Python (#3064) Project Report
Stock Price Plotting/Predicting Service

Zhihong (Jill) Liu

Jill.liu2017@gmail.com

Mar 16, 2017

Introduction

This python project provides a stock price predicting service for any valid stock symbols including public companies or exchange traded funds (ETFs) through a user friendly interfaces in a web browser or through a Window's command line.

- It retrieves the historical stock price data for the requested company from Yahoo Finance and displays the scattered data along with linear-fit line and polynomial-fit line using linear and polynomial regression algorithm.
- It predicts stock price for a selected date using polynomial regression.

User can access the service either through web browser on http://localhost:5000 or through Window's command line.

- Through web browser:
  Step 1> Start the python entry program on command line:
  ```
  python plt.py
  ```
  Step 2> When message "`* Running on http://127.0.0.1:5000/`" shows up, open a web browser and type http://localhost:5000 in the address bar.
  Step 3> When the page is loaded, fill in the corresponding form and click the button next to it to get the result.

- Through command line:
  Step 1> Start the python entry program on command line:
  ```
  python runPgm.py
  ```
  Step 2> Follow the instructions on command prompt to input data and enter, the result will be returned accordingly.

Requirements

Following modules are required to be installed:

- `requests`
- `csv`
- `pandas`
- `datetime`

- numpy
- sklearn
- matplotlib
- StringIO
- flask

This program is developed and verified on windows 10.0.14393 using python 2.7.11.

Program Description

This project exercises the following python elements:

1. Use data structure list, dictionary
2. Functions
3. Classes
4. Importing external modules
5. Error checks using try-except
6. File input and output
7. Decorators

This project includes three python modules and one html file:

- main.py:
  This file contains main classes and functions for the project.

  Classes:

  - *StockPredictor(object)*

    This is the main class which handles historical stock data retrieving, data plotting and price predicting when service is requested through web browser.

    Class Methods:

    getData():

    To retrieve historical data from Yahoo Finance for a particular company.

    prepPlotAndPrediction():

    To prepare data for stock price plotting and predicting.

    calculatePrediction():

    To generate the predicted stock price using polynomial regression.

    createPlot():

    To create the historical stock price chart.

    getPrediction():

    To return the predicted stock price to user.

`getName()`:

To retrieve the input company name.

`getRange()`:

To retrieve the input historical stock data range.

- `StockPredictorCmdln(StockPredictor)`

  It's a child class of `StockPredictor()`. It overrides method `createPlot()` to create chart when the program is started on Window's command line.

Functions:

- `printException()`: To print exception information.
- `validateDate()`: To validate the input stock date range.

➢ plt.py:
This file is the entry program to start the service on a web browser. It creates a webpage on http://localhost:5000, then creates a stock price chart or predicts stock price based on the user input in the forms.

