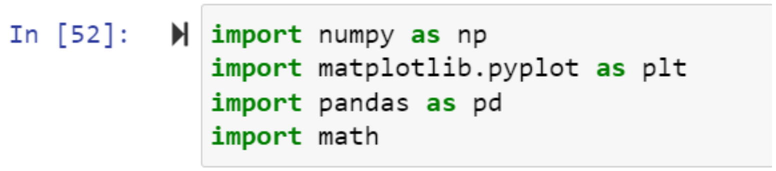
Abby Ortego
W0716476
CMPS 470 – 01
Program 1: Fun with Python

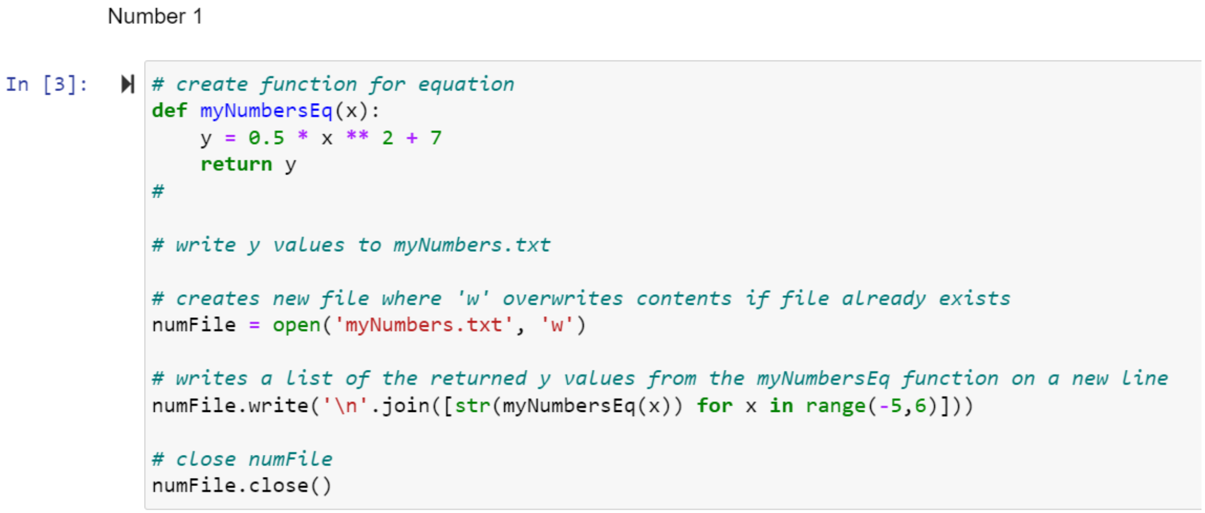
Imports



In [52]:

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import math
```

Number 1 Code

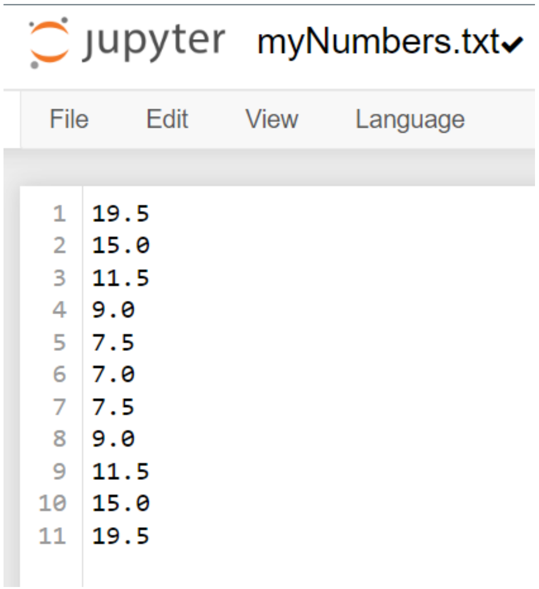


Number 1

In [3]:

```
# create function for equation
def myNumbersEq(x):
    y = 0.5 * x ** 2 + 7
    return y
#

# write y values to myNumbers.txt

# creates new file where 'w' overwrites contents if file already exists
numFile = open('myNumbers.txt', 'w')

# writes a list of the returned y values from the myNumbersEq function on a new line
numFile.write('\n'.join([str(myNumbersEq(x)) for x in range(-5,6)]))

