tune-reporting-python Documentation

Release 2.7.5

TUNE Inc.

Mar 07, 2018

Contents:

1 Introduction **1**

2 Badges **3**

3 Install **5**

4 Requirements **7**

5 Run Examples **9**

6 Run Tests **11**

7 Classes **13**

7.1 `class TuneV2Advertisers` 13

7.2 `class TuneV2SessionAuthenticate` 14

7.3 `class TuneV2AdvertiserSites` 15

7.4 `class TuneV2AdvertiserStatsActuals` 16

8 License **21**

CHAPTER 1

Introduction

TUNE Reporting API client library.

CHAPTER 2

Badges

docs	
info	
tests	
package	
other	

CHAPTER 3

Install

```
pip install tune_reporting
```

CHAPTER 4

Requirements

Prerequisites Python 3.0

API Key To use SDK, it requires you to Generate API Key

CHAPTER 5

Run Examples

```
make run-examples tmc_api_key=[TMC API KEY]
```

CHAPTER 6

Run Tests

```
make test tmc_api_key=[TMC API KEY]
```

CHAPTER 7

Classes

There are multiple TUNE API Classes available:

- `TuneV2Advertisers`
- `TuneV2AdvertiserSites`
- `TuneV2AdvertiserStatsActuals`
- `TuneV2SessionAuthenticate`
- `TuneV3LogsAdvertisersClicks`
- `TuneV3LogsAdvertisersImpressions`

7.1 class TuneV2Advertisers

Get **''ADVERTISER ID''** for this account based upon provided **''TMC_API_KEY''**.

Code

```
tune_v2_advertisers = TuneV2Advertisers(
    logger_level=logging.INFO,
    logger_format=LoggingFormat.JSON,
    logger_output=LoggingOutput.STDOUT_COLOR
)

try:
    tune_v2_advertisers.tmc_auth(tmc_api_key=tmc_api_key)

    if tune_v2_advertisers.get_advertiser_id(
        auth_value=tmc_api_key,
        auth_type=TuneV2AuthenticationTypes.API_KEY,
        request_retry=None
    ):
```

```
        advertiser_id = tune_v2_advertisers.advertiser_id
        pprint(advertiser_id)

except TuneRequestBaseError as tmc_req_ex:
    print_traceback(tmc_req_ex)
    pprint(tmc_req_ex.to_dict())
    print(str(tmc_req_ex))

except TuneReportingError as tmc_rep_ex:
    pprint(tmc_rep_ex.to_dict())
    print(str(tmc_rep_ex))

except Exception as ex:
    print_traceback(ex)
    print(get_exception_message(ex))
```

Example

```
$ cd examples
$ make example_tune_v2_advertisers tmc_api_key=[ ... TMC API-Key ...]

{"asctime": "2017-11-29 08:58:30 -0800", "levelname": "INFO", "name": "tune_reporting
↪", "version": "2.3.1",
"message": "TMC Authentication: Start"}
{"asctime": "2017-11-29 08:58:31 -0800", "levelname": "INFO", "name": "requests_mv_
↪integrations", "version": "00.06.01",
"message": "TMC Authentication: Finished", "request_time_msecs": 635}
{"asctime": "2017-11-29 08:58:31 -0800", "levelname": "INFO", "name": "tune_reporting
↪", "version": "2.3.1",
"message": "TMC v2 Advertisers: Advertiser ID"}
{"asctime": "2017-11-29 08:58:31 -0800", "levelname": "INFO", "name": "requests_mv_
↪integrations", "version": "00.06.01",
"message": "TMC v2 Advertisers: Finished", "request_time_msecs": 260}
{"asctime": "2017-11-29 08:58:31 -0800", "levelname": "INFO", "name": "tune_reporting
↪", "version": "2.3.1",
"message": "TMC v2 Advertisers: Advertiser ID: [ADVERTISER ID]"}

[ADVERTISER ID]
```

7.2 class TuneV2SessionAuthenticate

Get time-limited **''SESSION TOKEN''** after authenticating provided **''TMC_API_KEY''**.