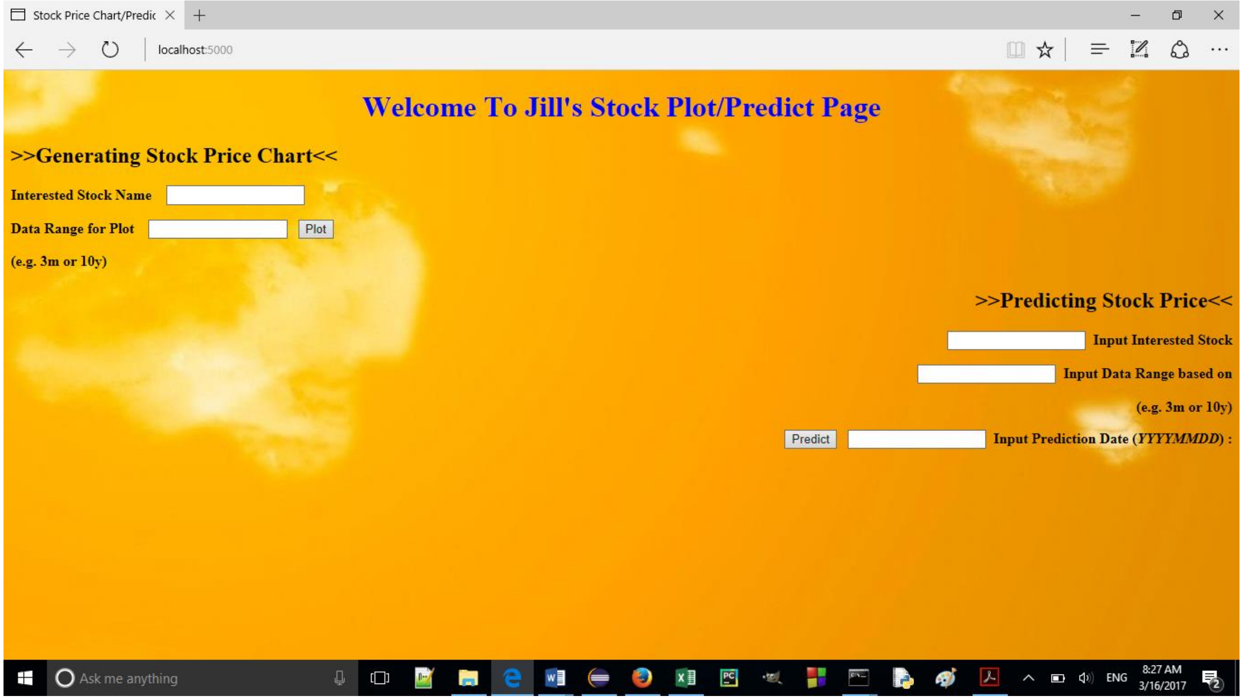
➢ runPgm.py:
This program is the entry program to start the service from Window's command line. It interacts with user through command line, creates a stock price chart and predicts stock price based on the user input on the command line.

➢ form.html:
This is the html page that appears in web browser on address http://localhost:5000 when user starts the service in a browser.

Program Screen Shots

I. Starting service in web browser:

>>Starting page:



Welcome To Jill's Stock Plot/Predict Page

>>Generating Stock Price Chart<<

Interested Stock Name

Data Range for Plot

(e.g. 3m or 10y)