# close numFile
numFile.close()
```

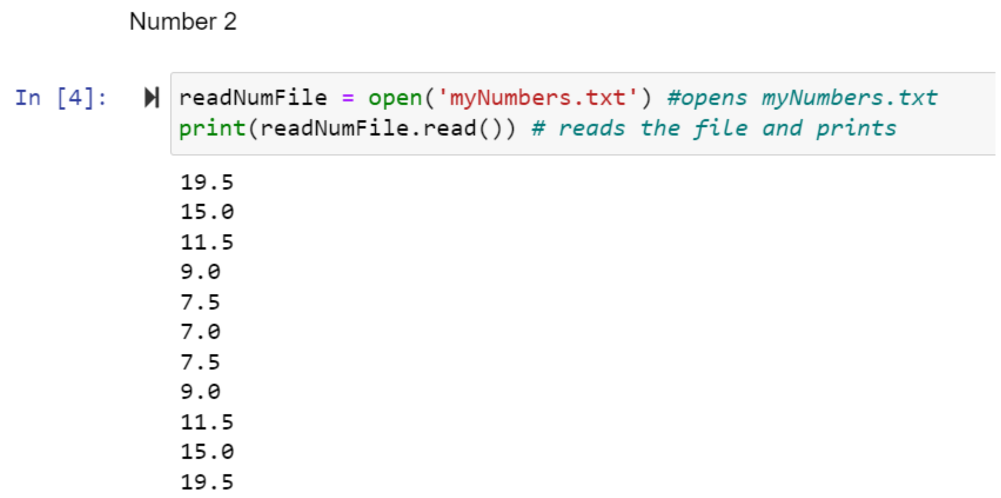
Number 1 Output to File



```
19.5
15.0
11.5
9.0
7.5
7.0
7.5
9.0
11.5
15.0
19.5
```

Abby Ortego
W0716476
CMPS 470 – 01
Program 1: Fun with Python

Number 2 Code & Output

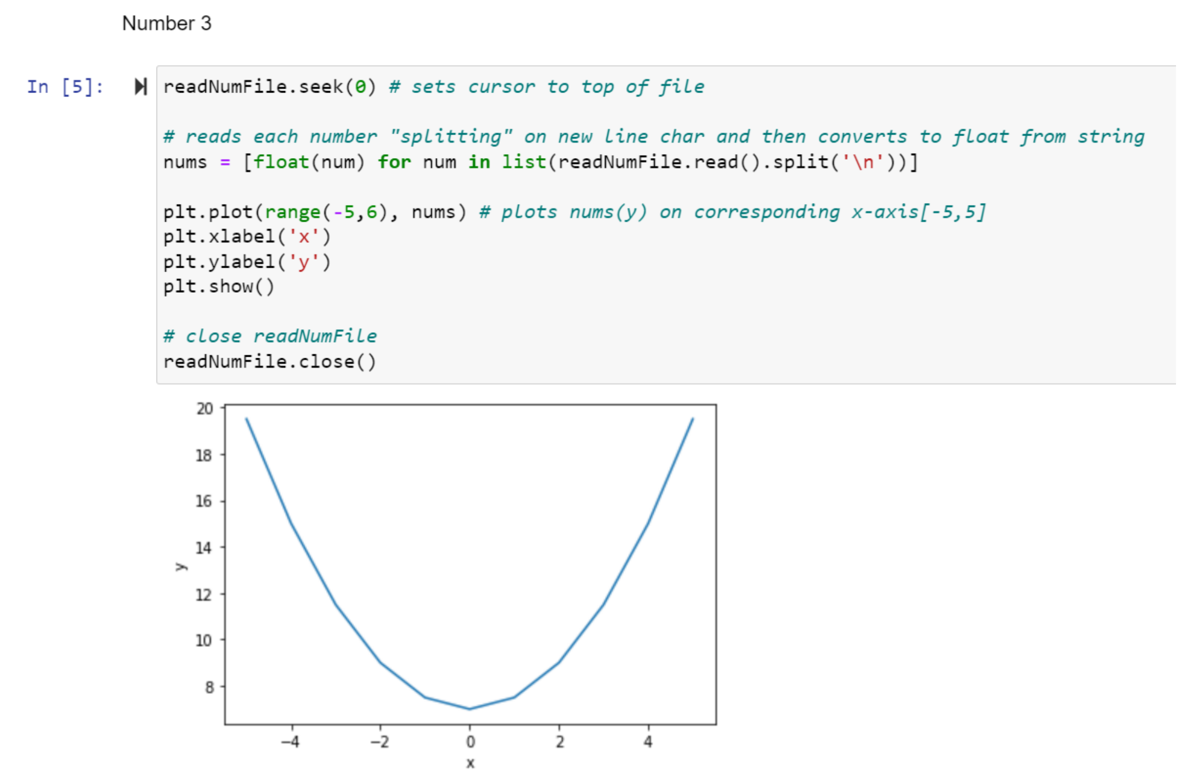


Number 2

