Basics of Python

Max Huggins

February 9, 2021

1 Spyder the Helpful IDE

Spyder is an integrated development environment (IDE). It is useful because of the numerous packages available to users for data science. We will talk about one very useful package called Matplotlib in a moment.

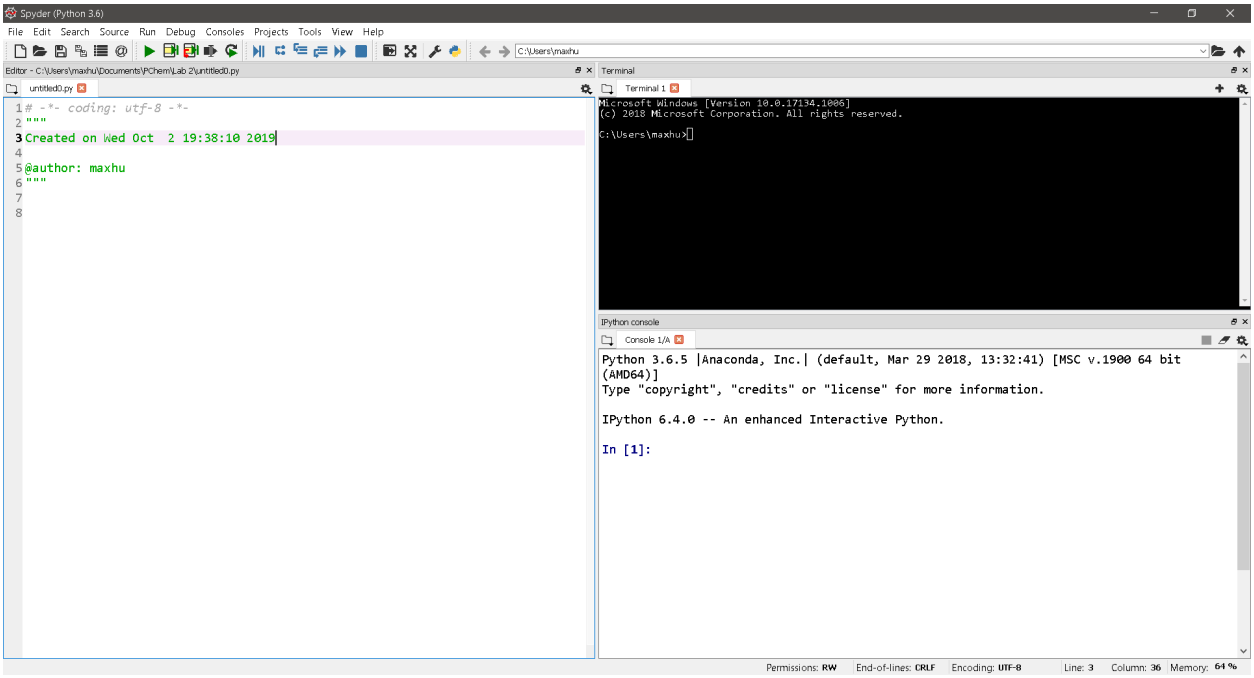

Figure 1: A typical Spyder screen.