>>Predicting Stock Price<<

Input Interested Stock

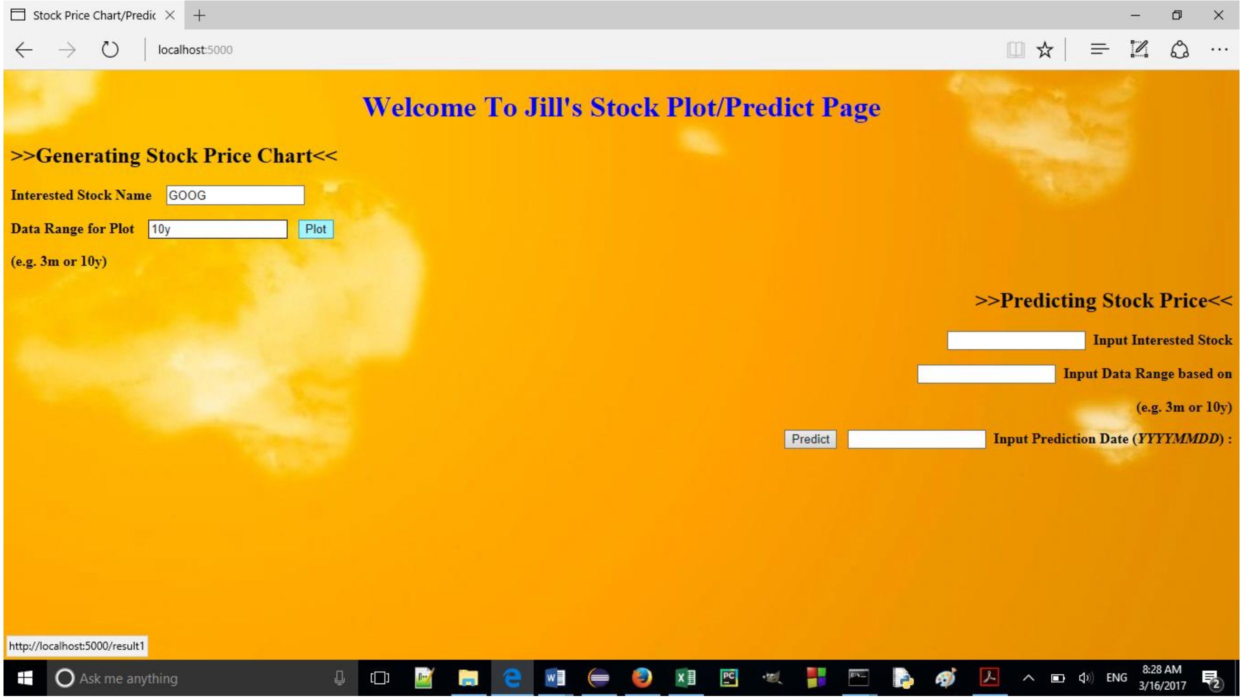
Input Data Range based on

(e.g. 3m or 10y)

Input Prediction Date (*YYYYMMDD*) :

>> Generating Stock Price Chart:

Before:



Welcome To Jill's Stock Plot/Predict Page

>>Generating Stock Price Chart<<

Interested Stock Name GOOG

Data Range for Plot 10y

(e.g. 3m or 10y)

>>Predicting Stock Price<<

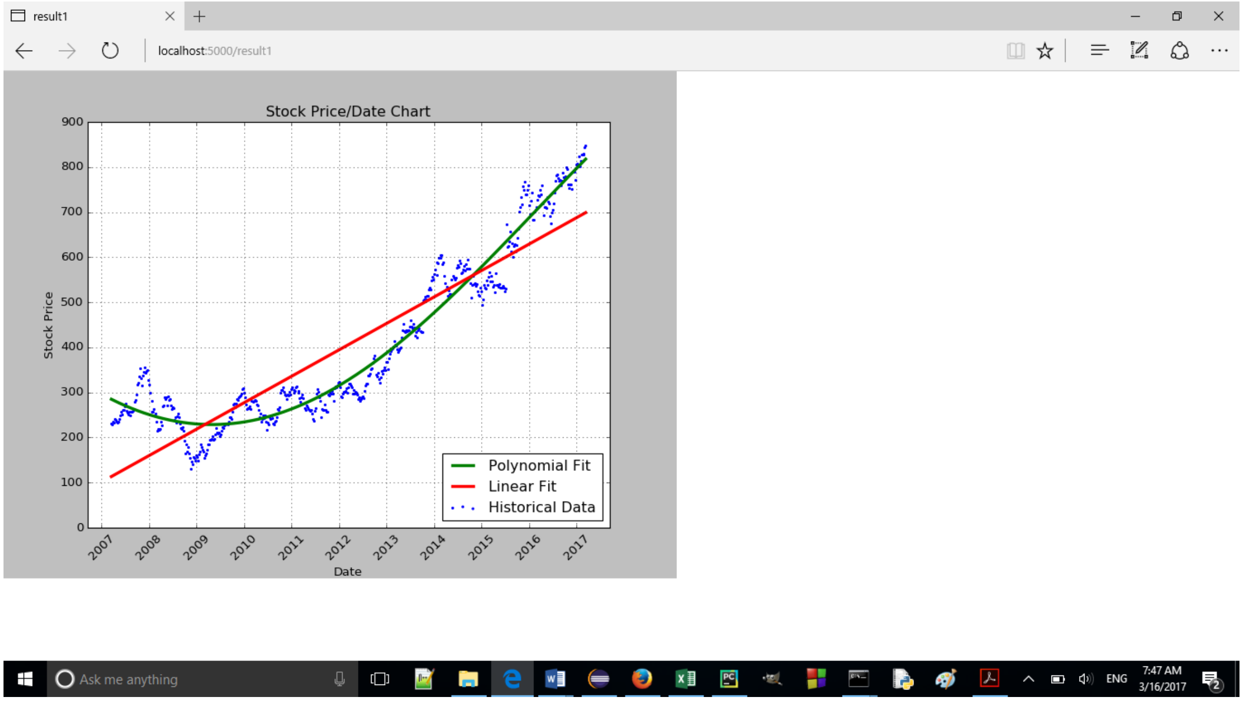
Input Interested Stock

Input Data Range based on

(e.g. 3m or 10y)