```
In [4]:  readNumFile = open('myNumbers.txt') #opens myNumbers.txt
         print(readNumFile.read()) # reads the file and prints
```

```
19.5
15.0
11.5
9.0
7.5
7.0
7.5
9.0
11.5
15.0
19.5
```

Number 3 Code & Output

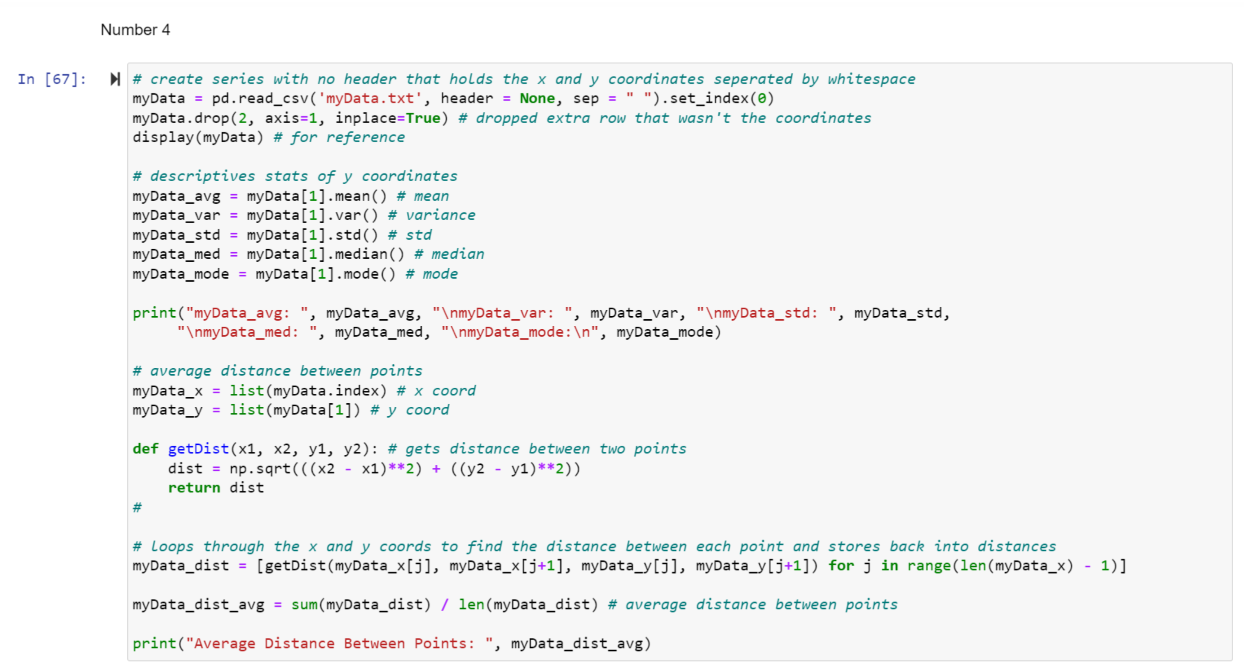


Number 3

```
In [5]:  readNumFile.seek(0) # sets cursor to top of file

         # reads each number "splitting" on new line char and then converts to float from string
         nums = [float(num) for num in list(readNumFile.read().split('\n'))]

         plt.plot(range(-5,6), nums) # plots nums(y) on corresponding x-axis[-5,5]
         plt.xlabel('x')
         plt.ylabel('y')
         plt.show()

