

TS Parameters

<https://docs.quandl.com/docs/parameters-2#section-times-series-parameters>

database_code yes string Code identifying the database to which the dataset belongs.

dataset_code yes string Code identifying the dataset.

limit no int Use limit=n to get the first n rows of the dataset. Use limit=1 to get just the latest row.

column_index no int Request a specific column. Column 0 is the date column and is always returned. Data begins at column 1.

start_date no string yyyy-mm-dd Retrieve data rows on and after the specified start date.

end_date no string yyyy-mm-dd Retrieve data rows up to and including the specified end date.

order no string asc

desc Return data in ascending or descending order of date. Default is desc.

collapse no string none

daily

weekly

monthly

quarterly

annual Change the sampling frequency of the returned data. Default is none; i.e., data is returned in its original granularity.

diff

rdiff

rdiff_from

cumul

transform no string none

normalize Perform elementary calculations on the data prior to downloading. Default is none. Calculation options are described below.

Py TS Quandl Steps

Package pip install quandl
import quandl

Set Key quandl.ApiConfig.api_key = "YOURAPIKEY"

Time Series Call data = quandl.get("EIA/PET_RWTC_D")

Numpy Format data = quandl.get("EIA/PET_RWTC_D", returns="numpy")

Make a filtered time-series call

To set start and end dates:

data = quandl.get("FRED/GDP", start_date="2001-12-31", end_date="2005-12-31")

To request specific columns:

data = quandl.get(["NSE/OIL.1", "WIKI/AAPL.4"])

To request the last 5 rows:

data = quandl.get("WIKI/AAPL", rows=5)

Preprocess the data

To change the sampling frequency:

data = quandl.get("EIA/PET_RWTC_D", collapse="monthly")

To perform elementary calculations on the data:

data = quandl.get("FRED/GDP", transformation="rdiff")

Download an entire time-series dataset quandl.bulkdownload("ZEA")

Table Data data = quandl.get_table("MER/F1", compnumber="39102", paginate=True)

SBI India

#Libraries

```
pip install quandl
import quandl

#pip install mpl_finance
from mpl_finance import candlestick_ohlc

import matplotlib.dates as mdates
import matplotlib.pyplot as plt

import pandas as pd

from matplotlib.dates import MONDAY, DateFormatter, DayLocator, WeekdayLocator
```

#Parameters

```
start1 = datetime.datetime(2017, 1, 1)
end1 = datetime.datetime(2019, 7, 20)

register in quandl for free
key = '4D8hkYAV4WEkcTmD9LMW'
S_sbi = "BSE/BOM500112" # SBI
```

#Fetch Data

```
sbi= quandl.get(dataset=S_sbi, authtoken=key,
start_date="2019-06-01", end_date="2019-08-30")

sbi.head()
```

#Analyse

```
sbi.Close.plot() plt.show();
```

#Plots

```
data=sbi data.columns data.dtypes
data.head() data.index
data[["Open","High","Low","Close"]]
data[["Open","High","Low","Close","WAP"]].plot()
data[["Open","Close"]].plot()
```

#Format Graph Values

```
# major ticks on the mondays
mondays = WeekdayLocator(MONDAY)

# minor ticks on the days
alldays = DayLocator()

# e.g., Jan 12
weekFormatter = DateFormatter("%b %d")

# e.g., 12
dayFormatter = DateFormatter("%d")
```

#Prepare Data

```
sbi.head()

data3 = sbi[["Open", "High", "Low", "Close"]].copy()
data3.head()

date1 = "2019-8-1" date2 = "2019-8-30"

data3b = data3[(data3.index >= date1) & (data3.index <= date2)]

data3b
```

#Plot

```
fig, ax = plt.subplots(figsize=(10,8))
fig.subplots_adjust(bottom=0.2)
ax.xaxis.set_major_locator(mondays)
ax.xaxis.set_minor_locator(alldays)
ax.xaxis.set_major_formatter(weekFormatter)
#ax.xaxis.set_minor_formatter(dayFormatter)

candlestick_ohlc(ax, zip( mdates.date2num(
data3b.index.to_pydatetime()), data3b['Open'],
data3b['High'],data3b['Low'], data3b['Close']),width=0.6)

ax.xaxis_date()
ax.autoscale_view()

plt.setp(plt.gca().get_xticklabels(), rotation=45,
horizontalalignment= 'right')

plt.show();
```