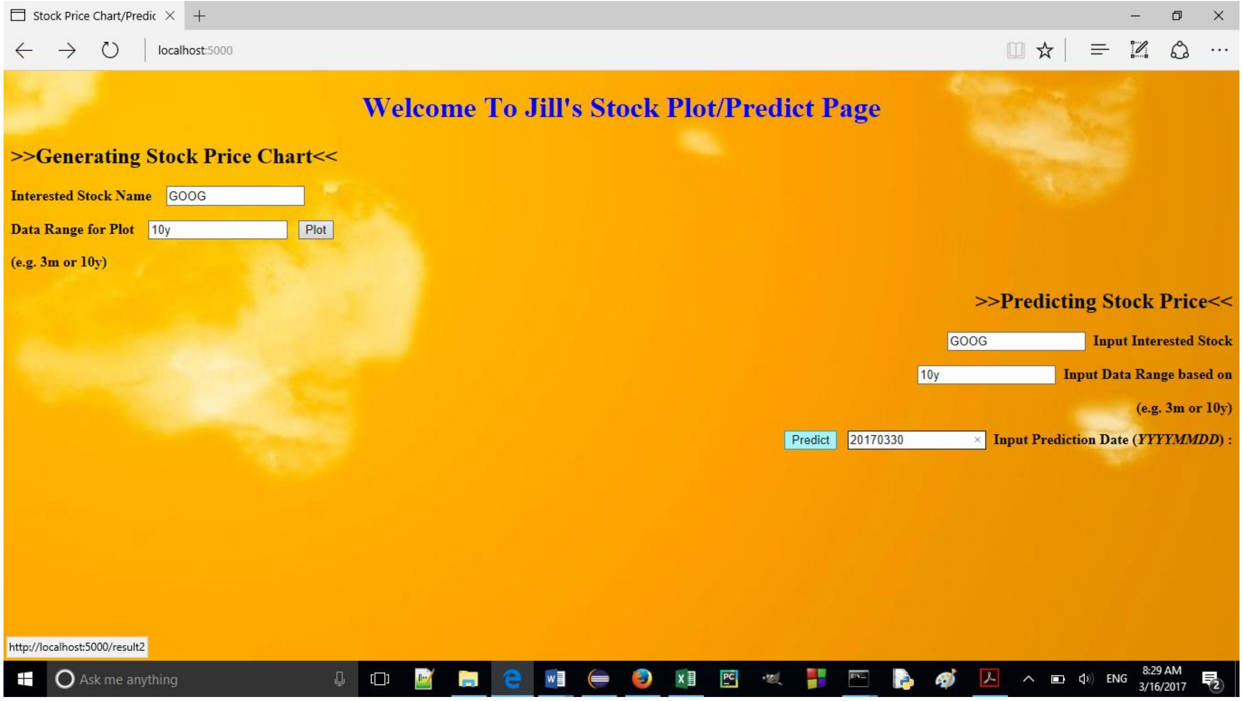
Input Prediction Date (*YYYYMMDD*) :

After:



>> Predicting Stock Price:

Before:



Welcome To Jill's Stock Plot/Predict Page

>>Generating Stock Price Chart<<

Interested Stock Name GOOG

Data Range for Plot 10y Plot

(e.g. 3m or 10y)