         # close readNumFile
         readNumFile.close()
```

Abby Ortego
W0716476
CMPS 470 – 01
Program 1: Fun with Python

Number 4 Code

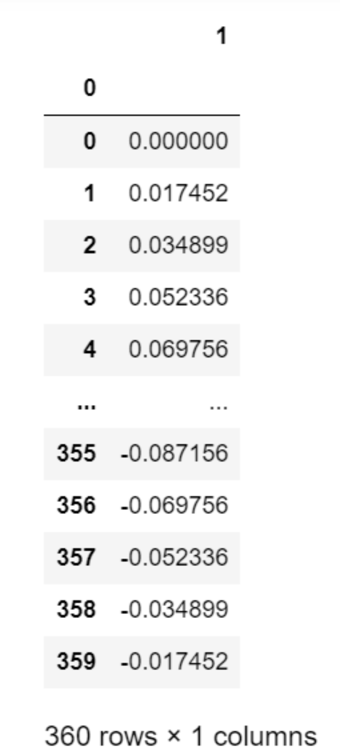


Number 4

In [67]:

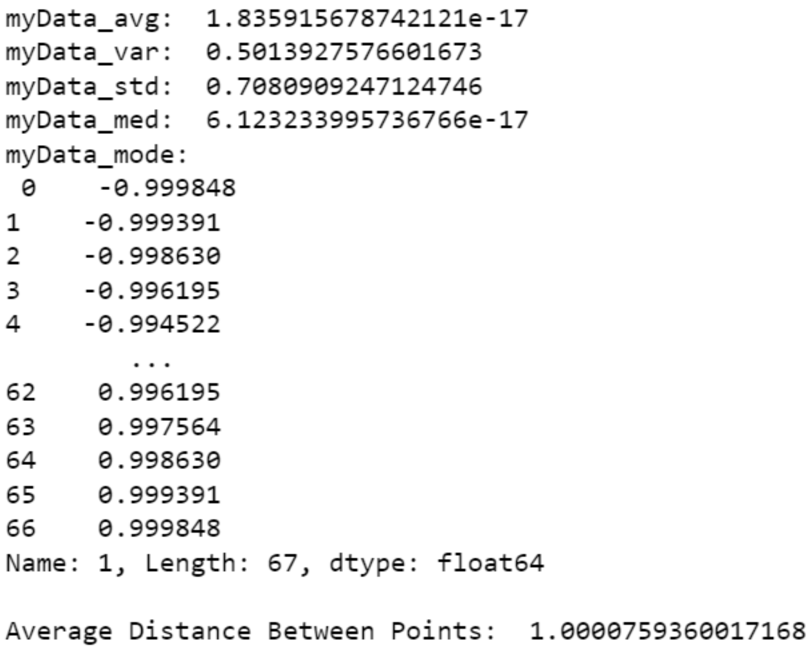
```
# create series with no header that holds the x and y coordinates seperated by whitespace
myData = pd.read_csv('myData.txt', header = None, sep = " ").set_index(0)
myData.drop(2, axis=1, inplace=True) # dropped extra row that wasn't the coordinates
display(myData) # for reference

# descriptives stats of y coordinates
myData_avg = myData[1].mean() # mean
myData_var = myData[1].var() # variance
myData_std = myData[1].std() # std
myData_med = myData[1].median() # median
myData_mode = myData[1].mode() # mode

print("myData_avg: ", myData_avg, "\nmyData_var: ", myData_var, "\nmyData_std: ", myData_std,
     "\nmyData_med: ", myData_med, "\nmyData_mode:\n", myData_mode)

# average distance between points
myData_x = list(myData.index) # x coord
myData_y = list(myData[1]) # y coord

def getDist(x1, x2, y1, y2): # gets distance between two points
    dist = np.sqrt(((x2 - x1)**2) + ((y2 - y1)**2))
    return dist
#

# loops through the x and y coords to find the distance between each point and stores back into distances
myData_dist = [getDist(myData_x[j], myData_x[j+1], myData_y[j], myData_y[j+1]) for j in range(len(myData_x) - 1)]

myData_dist_avg = sum(myData_dist) / len(myData_dist) # average distance between points