Code

```
tune_v2_session_authenticate = \
    TuneV2SessionAuthenticate(
        logger_level=logging.INFO
    )

try:
    if tune_v2_session_authenticate.get_session_token(
        tmc_api_key=tmc_api_key,
        request_retry=None
```

```
    ):
        session_token = tune_v2_session_authenticate.session_token
        print(session_token)

except TuneRequestBaseError as tmc_req_ex:
    print_traceback(tmc_req_ex)
    pprint(tmc_req_ex.to_dict())
    print(str(tmc_req_ex))

except TuneReportingError as tmc_rep_ex:
    pprint(tmc_rep_ex.to_dict())
    print(str(tmc_rep_ex))

except Exception as ex:
    print_traceback(ex)
    print(get_exception_message(ex))
```

Example

```
$ cd examples
$ make example_tune_v2_session_authenticate tmc_api_key=[ ... TMC API-Key ...]

{"asctime": "2017-11-29 09:11:09 -0800", "levelname": "INFO", "name": "tune_reporting
↪", "version": "2.3.1",
"message": "TMC v2 Session Authenticate: Get Token"}
{"asctime": "2017-11-29 09:11:11 -0800", "levelname": "INFO", "name": "requests_mv_
↪integrations", "version": "00.06.01",
"message": "TMC v2 Session Authenticate: Finished", "request_time_msecs": 1550}
{"asctime": "2017-11-29 09:11:11 -0800", "levelname": "INFO", "name": "tune_reporting
↪", "version": "2.3.1",
"message": "TMC v2 Session Authenticate", "session_token": "[SESSION TOKEN]"}
{"asctime": "2017-11-29 09:11:11 -0800", "levelname": "INFO", "name": "tune_reporting
↪", "version": "2.3.1",
"message": "TMC v2 Session Authenticate: Finished"}

[SESSION TOKEN]
```

7.3 class TuneV2AdvertiserSites

Get listing of Advertiser's Mobile Apps (aka Sites) for this account based upon provided **''TMC_API_KEY''**.

Code

```
tune_advertiser_sites = TuneV2AdvertiserSites(
    logger_level=logging.INFO
)

try:
    tune_advertiser_sites.tmc_auth(tmc_api_key=tmc_api_key)

    for collect_data_item, collect_error in tune_advertiser_sites.collect(
        auth_value=tmc_api_key,
        auth_type=TuneV2AuthenticationTypes.API_KEY,
        auth_type_use=TuneV2AuthenticationTypes.API_KEY,
```

```
        request_params={'limit': 5}
    ):
        pprint(collect_data_item)

except TuneRequestBaseError as tmc_req_ex:
    print_traceback(tmc_req_ex)
    pprint(tmc_req_ex.to_dict())
    print(str(tmc_req_ex))

except TuneReportingError as tmc_rep_ex:
    print_traceback(tmc_rep_ex)
    pprint(tmc_rep_ex.to_dict())
    print(str(tmc_rep_ex))

except Exception as ex:
    print_traceback(ex)
    print(get_exception_message(ex))
```

Example

```
$ cd examples
$ make example_tune_v2_advertiser_sites tmc_api_key=[ ... TMC API-Key ...]

{"asctime": "2017-11-29 09:04:25 -0800", "levelname": "INFO", "name": "tune_reporting
↪", "version": "2.3.1",
"message": "TMC Authentication: Start"}
{"asctime": "2017-11-29 09:04:25 -0800", "levelname": "INFO", "name": "requests_mv_
↪integrations", "version": "00.06.01",
"message": "TMC Authentication: Finished", "request_time_msecs": 593}
{"asctime": "2017-11-29 09:04:25 -0800", "levelname": "INFO", "name": "tune_reporting
↪", "version": "2.3.1",
"message": "Start Advertiser Sites find"}
{"asctime": "2017-11-29 09:04:26 -0800", "levelname": "INFO", "name": "requests_mv_
↪integrations", "version": "00.06.01",
"message": "TuneV2AdvertiserSites.collect: Finished", "request_time_msecs": 263}

[JSON RESPONSE]
{
    'id': 533,
    'name': 'TEST UP TIME - DONT DELETE',
    'package_name': 'unknown',
    'status': 'active',
    'url': 'http://website.com',
}
...
```

7.4 class TuneV2AdvertiserStatsActuals

Logs of Advertiser's Actuals Stats for this account based upon provided **''TMC_API_KEY''**.