>>Predicting Stock Price<<

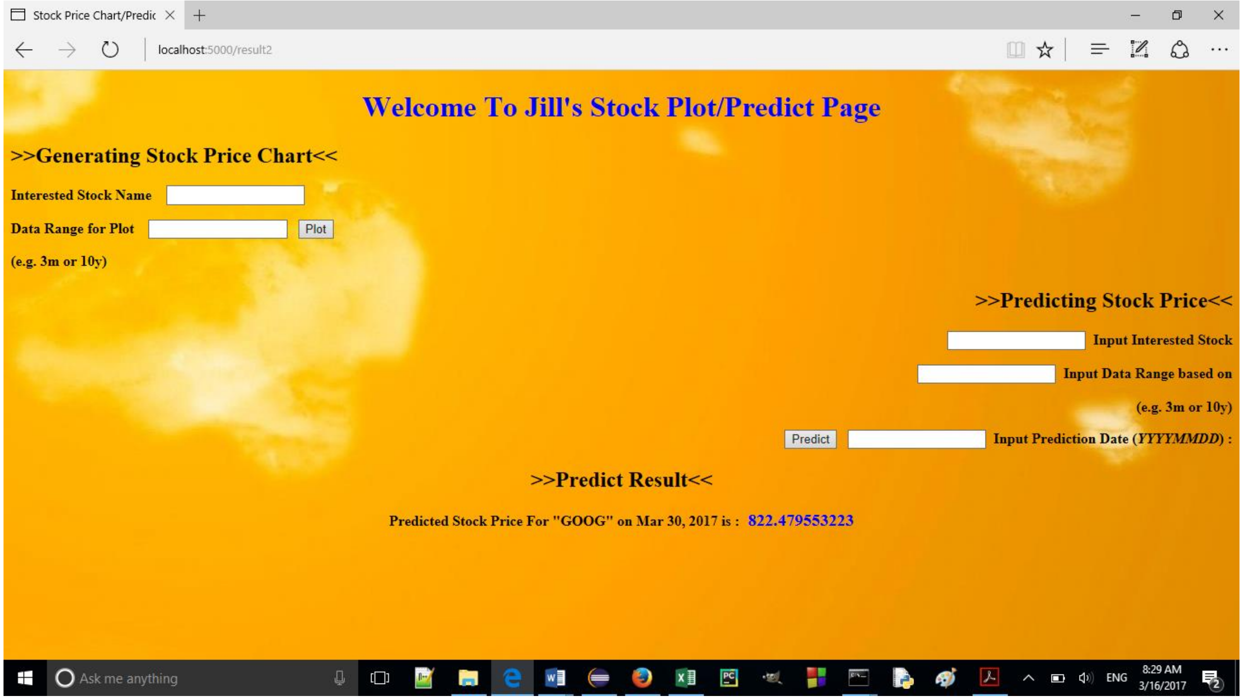
GOOG Input Interested Stock

10y Input Data Range based on

(e.g. 3m or 10y)

Predict 20170330 Input Prediction Date (*YYYYMMDD*) :

After:



Welcome To Jill's Stock Plot/Predict Page

>>Generating Stock Price Chart<<

Interested Stock Name

Data Range for Plot

(e.g. 3m or 10y)

>>Predicting Stock Price<<

Input Interested Stock

Input Data Range based on

(e.g. 3m or 10y)

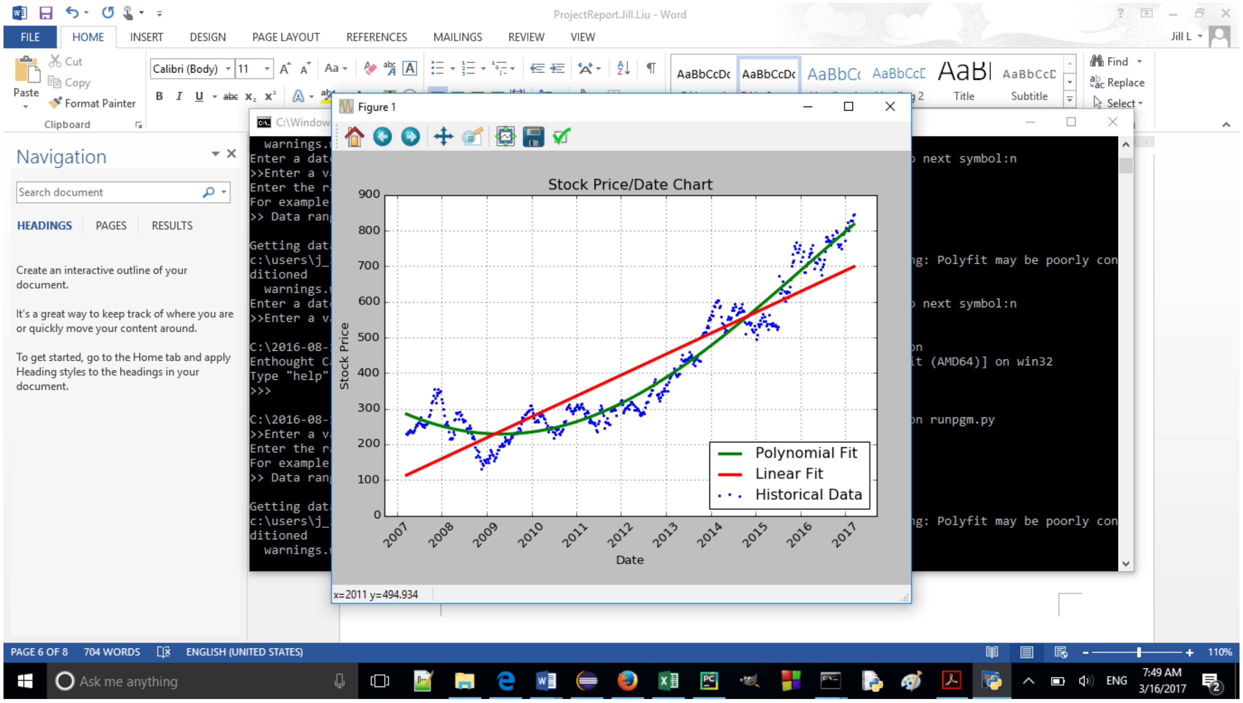
Input Prediction Date (*YYYYMMDD*) :

