Errata for Python for Finance (2nd edition,2017)

(ISBN: 978-1-78712-569-8) Publisher: Packt Publishing

Yuxing Yan[1]

11/24/2018

I thank Justin Lee, WANG Dehong, Kim JongHun, Matthew Hosseini, Sam K., and several anonymous readers for pointing out typos, errors and making some suggestions. I am sorry for the inconvenience caused to my readers. If you find new typos/errors, please let me know. A few useful links and paths.

For the fincal.py function, see the link at	http://canisius.edu/~yany/doc/fincal.pdf
For the Python data sets, the new path is	http://canisius.edu/yany/data/python/
	http://canisius.edu/~yany/data/python/list.txt
For the Python programs, the new path is	http://canisius.edu/~yany/soft/python/
For example, for p4f.pyc,	http://canisius.edu/~yany/soft/python/p4f.pyc
Or the second one	http://canisius.edu/~yany/soft/python/p4f.cpython-36.pyc
For fincal.pyc,	http://canisius.edu/~yany/soft/python/fincal.pyc
Or another link at	http://canisius.edu/~yany/soft/python/fincal.cpython-36.pyc
For the `loadYan()` function,	http://canisius.edu/~yany/soft/python/loadYan.py
	http://canisius.edu/~yany/soft/python/loadYan.py.txt
All code on Github,	https://github.com/PacktPublishing/Python-for-Finance-Second-Edition

For the SEC index files

One example: http://canisius.edu/~yany/data/sec/pickle/index1993q1.pkl (1993Q1 to 2018Q4)
http://canisius.edu/~yany/data/sec/pickle/list.txt

Python program: http://canisius.edu/~yany/soft/python/loadSECindex.py
http://canisius.edu/~yany/soft/python/loadSECindex.py.txt

1) The issue related Yahoo!Finance

Since Yahoo!Finance has changed its data structure, many old functions would not work, see one example below (on page 25)

```
import re
from matplotlib.finance import quotes_historical_yahoo_ochl
ticker='dell'
outfile=open("c:/temp/dell.txt","w")
begdate=(2013,1,1)
enddate=(2016,11,9)
p=quotes_historical_yahoo_ochl
(ticker,begdate,enddate,asobject=True,adjusted=True)
outfile.write(str(p))
outfile.close().
```

There are three solutions: 1) manually download the data first, then write a Python program to retrieve it, 2) use a fix_yahoo function, 3) use the Quandl data delivery platform.

Method I: manually download the data first, then write a Python program to retrieve it, see the code below.

[1] My email address is yany@canisius.edu. Location of this file: http://canisius.edu/~yany/doc/errataP4F.pdf, http://datayyy.com/doc/errataP4F.pdf, and https://github.com/sumhncku/errata-for-Python-for-Finance-2ed

```
Import pandas as pd
inFile='http://canisius.edu/~yany/data/ibmMonthly.csv'
df = pd.read_csv(inFile, index_col=0)
print(df.head())

                Open      High       Low     Close  Adj Close    Volume
Date
1962-01-01  7.713333  7.713333  7.003334  7.226666   0.634921   8760000
1962-02-01  7.300000  7.480000  7.093333  7.160000   0.629064   5737600
1962-03-01  7.186666  7.413333  7.070000  7.103333   0.624170   5344000
1962-04-01  7.100000  7.100000  6.000000  6.053333   0.531907  12851200
1962-05-01  6.053333  6.530000  4.733333  5.233333   0.459853  49307200
```

