

Ready Signal API - Python 3.6+

This library is designed to be a wrapper for the Ready Signal API: <http://app.readysignal.com>
It is also hosted on GitHub: <https://github.com/rxa-io/readysignal>
Please direct all questions and/or recommendations to support@readysignal.com

Installation

The Ready Signal API Python library can be found here: <https://pypi.org/project/readysignal/>

```
pip install readysignal
```

Usage

Your **access token** and **signal ID** can be found on your "Manage Signal" page within the Output information.

Your signal ID is also visible within the URL of the "Manage Signal" page:

```
...readysignal.com/signal/SIGNAL_ID/manage
```

Setup

```
import readysignal as rs

access_token = "your access token"
signal_id = 0 # this is your unique signal id number
```

List Signals

Using your `access_token`, you can list all signals and metadata that are associated with your Ready Signal account.

```
rs.list_signals(access_token)
```

Example Output

```
{'data': [{
  'id': 0,
  'name': 'Signal Name',
  'description': 'Signal Description',
  'desired_geo_grain': 'Country',
  'desired_time_grain': 'Month',
  'start_at': '11/01/2019',
  'end_at': '01/31/2020',
  'created_at': '07/23/2020',
  'updated_at': '07/28/2020',
  'deleted_at': None,
  'company': {
    'id': 0,
    'chargebee_id': 'ID Hash',
    'name': 'Company Name',
    'plan_id': 0,
    'plan_value_id': 0,
    'subscription_status': 'active',
    'created_at': '2020-07-23 13:20:17',
    'updated_at': '2020-08-03 11:33:50',
    'need_listining_webhook': 0,
    'addon_users': None,
    'notify_after_processing': 0
  },
  'user': {
    'id': 0,
    'company_id': 0,
    'name': 'Name',
    'email': 'Email',
    'email_verified_at': None,
    'type': 'owner',
    'active': 'yes',
    'reactivate': 'no',
    'created_at': '2020-07-23 13:20:17',
    'updated_at': '2020-07-23 13:20:17',
    'deleted_at': None
  },
  'industry': {
    'id': 1,
    'name': 'Automotive',
    'created_at': '2020-04-10 23:21:38',
    'updated_at': '2020-04-10 23:21:38',
    'deleted_at': None
  },
  'analysis_type': {
    'id': 3,
    'name': 'Classification',
    'created_at': '2020-04-10 23:21:45',
    'updated_at': '2020-04-10 23:21:45',
    'deleted_at': None
  },
},
  'output': {
    'json': 'https://app.readysignal.com/api/signals/0/output?format=json',
    'links': {
      'self': 'https://app.readysignal.com/signal/0/manage',
      'manage': 'https://app.readysignal.com/signal/0/manage'
    }
  },
},
{
  'id': 1
  ...
}
]
```

Signal Details

Using your `access_token` and your `signal_id` you can view the details of a specific signal.

```
# get signal details
rs.get_signal_details(access_token, signal_id)
```

Example Output

```

{'data': {
  'id': 0,
  'name': 'Signal Name',
  'description': 'Signal Description',
  'desired_geo_grain': 'Country',
  'desired_time_grain': 'Month',
  'start_at': '11/01/2019',
  'end_at': '01/31/2020',
  'created_at': '07/23/2020',
  'updated_at': '07/28/2020',
  'deleted_at': None,
  'company': {
    'id': 0,
    'chargebee_id': 'ID Hash',
    'name': 'Company Name',
    'plan_id': 0,
    'plan_value_id': 0,
    'subscription_status': 'active',
    'created_at': '2020-07-23 13:20:17',
    'updated_at': '2020-08-03 11:33:50',
    'need_listining_webhook': 0,
    'addon_users': None,
    'notify_after_processing': 0
  },
  'user': {
    'id': 0,
    'company_id': 0,
    'name': 'Name',
    'email': 'Email',
    'email_verified_at': None,
    'type': 'owner',
    'active': 'yes',
    'reactivate': 'no',
    'created_at': '2020-07-23 13:20:17',
    'updated_at': '2020-07-23 13:20:17',
    'deleted_at': None
  },
  'industry': {
    'id': 1,
    'name': 'Automotive',
    'created_at': '2020-04-10 23:21:38',
    'updated_at': '2020-04-10 23:21:38',
    'deleted_at': None
  },
  'analysis_type': {
    'id': 3,
    'name': 'Classification',
    'created_at': '2020-04-10 23:21:45',
    'updated_at': '2020-04-10 23:21:45',
    'deleted_at': None
  },
  'output': {
    'json': 'https://app.readysignal.com/api/signals/0/output?format=json',
    'links': {
      'self': 'https://app.readysignal.com/signal/0/manage',
      'manage': 'https://app.readysignal.com/signal/0/manage'
    }
  }
}
}

```

Signal Output

There are three different ways to receive your signal output:

- JSON
- Pandas DataFrame
- CSV export

JSON

```
# get signal data as json
rs.get_signal(access_token, signal_id)
```

Example Output

```
{'current_page': 1,
 'data': [
  {
    'start': '2019-11-01',
    'end': '2019-11-30',
    'country': 'United States of America',
    'actual-snow-fall': '160205.00000000000000000000',
    'resident-population-by-state': '9826013.22000000000000000000',
    'actual-snow-cover': '3.56158109943317700000',
    'population-total': '308745538.00000000000000000000',
    'population-total-transformed': 17,571.15642181115497622280458798
  },
  {
    'start': '2019-12-01',
    'end': '2019-12-31',
    'country': 'United States of America',
    'actual-snow-fall': '219691.00000000000000000000',
    'resident-population-by-state': '10153546.99400000000000000000',
    'actual-snow-cover': '8.28314041638492200000',
    'population-total': '308745538.00000000000000000000',
    'population-total-transformed': 17,571.15642181115497622280458798
  },
  {
    'start': '2020-01-01',
    'end': '2020-01-31',
    'country': 'United States of America',
    'actual-snow-fall': '220869.00000000000000000000',
    'resident-population-by-state': 10159386.99400000000000000000,
    'actual-snow-cover': '10.69758409714073700000',
    'population-total': '308745538.00000000000000000000',
    'population-total-transformed': 17,571.15642181115497622280458798
  }
 ],
 'first_page_url': 'https://app.readysignal.com/api/signals/0/output?page=1',
 'from': 1,
 'last_page': 1,
 'last_page_url': 'https://app.readysignal.com/api/signals/0/output?page=1',
 'next_page_url': None,
 'path': 'https://app.readysignal.com/api/signals/0/output',
 'per_page': 1000,
 'prev_page_url': None,
 'to': 3,
 'total': 3}
```

Pandas DataFrame

```
# get signal data as Pandas DataFrame
rs.get_signal_pandas(access_token, signal_id)
```

Example Output

```
start ... population-total-transformed
0 2019-11-01 ... 17,571.156421818115497622280458798
1 2019-12-01 ... 18,109.798447234298274239287429023
2 2020-01-01 ... 18,732.472983479821748127047902849
```

Export to CSV

```
# send signal data to csv file
file_name = "test_signal.csv"
rs.signal_to_csv(access_token, signal_id, file_name)
```