>>Predict Result<<

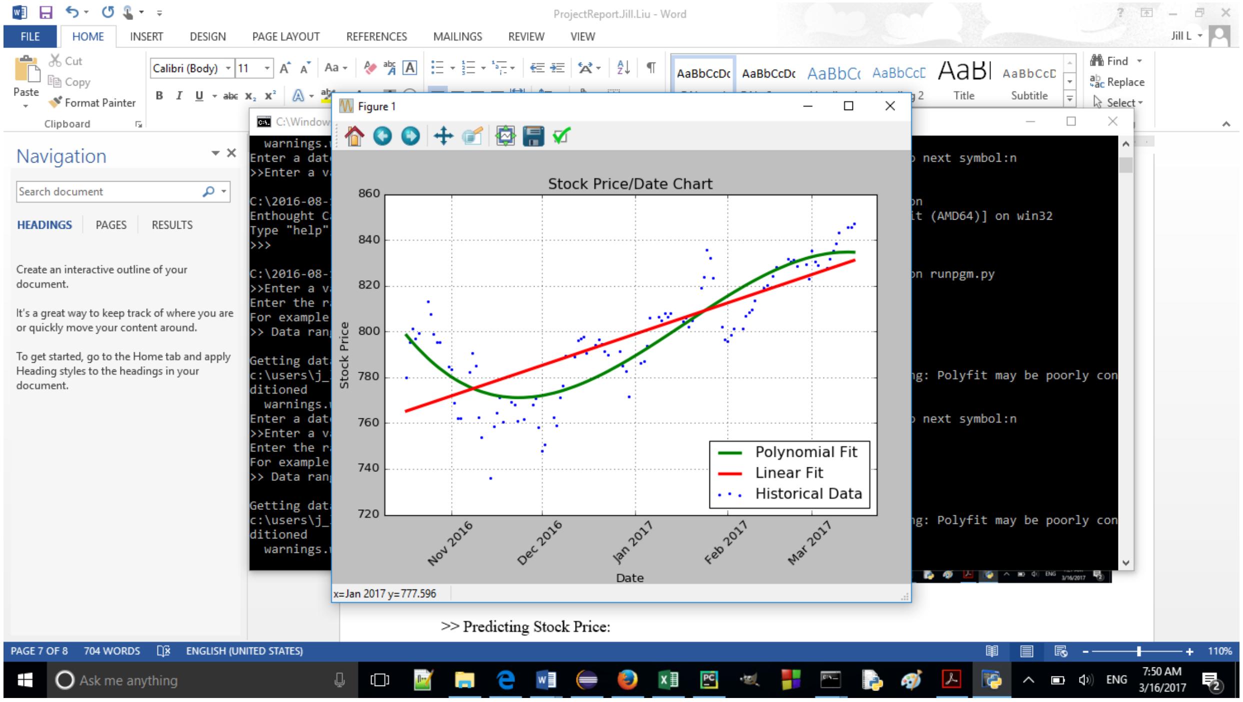
Predicted Stock Price For "GOOG" on Mar 30, 2017 is : 822.479553223