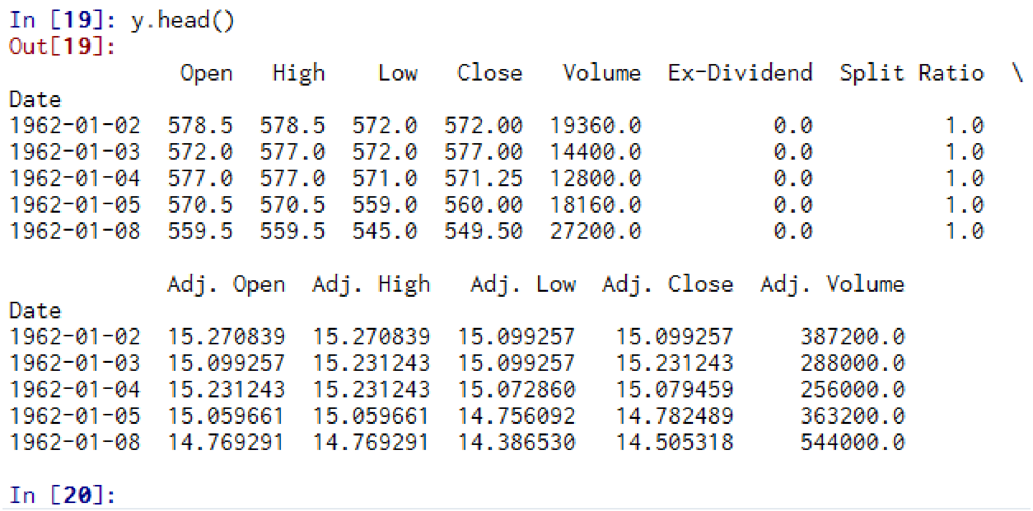
Method II: use a Python package called fix_yahoo_finance, see the code below.

```
import fix_yahoo_finance as yf
data = yf.download("IBM", start="2017-01-01", end="2017-04-30")
print(data.head())
print(data.head())
                  Open        High         Low       Close   Adj Close
Date
2017-01-03  225.039993  225.830002  223.880005  225.240005  219.079453
2017-01-04  225.619995  226.750000  225.610001  226.580002  220.382797
2017-01-05  226.270004  226.580002  225.479996  226.399994  220.207718
2017-01-06  226.529999  227.750000  225.899994  227.210007  220.995575
2017-01-09  226.910004  227.070007  226.419998  226.460007  220.266083
```

Method III: using Quandl data deliverary platform, see the code below.

```
import quandl as qd
y=qd.get("WIKI/ibm")
y.head()
```

The output is shown below.



```
In [19]: y.head()
Out[19]:
             Open   High    Low   Close   Volume  Ex-Dividend  Split Ratio  \
Date
1962-01-02  578.5  578.5  572.0  572.00  19360.0          0.0          1.0
1962-01-03  572.0  577.0  572.0  577.00  14400.0          0.0          1.0
1962-01-04  577.0  577.0  571.0  571.25  12800.0          0.0          1.0
1962-01-05  570.5  570.5  559.0  560.00  18160.0          0.0          1.0
1962-01-08  559.5  559.5  545.0  549.50  27200.0          0.0          1.0

            Adj. Open  Adj. High   Adj. Low  Adj. Close  Adj. Volume
Date
1962-01-02  15.270839  15.270839  15.099257   15.099257     387200.0
1962-01-03  15.099257  15.231243  15.099257   15.231243     288000.0
1962-01-04  15.231243  15.231243  15.072860   15.079459     256000.0
1962-01-05  15.059661  15.059661  14.756092   14.782489     363200.0
1962-01-08  14.769291  14.769291  14.386530   14.505318     544000.0

In [20]:
```

Note: see the next comment for the instruction on how to download the Quandl package.

2) How to install the Quandl package?

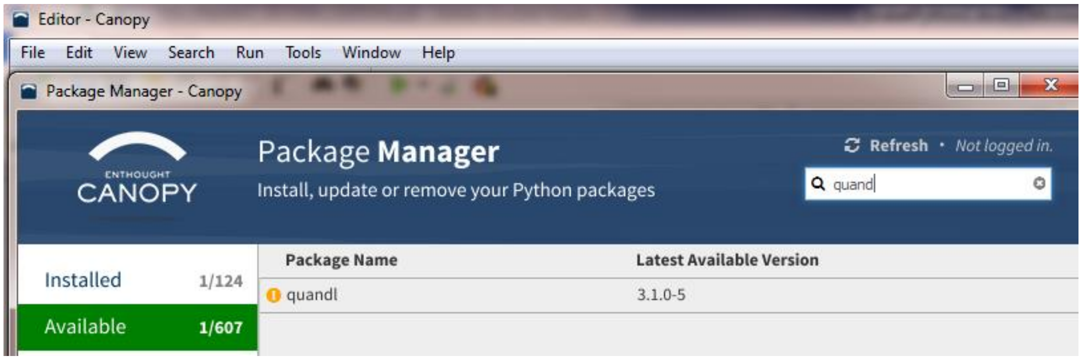
Method I:

```
conda install quandl
```

Method II:

```
pip install quandel
```

If using Canopy, see the image below.