Code

```
tune_v2_advertiser_stats_actuals = \
    TuneV2AdvertiserStatsActuals(
```

```
        logger_level=logging.INFO,
        logger_format=LoggingFormat.JSON,
        logger_output=LoggingOutput.STDOUT_COLOR
    )

tz = pytz.timezone("America/New_York")
yesterday = datetime.now(tz).date() - timedelta(days=1)
str_yesterday = str(yesterday)

try:
    auth_response = tune_v2_advertiser_stats_actuals.tmc_auth(tmc_api_key=tmc_api_key)
    assert auth_response

    tune_v2_advertiser_stats_actuals.collect(
        auth_value=tmc_api_key,
        auth_type=TuneV2AuthenticationTypes.API_KEY,
        auth_type_use=TuneV2AuthenticationTypes.API_KEY,
        start_date=str_yesterday,
        end_date=str_yesterday,
        request_params={
            'timezone': 'America/Los_Angeles',
            'format': TuneV2AdvertiserStatsFormats.CSV,
            'fields': (
                "ad_clicks,"
                "ad_clicks_unique,"
                "ad_impressions,"
                "ad_impressions_unique,"
                "ad_network_id,"
                "advertiser_id,"
                "country.code,"
                "date_hour,"
                "events,"
                "installs,"
                "is_reengagement,"
                "payouts,"
                "publisher_id,"
                "publisher_sub_ad.ref,"
                "publisher_sub_adgroup.ref,"
                "publisher_sub_campaign.ref,"
                "publisher_sub_publisher.ref,"
                "publisher_sub_site.ref,"
                "site_id"
            ),
            'group': (
                "country_id,"
                "is_reengagement,"
                "publisher_id,"
                "publisher_sub_ad_id,"
                "publisher_sub_adgroup_id,"
                "publisher_sub_campaign_id,"
                "publisher_sub_publisher_id,"
                "publisher_sub_site_id,"
                "site_id"
            ),
            'timezone': "America/Los_Angeles",
            'limit': 5
        },
        request_action=TuneV2AdvertiserStatsActions.EXPORT,
```

```
        request_retry={'delay': 15,
                       'timeout': 30,
                       'tries': 10}
    )

except TuneRequestBaseError as tmc_req_ex:
    print_traceback(tmc_req_ex)
    pprint(tmc_req_ex.to_dict())
    print(str(tmc_req_ex))

except TuneReportingError as tmc_rep_ex:
    pprint(tmc_rep_ex.to_dict())
    print(str(tmc_rep_ex))

except Exception as ex:
    print_traceback(ex)
    print(get_exception_message(ex))

for row in list(tune_v2_advertiser_stats_actuals.generator):
    pprint(row)
```

Example

```
$ cd examples
$ make example_tune_v2_advertiser_stats_actuals_export_download tmc_api_key=[ ... TMC␣
↪API-Key ...]

{"asctime": "2017-11-29 09:17:21 -0800", "levelname": "INFO", "name": "tune_reporting
↪", "version": "2.3.1",
"message": "TMC Authentication: Start"}
{"asctime": "2017-11-29 09:17:22 -0800", "levelname": "INFO", "name": "requests_mv_
↪integrations", "version": "00.06.01",
"message": "TMC Authentication: Finished", "request_time_msecs": 516}
{"asctime": "2017-11-29 09:17:22 -0800", "levelname": "INFO", "name": "tune_reporting
↪", "version": "2.3.1",
"message": "TMC v2 Advertiser Stats: Collect: export"}
{"asctime": "2017-11-29 09:17:23 -0800", "levelname": "INFO", "name": "requests_mv_
↪integrations", "version": "00.06.01",
"message": "TMC v2 Advertiser Stats Find: Finished", "request_time_msecs": 1490}

[ADVERTISER ACTUALS STATS]
{
    'ad_clicks': '48',
    'ad_clicks_unique': '0',
    'ad_impressions': '0',
    'ad_impressions_unique': '0',
    'ad_network_id': 0,
    'advertiser_id': 877,
    'conversions': '0',
    'country': {'code': 'NL', 'name': 'Netherlands'},
    'country_id': 528,
    'currency_code': 'USD',
    'date_hour': '2017-11-28 19:00:00',
    'events': '0',
    'installs': '0',
    'is_reengagement': '0',
    'payouts': '0.00000',
    'publisher': {'name': 'PINGDOM DO_NOT_DELETE'},
```

```
    'publisher_id': 142476,
    'publisher_sub_ad': {'ref': ''},
    'publisher_sub_ad_id': '0',
    'publisher_sub_adgroup': {'ref': ''},
    'publisher_sub_adgroup_id': '0',
    'publisher_sub_campaign': {'ref': ''},
    'publisher_sub_campaign_id': '0',
    'publisher_sub_publisher': {'ref': ''},
    'publisher_sub_publisher_id': '0',
    'publisher_sub_site': {'ref': ''},
    'publisher_sub_site_id': '0',
    'purchase_validation_status': '0',
    'site': {
        'mobile_app_type': 'iOS',
        'package_name': 'unknown',
        'store_app_id': None,
    },
    'site_id': 533,
}
...
```

CHAPTER 8

License

MIT License.