II. Starting service from command line:

>> Generating Stock Price Chart:



```
  warnings.
Enter a dat                                              next symbol:n
>>Enter a v
Enter the r
For example
>> Data ran

Getting dat
c:\users\j_                                             ng: Polyfit may be poorly con
ditioned
  warnings.
Enter a dat                                              next symbol:n
>>Enter a v

C:\2016-08-                                             on
Enthought C                                             it (AMD64)] on win32
Type "help"
>>>

C:\2016-08-                                             on runpgm.py
>>Enter a v
Enter the r
For example
>> Data ran

Getting dat
c:\users\j_                                             ng: Polyfit may be poorly con
ditioned
  warnings.
```



Create an interactive outline of your document.

It's a great way to keep track of where you are or quickly move your content around.

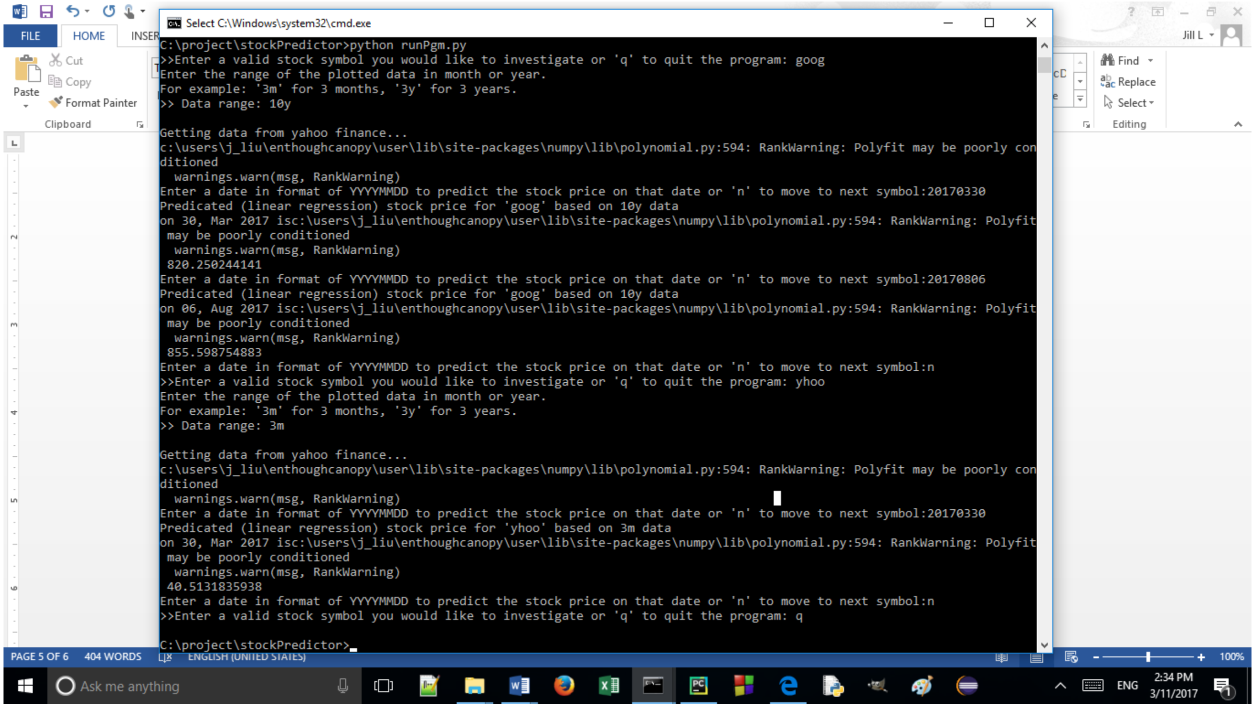
To get started, go to the Home tab and apply Heading styles to the headings in your document.

```
  warnings.
Enter a dat
>>Enter a v

C:\2016-08-
Enthought C
Type "help"
>>>

C:\2016-08-
>>Enter a v
Enter the r
For example
>> Data ran

Getting dat
c:\users\j_
ditioned
  warnings.
Enter a dat
>>Enter a v
Enter the r
For example
>> Data ran

Getting dat
c:\users\j_
ditioned
  warnings.
```

```
 next symbol:n

on
it (AMD64)] on win32

on runpgm.py

ng: Polyfit may be poorly con

 next symbol:n

ng: Polyfit may be poorly con
```

