

(/home)

[All APIs \(/all/api-overviews\)](/all/api-overviews)[Q & A](#)[\(/user/me/notifications\)](/user/me/notifications)

vaetit.sirasupprachai

[Home \(/\)](#) > [Eikon APIs \(/eikon-apis\)](/eikon-apis) > [Eikon Data APIs · Beta](#)

Eikon Data APIs · Beta

API Family: [Eikon APIs \(/eikon-apis/apis-in-this-family\)](/eikon-apis/apis-in-this-family) · ☆ [Add to My Watchlist \(/flag/flag/watchlists/5565?destination=node/5565/quick-start&token=JV1vW4NYEZ6dNs4wz4zc7Q5EGwseBLZqLkQ87Ehg-Gc\)](/flag/flag/watchlists/5565?destination=node/5565/quick-start&token=JV1vW4NYEZ6dNs4wz4zc7Q5EGwseBLZqLkQ87Ehg-Gc) Visibility: Registered

[Overview \(/eikon-apis/eikon-data-apis\)](/eikon-apis/eikon-data-apis)[Quick Start \(/eikon-apis/eikon-data-apis/quick-start\)](/eikon-apis/eikon-data-apis/quick-start)

Find in page...

< [Close Menu](#)[Quick Start guide for Windows](#) >[Quick Start guide for Mac OS \(/eikon-apis/eikon-data-apis/quick-start?content=43607&type=quick_start\)](/eikon-apis/eikon-data-apis/quick-start?content=43607&type=quick_start)

Last update

Jun 2018

Operating System

Windows

Interpreter

Python 2.x/3.x

4.0.36 or higher.

Eikon version

Mac OS users should refer to the Quick Start guide for MacOs (Beta version) [\(/eikon-data-apis/quick-start?content=5682&type=quick_start\)](/eikon-data-apis/quick-start?content=5682&type=quick_start)

Quick Start guide for Windows

Start with the Eikon Data API for Python in 4 steps

1. Run the TR Eikon application
2. Create an App Key for your application
3. Install the Python library for Eikon Data APIs
4. Let's have some fun !

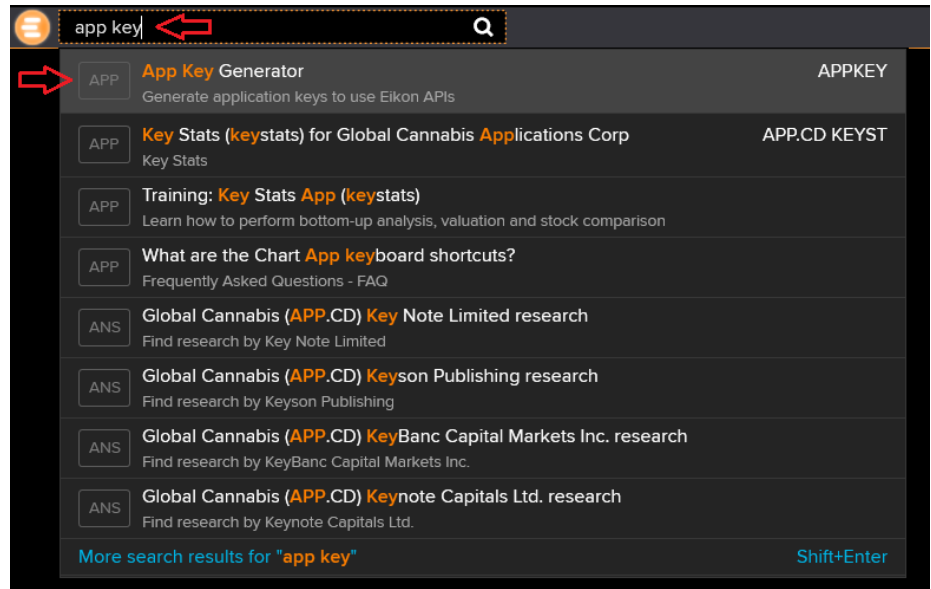
1. Run the TR Eikon application

The Eikon application integrates an API proxy that acts as an interface between the Eikon Data APIs Python library and the Eikon Platform. For this reason, the Eikon application must be running when you use the Eikon Data APIs Python library. When launched, the Eikon application starts listening for local websocket connections on port 9000 or the next available port if port 9000 is already occupied. You need to have a valid Eikon user account to be able to launch the Eikon application.