print("Average Distance Between Points: ", myData_dist_avg)
```

Number 4 Output



	1
0	
0	0.000000
1	0.017452
2	0.034899
3	0.052336
4	0.069756
...	...
355	-0.087156
356	-0.069756
357	-0.052336
358	-0.034899
359	-0.017452

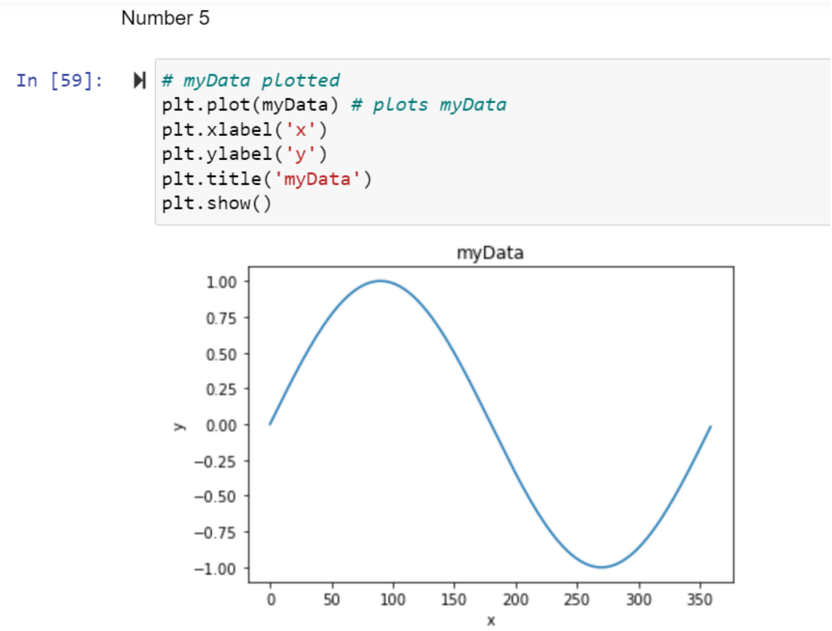
360 rows × 1 columns



```
myData_avg:  1.8359156787421210e-17
myData_var:  0.5013927576601673
myData_std:  0.7080909247124746
myData_med:  6.123233995736766e-17
myData_mode:
 0    -0.999848
1    -0.999391
2    -0.998630
3    -0.996195
4    -0.994522
        ...
62    0.996195
63    0.997564
64    0.998630
65    0.999391
66    0.999848
Name: 1, Length: 67, dtype: float64

Average Distance Between Points:  1.0000759360017168
```

Abby Ortego
W0716476
CMPS 470 – 01
Program 1: Fun with Python

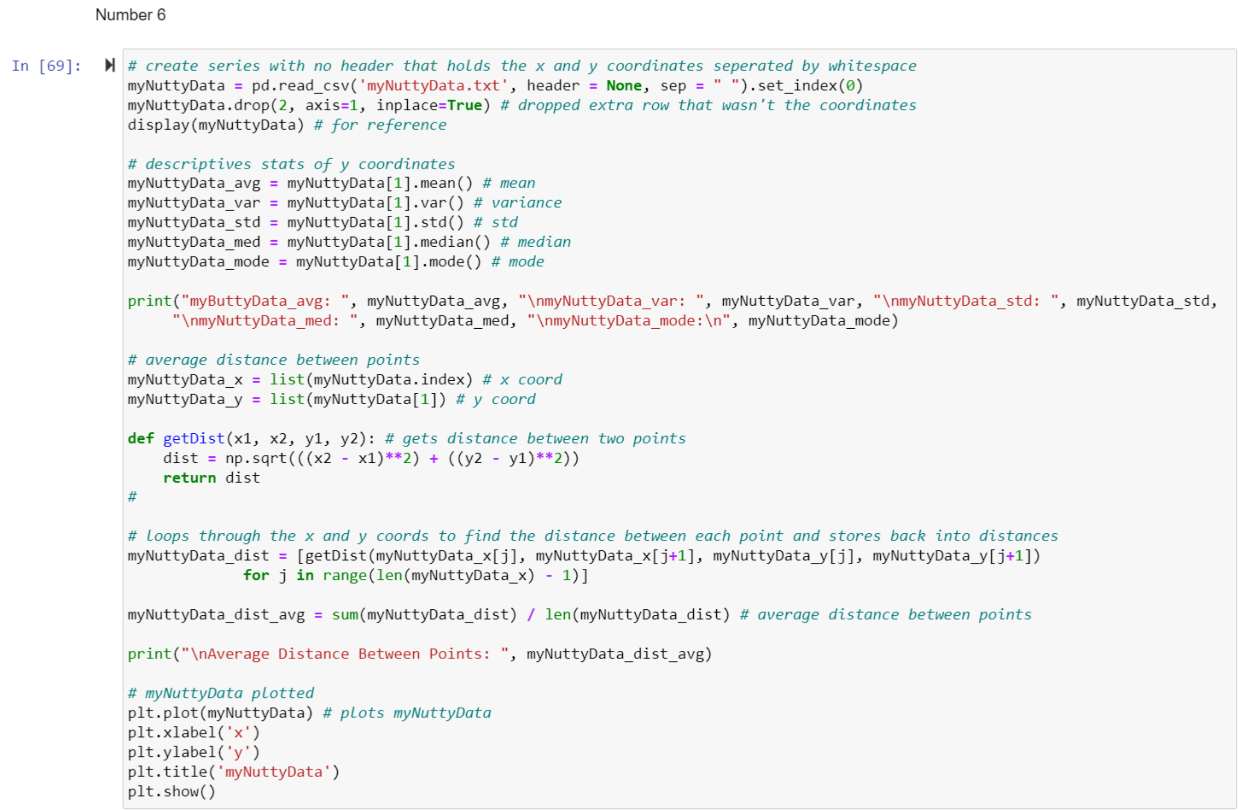
Number 5 Code & Plot



Number 5