Install, update or remove your Python packages

Package Name	Latest Available Version
quandl	3.1.0-5

Help: https://docs.quandl.com/

3) Chapter 1, page 17

From

```
>>> import pandas as pd
>>>url=url='http://canisius.edu/~yany/data/ibm.csv'
```

To

```
>>> import pandas as pd
>>> url='http://canisius.edu/~yany/data/ibm.csv'
```

4) Chapter 2, page 47

For the old code, see below.

```
import datetime
import matplotlib.pyplot as plt
from matplotlib.finance import quotes_historical_yahoo_ochl
from matplotlib.dates import MonthLocator,DateFormatter
ticker='AAPL'
begdate= datetime.date( 2012, 1, 2 )
```

To

```
import datetime
import fix_yahoo_finance as yf
import matplotlib.pyplot as plt
from matplotlib.dates import MonthLocator,DateFormatter
x=yf.download("AAPL", start="2012-01-02", end="2013-12-05")
months = MonthLocator(range(1,13), bymonthday=1, interval=3)
monthsFmt = DateFormatter("%b '%Y")
if len(x) == 0:
   print ('Found no quotes')
   raise SystemExit
dates=x.index
closes=x['Adj Close']
fig, ax = plt.subplots()
ax.plot_date(dates, closes, '-')
ax.xaxis.set_major_locator(months)
ax.xaxis.set_major_formatter(monthsFmt)
ax.autoscale_view()
ax.grid(True)
fig.autofmt_xdate()
```

5) Chapter 2, pages 48 and 50

For the related code, see below.

```
monthsFmt = DateFormatter("%b '%Y")
x = quotes_historical_yahoo_ochl(ticker, begdate, enddate)
```

To

See comments 1) and 2)

6) Chapter 2, page 52

From

The columns() function defines the names of those columns

To

The 'columns' input variable defines the names of those columns

7) Chapter 2, pages 54-55

From

```
import pandas as pd
import numpy as np
np.random.seed(123) # fix the random numbers
x=np.arange(1, 10.1, .25)**2
n=np.size(x)
y = pd.Series(x + np.random.randn(n))
bad=np.array([4,13,14,15,16,20,30]) # generate a few missing values
x[bad] = np.nan # missing code is np.nan
methods = ['linear', 'quadratic', 'cubic']
df = pd.DataFrame({m: x.interpolate(method=m) for m in methods})
df.plot()
```

To

```
import numpy as np
import pandas as pd
np.random.seed(123) # fix the random numbers
x=np.arange(1, 10.1, .25)**2
n=np.size(x)
y = pd.Series(x + np.random.randn(n))
bad=np.array([4,13,14,15,16,20,30]) # generate a few missing values
y[bad] = np.nan # missing code is np.nan
methods = ['linear', 'quadratic', 'cubic']
df = pd.DataFrame({m: y.interpolate(method=m) for m in methods})
df.plot()
```

8) Chapter 2, page 55

From

that is, an absolute address, we have the following code: df.to_pickle('test.pkl')

To

that is, an absolute address, we have the following code: df.to_pickle('c:/temp/test.pkl')

9) Chapter 2, page 56

```
>>> y
key value
0  B  2
1  D  3
2  D  4
3  E  6numpy as np
```

→

```
>>> y
key value
0  B  2
1  D  3
2  D  4
3  E  6
```

10) Chapter 2, page 60

From

```
import pandas_datareader.data as web
df=web.get_data_google("ibm")
```

To

```
import fix_yahoo_finance as yf
data = yf.download("IBM", start="2018-01-01", end="2018-11-13")
print(data.head())
```

11) Chapter 2, page 62

From

```
http://canisius.edu/~yany/fincal.cpython-35.pyc
```

To

```
http://canisius.edu/~yany/soft/python/fincal.cpython-36.pyc
or
http://canisius.edu/~yany/soft/python/fincal.pyc
```

12) Chapter 3, page 82

From

Appendix B shows how to download it

To

Appendix D shows how to download it

13) Chapter 3, page 83

```
with the sign
convention:
>>>import fincal
```

→

```
with the sign
convention:
>>> import fincal
```

>>> fincal.pv_f(0.1,1100)		>>> fincal.pv_f(0.1,1,100)

14) Chapter 3, page 84

From

see the Writing your own financial calculator written in Python section and Appendix H.

To

see the Writing your own financial calculator in Python section and Appendix G.