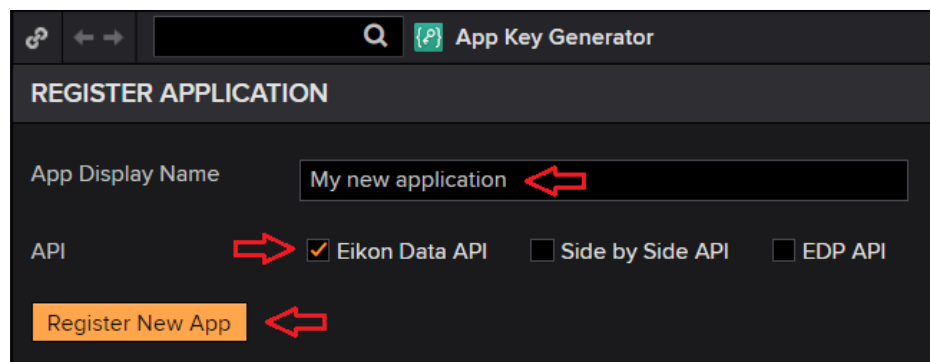
2. Create an App Key for your application

Every application using Eikon Data APIs must identify itself thanks to a Key called *Application Key* or *App Key* for short. This Key, that is unique to your application, must be created using the *App Key Generator* Eikon app that you can find in the Eikon App Library or via the toolbar of the TR Eikon Desktop application. Here is how to run it and create an *App Key* for a new application:

1. In the Eikon toolbar type "App Key" and run the *App Key Generator* app:



2. In the *App Key Generator* app indicate the name of your new application, the APIs it uses and click on the "Register New App" button:



3. Read the "Terms & Conditions" and click on the "Accept" button

Terms & Conditions

By clicking "Accept", you agree that you have read and agree to the following Thomson Reuters ("TR") terms and conditions.

- Your access and use of Eikon, including Thomson Reuters Eikon APIs, is subject to the terms and conditions of your organization's agreement with TR.
- Eikon is an information service that is licensed for individual use and requires a log in session. Sessions control entitlements and reasonable use for a single user.
- You may not share your login credentials, run any instances of Eikon on a server or use, distribute or redistribute information in any way that is inconsistent with your organization's agreement with TR.
- In addition, you may not use information in any manner that would violate rules or policies set by Third Party Providers. For example, a number of Exchanges and other Third Party Providers have introduced separate policies for use of their data for Non-Display purposes. Use of their data for Non-Display purposes usually requires a direct contract with these sources and the payment of fees directly to these sources.

Non-Display Use generally includes, but is not limited to:

- Use of data for the creation of new original works/derived data.
- Use of data to create indices.
- Use of data in electronic trading systems for the purpose of generating orders or executing transactions in an automated and/or semi-automated manner without the

Accept

Cancel

4. Copy the *App Key* that you will use later in your new application:

REGISTER APPLICATION

App Display Name

Choose a meaningful name so users can identify your app

API

☒ Eikon Data API
 ☐ Side by Side API
 ☐ EDP API

Register New App

MANAGE APPLICATIONS

App Name	ProductID	App Key
My new application	THOMSONREUTERS.COMPANY.MYNEWAPPLICATION	8e5a3ac17e0c41770e81a04877e056221b031c

3. Install the Python library for Eikon Data APIs

The Python library for Eikon Data APIs is an ease of use library, which conveniently wraps the raw message transcription between Eikon and Python and provides user friendly data retrieval calls. The data output from Python library for Eikon Data APIs is available as Pandas Dataframe or as JSON objects.