```
In [59]:  # myData plotted
          plt.plot(myData) # plots myData
          plt.xlabel('x')
          plt.ylabel('y')
          plt.title('myData')
          plt.show()
```

Number 6 Code

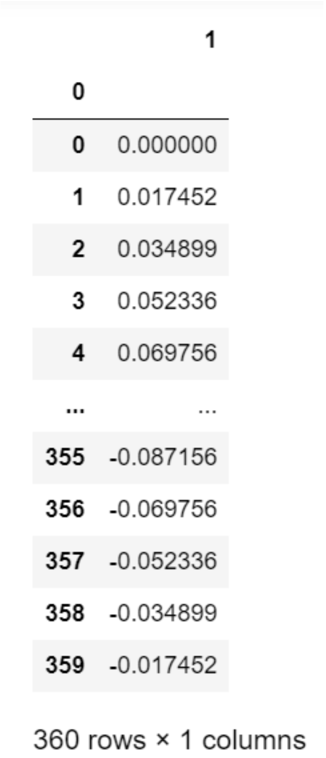


Number 6

```
In [69]:  # create series with no header that holds the x and y coordinates seperated by whitespace
          myNuttyData = pd.read_csv('myNuttyData.txt', header = None, sep = " ").set_index(0)
          myNuttyData.drop(2, axis=1, inplace=True) # dropped extra row that wasn't the coordinates
          display(myNuttyData) # for reference

          # descriptives stats of y coordinates
          myNuttyData_avg = myNuttyData[1].mean() # mean
          myNuttyData_var = myNuttyData[1].var() # variance
          myNuttyData_std = myNuttyData[1].std() # std
          myNuttyData_med = myNuttyData[1].median() # median
          myNuttyData_mode = myNuttyData[1].mode() # mode

          print("myButtyData_avg: ", myNuttyData_avg, "\nmyNuttyData_var: ", myNuttyData_var, "\nmyNuttyData_std: ", myNuttyData_std,
               "\nmyNuttyData_med: ", myNuttyData_med, "\nmyNuttyData_mode:\n", myNuttyData_mode)

          # average distance between points
          myNuttyData_x = list(myNuttyData.index) # x coord
          myNuttyData_y = list(myNuttyData[1]) # y coord

          def getDist(x1, x2, y1, y2): # gets distance between two points
              dist = np.sqrt(((x2 - x1)**2) + ((y2 - y1)**2))
              return dist
          #

          # loops through the x and y coords to find the distance between each point and stores back into distances
          myNuttyData_dist = [getDist(myNuttyData_x[j], myNuttyData_x[j+1], myNuttyData_y[j], myNuttyData_y[j+1])
                       for j in range(len(myNuttyData_x) - 1)]

          myNuttyData_dist_avg = sum(myNuttyData_dist) / len(myNuttyData_dist) # average distance between points

          print("\nAverage Distance Between Points: ", myNuttyData_dist_avg)