2 Basic Math

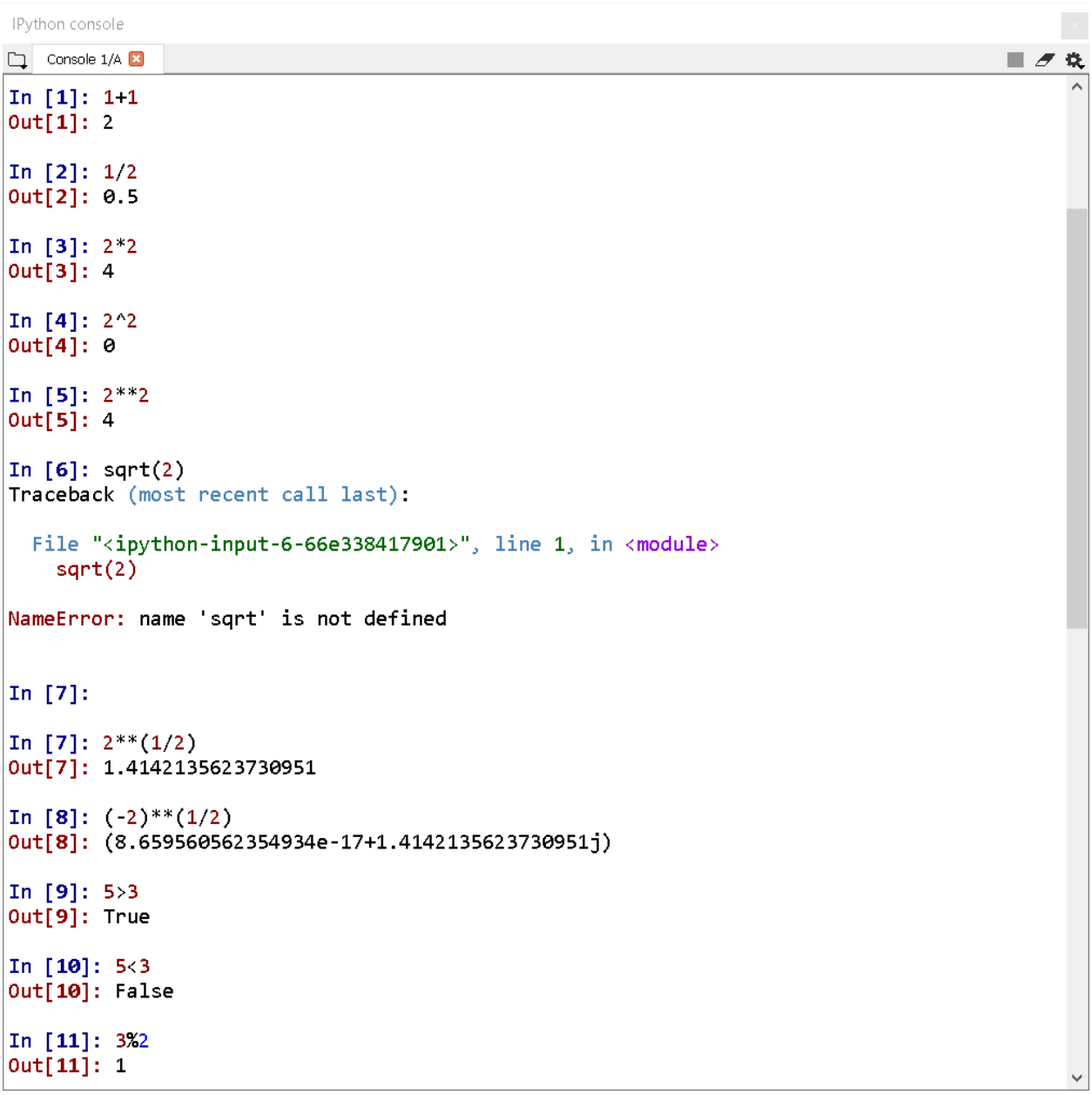

Figure 2: My console (kernel) when performing these operations.

3 Types

Python has several types: tall, skinny, well-fit, and so on. Just kidding, ha ha.
It does, however, have several data types it uses. Some of the most basic are listed here:
1) Integers
2) Strings
3) Floats
All of these data types serve different purposes and it is important to keep them differentiated.

