_call` method. This makes it possible to perform queries without instantiation of model instances.

```
>>> hasoffers.advertisers._call('findAllIds', raw=True)
['1', '2', '3', '4']
```

1.2.2 Managers

- **Advertiser** - `hasoffers.advertisers`
- **Conversion** - `hasoffers.conversions`
- **Country** - As the model has no methods then `hasoffers.countries` manager has no methods too.
- **Goal** - `hasoffers.goals`
- **Offer** - `hasoffers.offers`
- **Affiliate** - `hasoffers.affiliates`
- **AffiliateUser** - `hasoffers.affiliate_users`
- **OfferFile** - `hasoffers.offer_files`

1.2.3 Related managers

`Offer` model has `conversions` manager, which allows to perform queries on conversions related to concrete offer.

```
>>> offer = hasoffers.offers.find_by_id(7)
>>> conversions = offer.conversions.find_all()
```

Other related managers:

- `Affiliate.files`
- `Offer.files`

1.2.4 Raw logs

`pyoffers` provides `RawLog` model with 3 managers for different types of logs. They behave identically:

- `hasoffers.raw_logs.clicks` - logs about clicks.
- `hasoffers.raw_logs.conversions` - about conversions.

- `hasoffers.raw_logs.impressions` - about impressions.

To get all logs directories:

```
>>> hasoffers.raw_logs.clicks.list_date_dirs()
[<DateDir: Sep 20, 2016 (20160920)>, <DateDir: Sep 19, 2016 (20160919)>, ...]
```

Every directory has 2 attributes:

- `displayName`
- `dirName`

And `list_logs` method:

```
>>> directory = hasoffers.raw_logs.clicks.list_date_dirs()[0]
>>> directory.list_logs()
[<LogFile: Sep 20, 2016 - 11:00 am (20160920/clicks-1474369200-ewU6Y1.zip)>, ...]
```

Each log file has:

- `download_link` - link to ZIP archive at Amazon S3.
- `content` - raw CSV content of this archive.
- `records` - all data from CSV wrapped as `LogRecord` instances.

All these attributes are cached on `LogFile` instance level. For convenience it is possible to get all records for some date:

```
>>> hasoffers.raw_logs.clicks.find_all('20160920')
[<LogRecord: 7 (1027a606128bd067105f0b0921840f)>, ...]
```

Also it is possible to get records for month or even for year. But it will take some time to download all data.

```
>>> september_clicks = hasoffers.raw_logs.clicks.find_all('201609')
>>> year_clicks = hasoffers.raw_logs.clicks.find_all('2016')
```

1.2.5 Offer files (Creatives)

`OfferFile` provides and ability to upload offer file (aka "Creative" in Hasoffers terms). To do so, you can use `create` method:

```
>>>hasoffers.offer_files.create('/relative/or/absolute/path/test.png', display='TEST_
↪FILE',
    type='offer thumbnail', width=200,height=100, offer_id=438, filename='custom_
↪filename.png')
<OfferFile: 98>
```

CHAPTER 2

Contributing to PyOffers

PyOffers is contribute-friendly project. There is a guideline for contributing.

2.1 Quickstart

- All code contributions should be tested
- Code should conform to syntax conventions
- Documentation should be updated if it is required
- Put a note to the changelog.

2.2 Syntax and conventions

The source code should conform to PEP8 with following notice:

- Line length should not exceed **120** characters.

2.3 Running the tests

We use `tox` to run the tests:

```
$ tox
```

CHAPTER 3

Changelog

3.1 Unreleased

3.1.1 Added

- `display_attribute` property to models, that defines what property will be displayed in __str__ representation. (iamanikeev)

3.2 ‘0.6.9‘_ - 2018-06-19

3.2.1 Added

- `find_all_affiliate_approvals`. (iamanikeev)
- Fix `find_all_(offer|affiliate|advertiser)_tag_relations` methods for Tag model. (iamanikeev)

3.3 ‘0.6.8‘_ - 2018-06-15

3.3.1 Added

- Support of Tag controller. (iamanikeev)

3.4 '0.6.7'_ - 2018-05-11

3.4.1 Added

- `generate_tracking_link` method for `Offer` model. (iamanikeev)

3.5 '0.6.6'_ - 2018-04-24

3.5.1 Added

- `unblock_affiliate`, `set_affiliate_approval_status`, `get_affiliate_approval_status`, `get_blocked_affiliate_ids`, `get_approved_affiliate_ids`, `get_unapproved_affiliate_ids` methods for `Offer` model. (iamanikeev)