Run the following command to install the Eikon package from the Python Package Index (https://en.wikipedia.org/wiki/Python_Package_Index) (PyPI):

```
pip install eikon
```

Note: This command must be run at the prompt of the command-line interpreter (CMD). If the "Script/" folder of your Python installation is not included in your PATH environment variable, you may even have to navigate to this folder before you run the pip command.

Below is an example of the expected output

```

C:\>
C:\>cd Python36-32\Scripts
C:\Python36-32\Scripts>pip install eikon
Collecting eikon
  Using cached eikon-0.1.11-py3-none-any.whl
Collecting websocket-client (from eikon)
  Using cached websocket_client-0.46.0-py2.py3-none-any.whl
...
Installing collected packages: six, websocket-client, zope.interface, pytime, appdirs, numpy, python-dateutil, pandas, certifi, chardet, idna, urllib3, requests, eikon
Successfully installed appdirs-1.4.3 certifi-2018.1.18 chardet-3.0.4 dateutil-2.6.0 idna-2.6 numpy-1.11.1 pandas-0.19.2 pytime-0.1.1 requests-2.18.4 six-1.11.0 urllib3-1.23.1 websocket-client-0.46.0 zope.interface-4.4.2
C:\Python36-32\Scripts>

```

4. Let's have some fun !

Launch Python IDE and try the following commands.

Initialize the Eikon object. Replace '8e5a3xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx21b031c' with the App Key generated in step 2.

```

import eikon as ek
ek.set_app_key('8e5a3xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx21b031c')

```

Note: In earlier versions of the Python library for Eikon Data APIs, the **set_app_key** function was called **set_app_id**. The **set_app_id** function is now deprecated but still works. If you use an earlier version of the Python library you can call **set_app_id** to set the App Key of your application.

Great, you have imported the library and set the App Key of your application. You are now ready to send data retrieval requests.

The following instruction retrieves news headlines on Deutsche Lufthansa AG (equity RIC: LHAG.DE), between 09:00 and 18:00 GMT on the 5th of Apr 2017.

```

ek.get_news_headlines('R:LHAG.DE', date_from='2017-04-05T09:00:00', date_to='2017-04-05T18:00:00')

```

out:

	versionCreated	text	storyId
2017-04-05 14:01:51	2017-04-05 14:01:56.001	Austrian Airlines enhances Boeing 777 Jet with...	urn:newsml:reuters.com:20170405:nNRA3m80y
2017-04-05 13:12:17	2017-04-05 13:12:17.000	FOKUS 1-Tiergesundheit gibt Boehringer Schub -...	urn:newsml:reuters.com:20170405:nL5N1HD1R

	versionCreated	text	storyId
2017-04-05 10:41:12	2017-04-05 10:41:12.000	Airline services group dnata looking for more ...	urn:newsml:reuters.com:20170405:nL5N1HD26
2017-04-05 09:05:13	2017-04-05 09:05:31.004	Andreas Zerlauth Named New Head of Controlling...	urn:newsml:reuters.com:20170405:nNRA3m672
2017-04-05 09:05:06	2017-04-05 09:05:19.005	Restaurant Brands International announces agre...	urn:newsml:reuters.com:20170405:nNRA3m66r

Now, let's display the latest news story satisfying the news search expression "EU AND POL", which represents news on the European Union politics.

First, retrieve the news headlines containing the string 'world cup', then get the story ID from the response and finally request the story.

```
headlines = ek.get_news_headlines('EU AND POL',1)
story = headlines.iat[0,2]
ek.get_news_story(story)
```

The output is the HTML of the news story.