15) Chapter 3, page 85

From

If the monthly rate is 0.25% and he plans to pay back $200 per month

To

If the monthly rate is 1.2% and he plans to pay back $200 per month

16) Chapter 3, page 100

From

If the same cash flow happens at the same interval forever, it is called perpetuity. If the discount rate is a constant and the first cash flows happens at the end of the first period, its present value has the following.

To

If the same cash flow happens at the same interval forever, it is called perpetuity. If the discount rate is a constant and the first cash flows happens at the end of the first period, its present value has the following equation.

PV(Perpetuity) = C / R

17) Chapter 3, page 107

From

Richard has just finished a very difficult sophomore

(second) year, including taking several finance courses. Richard would very much like to take a long vacation.

To

Peter has just finished a very difficult sophomore

(second) year, including taking several finance courses. Peter would very much like to take a long vacation.

18) Chapter 4, page 123

From

The following graph shows how IBM's returns distributed plus a normal distribution. The price moment is shown on the right and its Python program is included in Appendix A:

To

The following graph shows how IBM's returns distributed plus a normal distribution and its Python program is included in Appendix A. The price movement is shown on the right and its Python program is included in Appendix C:

19) Chapter 4, page 127

From

```
import pandas_datareader.data as getData
vix = DataReader("VIXCLS", "fred")
```

To

```
import pandas_datareader.data as getData
vix = getData.DataReader("VIXCLS", "fred")
```

20) Chapter 4, page 124

From

The so-called candle-stick picture could be used to vividly present a stock price or trading volume, as shown in the following screenshot. The corresponding Python
program is in Appendix C:

To

The so-called candle-stick picture could be used to vividly present a stock price or trading volume, as shown in the following screenshot. The corresponding Python
program is in Appendix B:

21) Chapter 4, page 125

From

The following screenshot shows a stock's intraday moment. The related Python program is included in Appendix C:

To

The following screenshot shows a stock's intraday movement. The related Python program is included in Appendix D:

22) Chapter 4, page 131

From

http://canisius.edu/~yany/ loadYan.py

To

http://canisius.edu/~yany/soft/python/loadYan.py
http://datayyy.com/python/loadYan.py.txt
http://canisius.edu/~yany/python/loadYan.py

23) Chapter 4, page 133

From

indexDaily.pkl Index file with a monthly frequency

To

indexDaily.pkl Index file with a daily frequency

24) Chapter 4, page 139 (candle stick image)

For the new code, see below.

```
from math import pi
import pandas as pd
from bokeh.sampledata.stocks import MSFT
from bokeh.plotting import figure, show, output_file

df = pd.DataFrame(MSFT)[:50]
df["date"] = pd.to_datetime(df["date"])
mids = (df.open + df.close)/2
spans = abs(df.close-df.open)
inc = df.close > df.open
dec = df.open > df.close
w = 12*60*60*1000 # half day in ms
output_file("c://temp/candlestick.html", title="candlestick.py example")
TOOLS = "pan,wheel_zoom,box_zoom,reset,save"
p = figure(x_axis_type="datetime", tools=TOOLS, plot_width=1000,
toolbar_location="left")
p.segment(df.date, df.high, df.date, df.low, color="black")
p.rect(df.date[inc], mids[inc], w, spans[inc], fill_color="#D5E1DD",
line_color="black")
p.rect(df.date[dec], mids[dec], w, spans[dec], fill_color="#F2583E",
line_color="black")
#p.title = "MSFT Candlestick"
p.xaxis.major_label_orientation = pi/4
p.grid.grid_line_alpha=0.3
#show(p)  # open a browser
```

25) Chapter 5, page 152

Bank A offers an annual rate of 8% compounded semi-annually

Bank A offers an annual percentage rate of 8% compounded semi-annually

26) Chapter 5, page 154

>>> APR2Rm(0.08,2,12) 0.008164846051901042	→	>>> APR2Rm(0.08,2,12) 0.006558196936559346

27) Chapter 5, page 155

>>>Rs=(1+0.05/2)**(2/12)-1	→	>>>Rs=(1+0.05/2)**(2/12)-1

>>>Rs*2 0.008247830930288469		>>>Rs*12 0.049486985581730814

28) Chapter 5, page 168

$YTM = (\frac{FV}{PV})^{\frac{1}{n}}$	→	$YTM = (\frac{FV}{PV})^{\frac{1}{n}} - 1$

29) Chapter 5, page 170

From

http://canisius.edu/~yany/fincal.cpython-35.pyc

To

http://canisius.edu/~yany/soft/python/fincal.cpython-36.pyc
or
http://canisius.edu/~yany/soft/python/fincal.pyc