3.6 '0.6.5'_ - 2018-04-16

3.6.1 Fixed

- Removed internal dependency on `pip.utils`, which makes the library compatible with pip 10. #70 (butorov, Stranger6667)

3.7 0.6.4 - 2018-03-28

3.7.1 Added

- New related managers `Offer.files` and `Affiliate.files`. (Stranger6667)
- Support for `CreativeCode` in `contains` list. (Stranger6667)
- `get_offer_files_with_creative_code` method for `Affiliate` and `Offer` models. (Stranger6667)

3.8 0.6.3 - 2018-03-15

3.8.1 Added

- Support managing of `OfferFile` objects. (iamanikeev)

3.9 0.6.2 - 2018-03-12

3.9.1 Fixed

- Initialization of singular objects from API calls. (iamanikeev)

3.10 0.6.1 - 2018-02-28

3.10.1 Added

- Introduce `Application` model. (iamanikeev)
- Support managing of `OfferCategory` objects. (iamanikeev)

3.10.2 Fixed

- Initialization of array of objects from API calls. (iamanikeev)

3.11 0.6.0 - 2018-02-20

3.11.1 Added

- Support `blockAffiliate` method. (iamanikeev)
- Add `Affiliate` and `AffiliateUser` models. (iamanikeev)
- Recreate session in case of ConnectionReset errors. (iamanikeev)

3.12 0.5.0 - 2016-09-20

3.12.1 Added

- Added `fields` parameter. #34
- Added retrying support on rate limit exceeding. #43
- Added type checks for calls parameters. #51
- Added `conversions` manager for `Offer` instances. #48

3.12.2 Changed

- Better exceptions representation. #52

3.12.3 Fixed

- Fixed credentials cleaning for non gzipped content. #45

3.13 0.4.4 - 2016-09-09

3.13.1 Changed

- Improved logs filtration. #42

3.14 0.4.3 - 2016-09-09

3.14.1 Added

- Added caching for raw logs. #41
- Added `RawLog` model. #40
- Added `as_dict` method to models. #39

3.14.2 Changed

- Better interface for models. #38

3.15 0.4.2 - 2016-09-01

3.15.1 Added

- Added `raw` argument to `HasOffersAPI._call`. #36
- Implemented `findAllIds` method. #35

3.15.2 Changed

- Made SSL certificate verification optional. #33

3.16 0.4.1 - 2016-09-01

3.16.1 Changed

- Improved sorting. #31

3.16.2 Fixed

- Fixed `contain` behaviour. #32

3.17 0.4.0 - 2016-08-31

3.17.1 Added

- Implemented generic methods. #10
- Implemented `OR` queries. #19
- Implemented `sort` in queries. #29

3.17.2 Changed

- Better interface for `update` method. #28
- Improved `contain`. Added `contain` support to `find_by_id` method. #27

3.17.3 Fixed

- Fixed managers sharing between API instances. #26

3.18 0.3.2 - 2016-08-30

3.18.1 Added

- Added `Country` model. #24
- Added `get_target_countries` method to `Offer` model. #25

3.19 0.3.1 - 2016-08-30

3.19.1 Added

- Added `find_all` methods for all defined models. #23
- Initial support for `contain` in `find_all` queries. #22

3.19.2 Changed

- Improved logging. #20, #21

3.20 0.3.0 - 2016-08-30

3.20.1 Added

- Filters implementation. #6

3.20.2 Fixed

- Fixed invalid queries building. #16
- Fixed error on empty not paginated results. #17
- Fixed error on single result in `find_all` call. #18

3.21 0.2.2 - 2016-08-29

3.21.1 Fixed

- Fixed error on not paginated results. #14

3.22 0.2.1 - 2016-08-29

3.22.1 Fixed

- Don't touch cassette if it contains no production credentials. #12
- Fixed exception if no results were found. #13

3.23 0.2.0 - 2016-08-29

3.23.1 Added

- Added support for returning multiple objects. #7
- Added `Conversion` model. #3
- Better models representation. #8

3.23.2 Changed

- Refactored models instantiation. #9

3.23.3 Fixed

- Fixed credentials leak. #11

3.24 0.1.2 - 2016-08-29

3.24.1 Fixed

- Fixed query building for lists and tuples. #5

3.25 0.1.1 - 2016-08-28

3.25.1 Fixed

- Fixed packaging issue.

3.26 0.1.0 - 2016-08-28

- Initial release.

CHAPTER 4

Indices and tables

- genindex
- modindex
- search

Index

C

create() (built-in function), 4

F

find_all() (built-in function), 4
find_by_id() (built-in function), 4

U

update() (built-in function), 4