          # myNuttyData plotted
          plt.plot(myNuttyData) # plots myNuttyData
          plt.xlabel('x')
          plt.ylabel('y')
          plt.title('myNuttyData')
          plt.show()
```

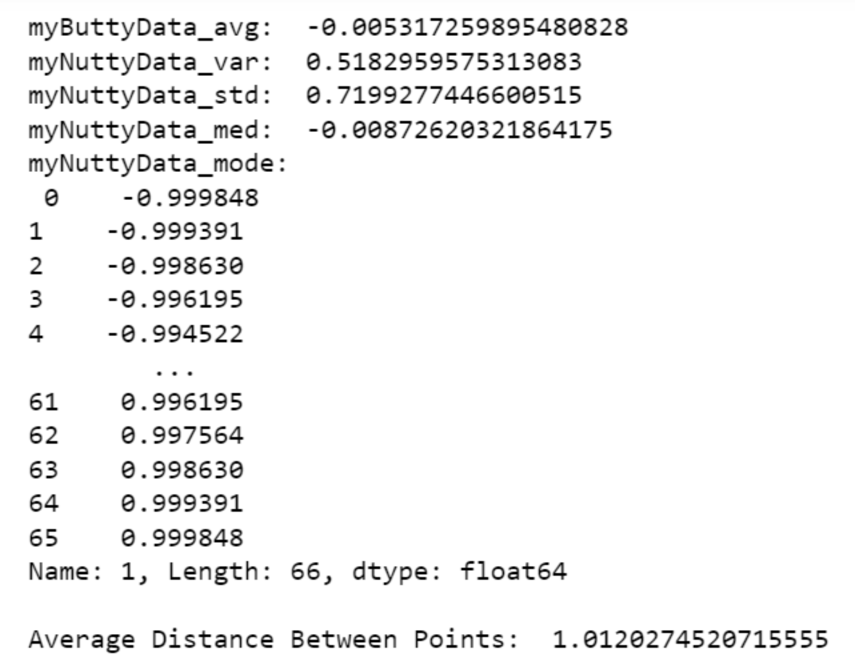
Abby Ortego
W0716476
CMPS 470 – 01
Program 1: Fun with Python

Number 6 Output

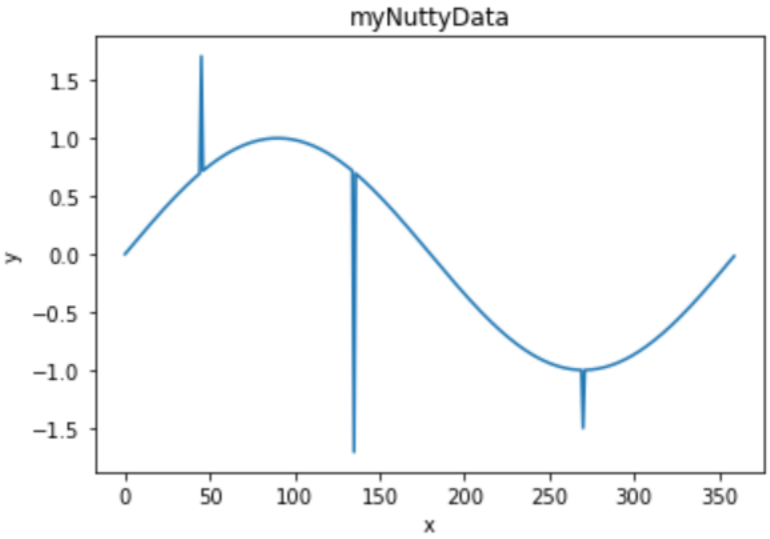


	1
0	
0	0.000000
1	0.017452
2	0.034899
3	0.052336
4	0.069756
...	...
355	-0.087156
356	-0.069756
357	-0.052336
358	-0.034899
359	-0.017452

360 rows × 1 columns



```
myButtyData_avg:  -0.005317259895480828
myNuttyData_var:  0.5182959575313083
myNuttyData_std:  0.7199277446600515
myNuttyData_med:  -0.00872620321864175
myNuttyData_mode:
 0    -0.999848
1    -0.999391
2    -0.998630
3    -0.996195
4    -0.994522
        ...
61    0.996195
62    0.997564
63    0.998630
64    0.999391
65    0.999848
Name: 1, Length: 66, dtype: float64

Average Distance Between Points:  1.0120274520715555
```


myNuttyData
y
x