30) Chapter 6, page 189

From

Now let's look at how to estimate the beta (market risk) for Microsoft

To

Now let's look at how to estimate the beta (market risk) for IBM

31) Chapter 6, page 189

ticker='MSFT'	→	ticker='IBM'

32) Chapter 6, page 191

From

The output for Walmart's beta (market risk) is as follows:

To

The output for IBM's beta (market risk) is as follows:

33) Chapter 6, page 195

From

http://canisius.edu/~yany/python/yanMonthly.pkl

To

http://canisius.edu/~yany/data/python/yanMonthly.pkl

34) Chapter 6, page 198

From

```
from
….
f.close()
```

To [note the data set at : http://canisius.edu/~yany/python/callsFeb2014.pkl]

```
import pandas as pd
infile="c:/temp/callsFeb2014.pkl"
outFile=open("c:/temp/callsFeb2014.csv","w")
calls=pd.read_pickle(infile)
calls.to_csv(outFile,index=False)
```

35) Chapter 6, page 199

From

The following program first retrieves IBM price data, and then saves it as a .csv file under c:/temp:

To

The following program first retrieves IBM price data, and then saves it as a .xlsx file under c:/temp:

36) Chapter 6, page 202

From

# lstrip() would remove spaces before and the end of string

# rstrip() would remove spaces before and the end of string

To

# lstrip() would remove leading white spaces of string

# rstrip() would remove trailing white spaces of string

37) Chapter 6, page 204

From

download Canopy, such as winders 32-bit

To

download Canopy, such as Windows 32-bit

38) Chapter 6, page 206

From

After clicking the green bottom, we can run the program:

To

After clicking the green button, we can run the program:

39) Chapter 7,

From

https://github.com/PacktPublishing/Python-for-Finance-Second-Edition/blob/master/Chapter07/c7_01_3factor_model.py

To

http://canisius.edu/~yany/python/c7_01_3factor_model2.py.txt

40) Chapter 7, page 217

`y = df['Adj.Close']`	→	y = df['Adj Close']

41) Chapter 7, page 220

From

```
Next, we show how to run a Fama-French three-factor regression using 5-year
monthly data. The added twist is that the historical price data is
downloaded first. Then we calculate monthly returns and convert them to
monthly ones
```

To

Next, we show how to run a Fama-French three-factor regression using 5-year daily data. The added twist is that the historical price data is downloaded first. Then we calculate daily returns and convert them to monthly ones

42) Chapter 7, page 235

From

ffDaily.pkl Fama-French-Carhart daily four factors

ffcDaily.pkl Fama-French daily five factors

ffDaily5.pkl Fama-French monthly four factors

To

ffDaily.pkl Fama-French daily three factors

ffcDaily.pkl Fama-French-Carhart daily four factors

ffDaily5.pkl Fama-French daily five factors

43) Chapter 9, page 314

From

# function 4: for given n-1 weights, return a negative Sharpe ratio def negative_treynor_n_minus_1_stock(w):

To

# function 4: for given n-1 weights, return a negative Treynor ratio def negative_treynor_n_minus_1_stock(w):

44) Chapter 10, page 379 (Volatility simile and skewness)

Issue: quotes_historical_yahoo_ochl is no longer working.

The original program:

http://canisius.edu/~yany/python/c10_37_volatility_smile.txt

New program.

http://canisius.edu/~yany/python/volatility_smile_using_quandl.py
http://canisius.edu/~yany/python/volatility_smile_using_quandl.py.txt (easy to view)

45) Chapter 10, page 379

Issue: how to get call options data

Method I: download yourself. Below I use IBM call options data as an example.

Step 1: go to http://finance.yahoo.com
Step 2: enter IBM
Step 3: click "Options",
https://finance.yahoo.com/quote/IBM/options?p=IBM
Step 4: manually copy and paste

Method II: download the text file from my website.

http://canisius.edu/~yany/data/callsIBM3Aug2018.txt

Method III: download a pickle file from my website.