>> Predicting Stock Price:

>> Predicting Stock Price:



```
C:\project\stockPredictor>python runPgm.py
>>Enter a valid stock symbol you would like to investigate or 'q' to quit the program: goog
Enter the range of the plotted data in month or year.
For example: '3m' for 3 months, '3y' for 3 years.
>> Data range: 10y

Getting data from yahoo finance...
c:\users\j_liu\enthoughcanopy\user\lib\site-packages\numpy\lib\polynomial.py:594: RankWarning: Polyfit may be poorly con
ditioned
  warnings.warn(msg, RankWarning)
Enter a date in format of YYYYMMDD to predict the stock price on that date or 'n' to move to next symbol:20170330
Predicated (linear regression) stock price for 'goog' based on 10y data
on 30, Mar 2017 isc:\users\j_liu\enthoughcanopy\user\lib\site-packages\numpy\lib\polynomial.py:594: RankWarning: Polyfit
 may be poorly conditioned
  warnings.warn(msg, RankWarning)
 820.250244141
Enter a date in format of YYYYMMDD to predict the stock price on that date or 'n' to move to next symbol:20170806
Predicated (linear regression) stock price for 'goog' based on 10y data
on 06, Aug 2017 isc:\users\j_liu\enthoughcanopy\user\lib\site-packages\numpy\lib\polynomial.py:594: RankWarning: Polyfit
 may be poorly conditioned
  warnings.warn(msg, RankWarning)
 855.598754883
Enter a date in format of YYYYMMDD to predict the stock price on that date or 'n' to move to next symbol:n
>>Enter a valid stock symbol you would like to investigate or 'q' to quit the program: yhoo
Enter the range of the plotted data in month or year.
For example: '3m' for 3 months, '3y' for 3 years.
>> Data range: 3m

Getting data from yahoo finance...
c:\users\j_liu\enthoughcanopy\user\lib\site-packages\numpy\lib\polynomial.py:594: RankWarning: Polyfit may be poorly con
ditioned
  warnings.warn(msg, RankWarning)
Enter a date in format of YYYYMMDD to predict the stock price on that date or 'n' to move to next symbol:20170330
Predicated (linear regression) stock price for 'yhoo' based on 3m data
on 30, Mar 2017 isc:\users\j_liu\enthoughcanopy\user\lib\site-packages\numpy\lib\polynomial.py:594: RankWarning: Polyfit
 may be poorly conditioned
  warnings.warn(msg, RankWarning)
 40.5131835938
Enter a date in format of YYYYMMDD to predict the stock price on that date or 'n' to move to next symbol:n
>>Enter a valid stock symbol you would like to investigate or 'q' to quit the program: q

C:\project\stockPredictor>_
```

<u>Conclusion</u>

This project provides an intuitive user-friendly GUI interface to help user to retrieve historical stock price data with ease. It displays the historical stock price data in a scatter chart along with a linear-fit line using linear-fit regression and a polynomial-fit line using polynomial regression, and predicts stock price for a user-designated date.

In the main module *main.py*, class *StockPredictor()* is used to handle major operations for retrieving stock data, creating stock price chart and predicting stock price.

For users who are comfortable with command line, they can also perform the same tasks through Window's command prompt where they simply follow instructions on the command line to type in inputs.

Program Source Code

Refer to the separate files in the zip file. Files included:

- main.py
- plt.py
- runPgm.py
- form.html