The following commands return time series of daily price history for Microsoft Corp ordinary share between 1st of Jan and 10th of Jan 2016.

```
df = ek.get_timeseries(["MSFT.O"],
                       start_date="2016-01-01",
                       end_date="2016-01-10")
df
```

out:

MSFT.O	HIGH	CLOSE	LOW	OPEN	COUNT	VOLUME
Date						
2016-01-04	54.8000	54.80	53.39	54.32	272781.0	53777963.0
2016-01-05	55.3900	55.05	54.54	54.93	180637.0	34079674.0
2016-01-06	54.4001	54.05	53.64	54.32	225856.0	39518863.0
2016-01-07	53.4850	52.17	52.07	52.70	303271.0	56564852.0
2016-01-08	53.2800	52.33	52.15	52.37	261949.0	48753969.0

The following commands retrieve fundamental data - Revenue and Gross Profit - for Google, Microsoft and Facebook

```
df, err = ek.get_data(['GOOG.O', 'MSFT.O', 'FB.O'],
                      ['TR.Revenue', 'TR.GrossProfit'])
df
```

out:

	Instrument	Revenue	Gross Profit
0	GOOG.O	90272000000	55134000000
1	MSFT.O	85320000000	52540000000
2	FB.O	27638000000	23849000000

You can specify additional parameters and request full year revenue and gross profit for the last two years scaled to millions and converted to Euros. E.g.

```
df, err = ek.get_data(['GOOG.O', 'MSFT.O', 'FB.O', 'AMZN.O', 'TWTR.K'],
                      ['TR.Revenue.date', 'TR.Revenue', 'TR.GrossProfit'],
                      {'Scale': 6, 'SDate': 0, 'EDate': -2, 'FRQ': 'FY', 'Curn': 'EUR'})
df
```

out:

	Instrument	Date	Revenue	Gross Profit
0	GOOG.O	2016-12-31T00:00:00Z	85866.726400	52443.460800
1	GOOG.O	2015-12-31T00:00:00Z	69050.621090	43116.928250
2	GOOG.O	2014-12-31T00:00:00Z	54559.726650	33634.735200
3	MSFT.O	2016-06-30T00:00:00Z	76837.485600	47316.473200
4	MSFT.O	2015-06-30T00:00:00Z	84041.390600	54370.953940
5	MSFT.O	2014-06-30T00:00:00Z	63423.691530	43645.649550
6	FB.O	2016-12-31T00:00:00Z	26289.265600	22685.168800
7	FB.O	2015-12-31T00:00:00Z	16508.281680	13868.319410
8	FB.O	2014-12-31T00:00:00Z	10305.018900	8525.241450
9	AMZN.O	2016-12-31T00:00:00Z	129350.834400	45393.166400
10	AMZN.O	2015-12-31T00:00:00Z	98532.194860	32555.237550
11	AMZN.O	2014-12-31T00:00:00Z	73561.930200	21687.989400
12	TWTR.K	2016-12-31T00:00:00Z	2406.173593	1519.426905
13	TWTR.K	2015-12-31T00:00:00Z	2042.386046	1370.879829
14	TWTR.K	2014-12-31T00:00:00Z	1159.791603	790.850268

You can also mix fundamental data with a snapshot of market data.

```
df, err = ek.get_data(['VOD.L', 'FB.O'],
                      ['TR.Revenue', 'TR.GrossProfit', 'CF_LAST'])
df
```

Out:

	Instrument	Revenue	Gross Profit	CF_LAST
0	VOD.L	55945230741	14388764346	203.1992

	Instrument	Revenue	Gross Profit	CF_LAST
1	FB.O	27638000000	23849000000	140.78

For further information visit the Tutorials
(<https://developers.thomsonreuters.com/eikon-data-apis/learning>) section



(<http://thomsonreuters.com/>)

- Feedback
(<https://developers.thomsonreuters.com/feedback?previous=node/5565/quick-start>)
- Technical Requirements (/technical-requirements)
- Copyright
(<https://www.thomsonreuters.com/en/copyright.html>)
- Cookie Policy
(<https://www.thomsonreuters.com/en/privacy-statement.html#cookies>)
- Privacy Statement
(<https://www.thomsonreuters.com/en/privacy-statement.html>)
- Terms of Use (/terms-use)
- Careers (<http://thomsonreuters.com/en/careers.html>)