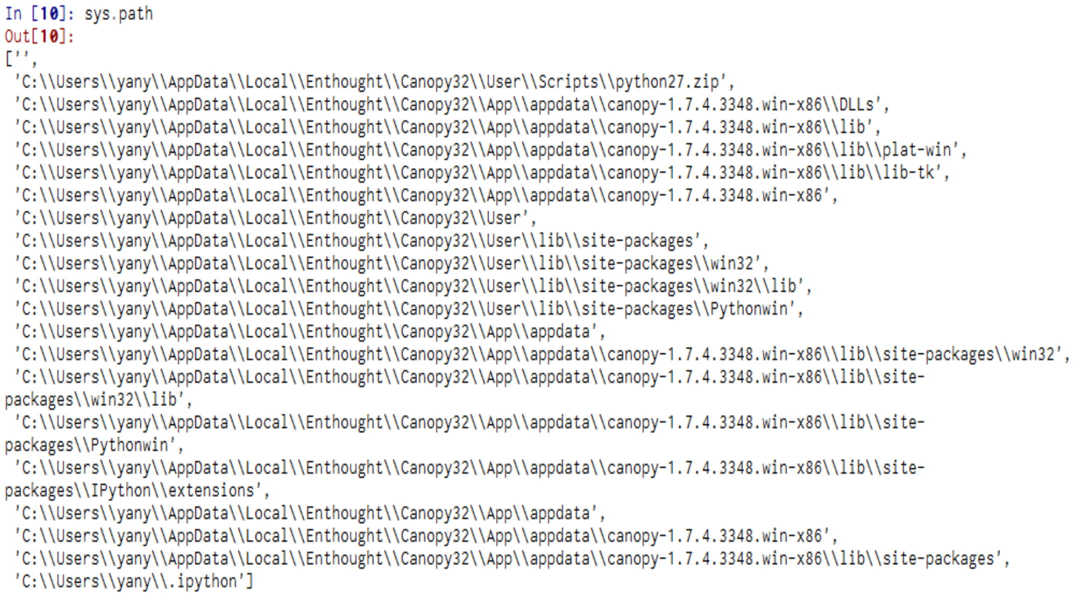
http://canisius.edu/~yany/python/callsIBM3Aug2018.pkl

99) Pages 170, 191, 211 etc.: How to call p4f module

First, you could download p4f.pyc at http://canisius.edu/~yany/python/p4f.pyc

Method I:

Step 1: find out all directories the Python software could access by using sys.path command to see a list of directories that Python software could access.

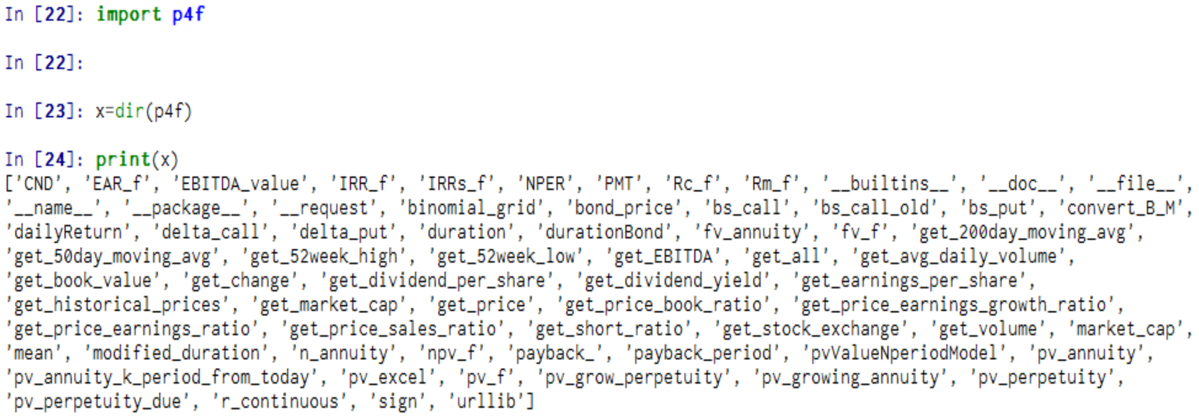


```
In [10]: sys.path
Out[10]:
['',
 'C:\\Users\\yany\\AppData\\Local\\Enthought\\Canopy32\\User\\Scripts\\python27.zip',
 'C:\\Users\\yany\\AppData\\Local\\Enthought\\Canopy32\\App\\appdata\\canopy-1.7.4.3348.win-x86\\DLLs',
 'C:\\Users\\yany\\AppData\\Local\\Enthought\\Canopy32\\App\\appdata\\canopy-1.7.4.3348.win-x86\\lib',
 'C:\\Users\\yany\\AppData\\Local\\Enthought\\Canopy32\\App\\appdata\\canopy-1.7.4.3348.win-x86\\lib\\plat-win',
 'C:\\Users\\yany\\AppData\\Local\\Enthought\\Canopy32\\App\\appdata\\canopy-1.7.4.3348.win-x86\\lib\\lib-tk',
 'C:\\Users\\yany\\AppData\\Local\\Enthought\\Canopy32\\App\\appdata\\canopy-1.7.4.3348.win-x86',
 'C:\\Users\\yany\\AppData\\Local\\Enthought\\Canopy32\\User',
 'C:\\Users\\yany\\AppData\\Local\\Enthought\\Canopy32\\User\\lib\\site-packages',
 'C:\\Users\\yany\\AppData\\Local\\Enthought\\Canopy32\\User\\lib\\site-packages\\win32',
 'C:\\Users\\yany\\AppData\\Local\\Enthought\\Canopy32\\User\\lib\\site-packages\\win32\\lib',
 'C:\\Users\\yany\\AppData\\Local\\Enthought\\Canopy32\\User\\lib\\site-packages\\Pythonwin',
 'C:\\Users\\yany\\AppData\\Local\\Enthought\\Canopy32\\App\\appdata',
 'C:\\Users\\yany\\AppData\\Local\\Enthought\\Canopy32\\App\\appdata\\canopy-1.7.4.3348.win-x86\\lib\\site-packages\\win32',
 'C:\\Users\\yany\\AppData\\Local\\Enthought\\Canopy32\\App\\appdata\\canopy-1.7.4.3348.win-x86\\lib\\site-
packages\\win32\\lib',
 'C:\\Users\\yany\\AppData\\Local\\Enthought\\Canopy32\\App\\appdata\\canopy-1.7.4.3348.win-x86\\lib\\site-
packages\\Pythonwin',
 'C:\\Users\\yany\\AppData\\Local\\Enthought\\Canopy32\\App\\appdata\\canopy-1.7.4.3348.win-x86\\lib\\site-
packages\\IPython\\extensions',
 'C:\\Users\\yany\\AppData\\Local\\Enthought\\Canopy32\\App\\appdata',
 'C:\\Users\\yany\\AppData\\Local\\Enthought\\Canopy32\\App\\appdata\\canopy-1.7.4.3348.win-x86',
 'C:\\Users\\yany\\AppData\\Local\\Enthought\\Canopy32\\App\\appdata\\canopy-1.7.4.3348.win-x86\\lib\\site-packages',
 'C:\\Users\\yany\\.ipython']
```

Step 2: copy p4f.pyc file to one of the above subdirectories. For me, I coped it to C:\Users\yany\AppData\Local\Enthought\Canopy32\User

Method II: assume the download file is under c:/temp/. Add the directory using sys.path.append() function, see below.

```
>>>sys.path.append("c:/temp")
```



```
In [22]: import p4f

In [22]:

In [23]: x=dir(p4f)

In [24]: print(x)
['CND', 'EAR_f', 'EBITDA_value', 'IRR_f', 'IRRs_f', 'NPER', 'PMT', 'Rc_f', 'Rm_f', '__builtins__', '__doc__', '__file__',
'__name__', '__package__', '__request', 'binomial_grid', 'bond_price', 'bs_call', 'bs_call_old', 'bs_put', 'convert_B_M',
'dailyReturn', 'delta_call', 'delta_put', 'duration', 'durationBond', 'fv_annuity', 'fv_f', 'get_200day_moving_avg',
'get_50day_moving_avg', 'get_52week_high', 'get_52week_low', 'get_EBITDA', 'get_all', 'get_avg_daily_volume',
'get_book_value', 'get_change', 'get_dividend_per_share', 'get_dividend_yield', 'get_earnings_per_share',
'get_historical_prices', 'get_market_cap', 'get_price', 'get_price_book_ratio', 'get_price_earnings_growth_ratio',
'get_price_earnings_ratio', 'get_price_sales_ratio', 'get_short_ratio', 'get_stock_exchange', 'get_volume', 'market_cap',
'mean', 'modified_duration', 'n_annuity', 'npv_f', 'payback_', 'payback_period', 'pvValueNperiodModel', 'pv_annuity',
'pv_annuity_k_period_from_today', 'pv_excel', 'pv_f', 'pv_grow_perpetuity', 'pv_growing_annuity', 'pv_perpetuity',
'pv_perpetuity_due', 'r_continuous', 'sign', 'urllib']
```