4 Strings

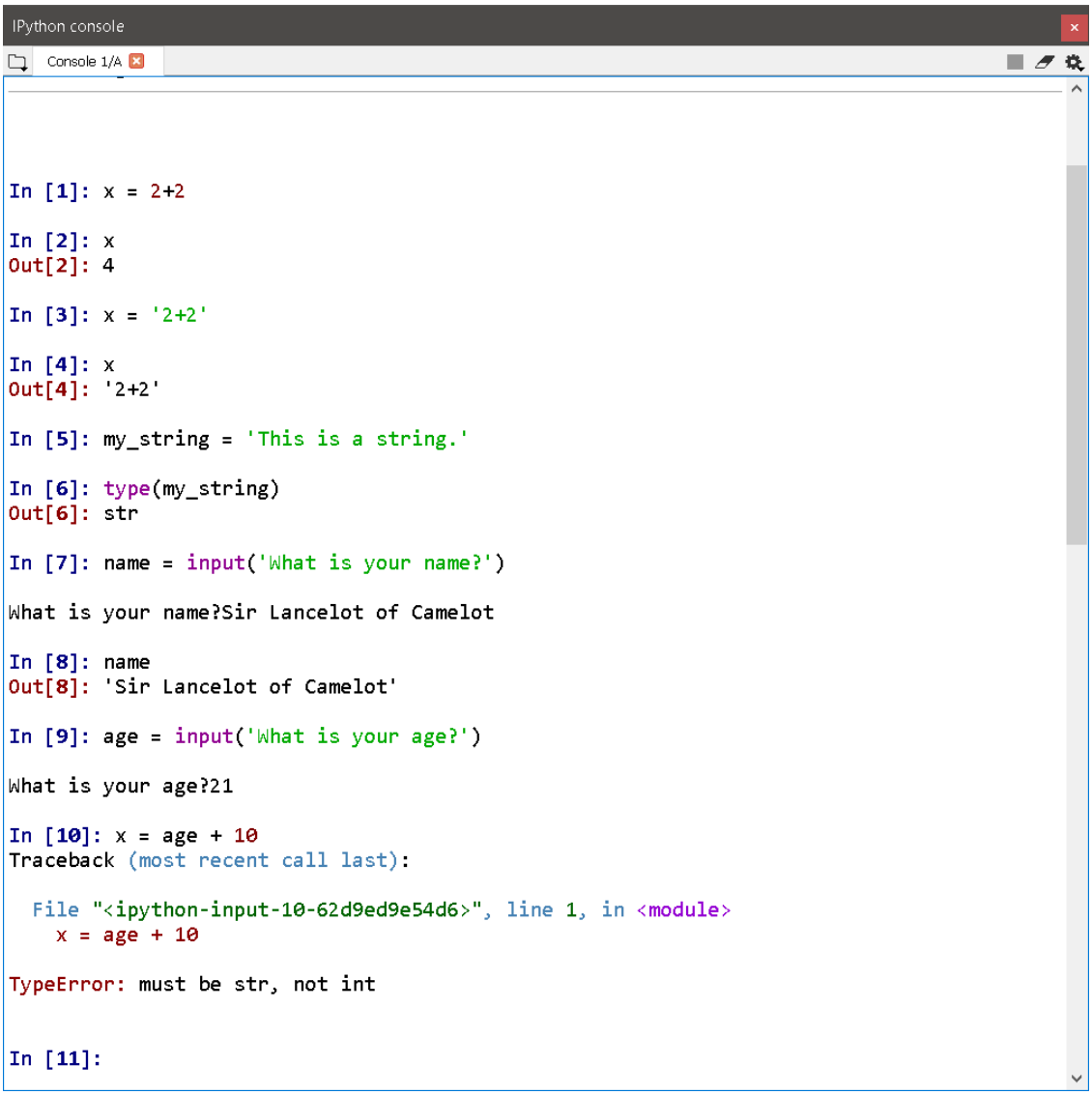

Figure 3: My console (kernel) when performing these operations.

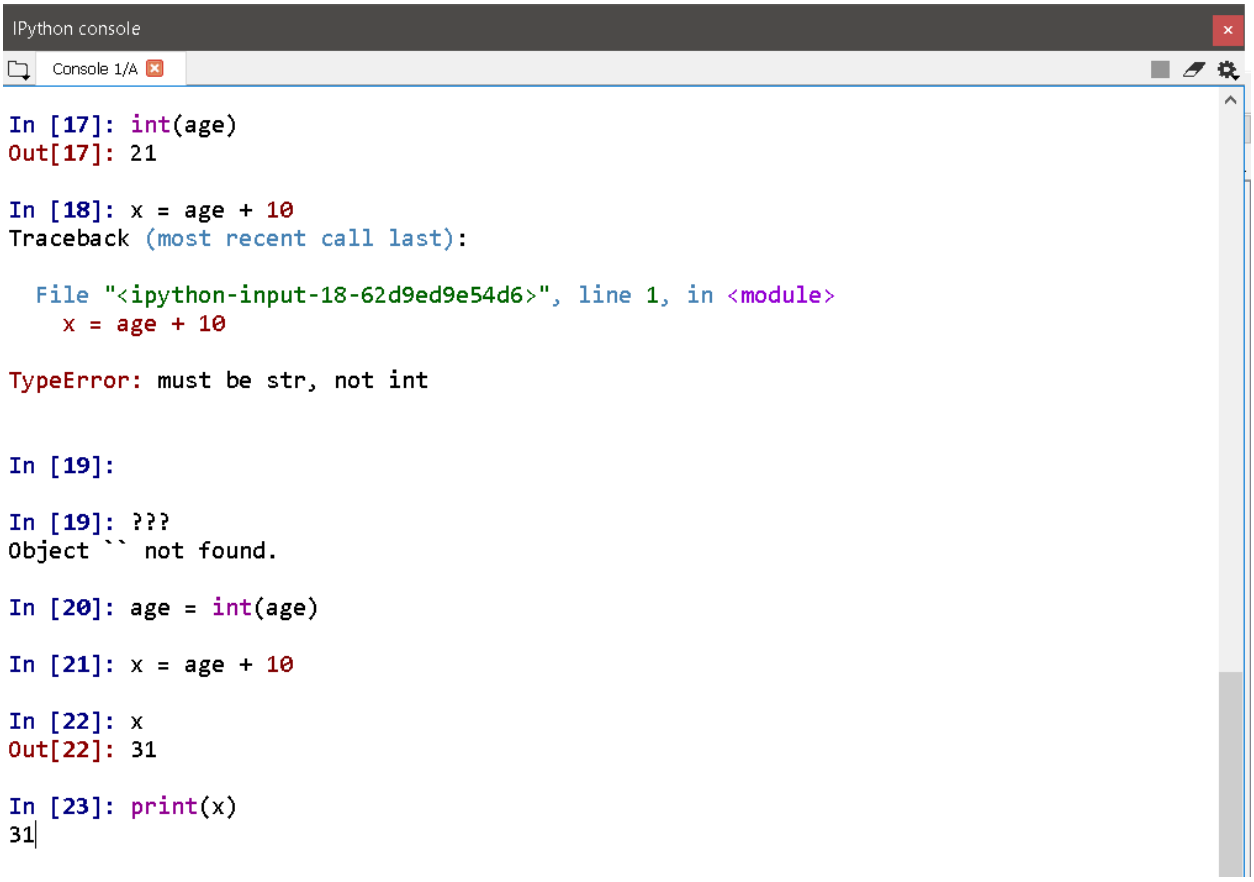

Figure 4: My console (kernel) when performing these operations.

5 Lists

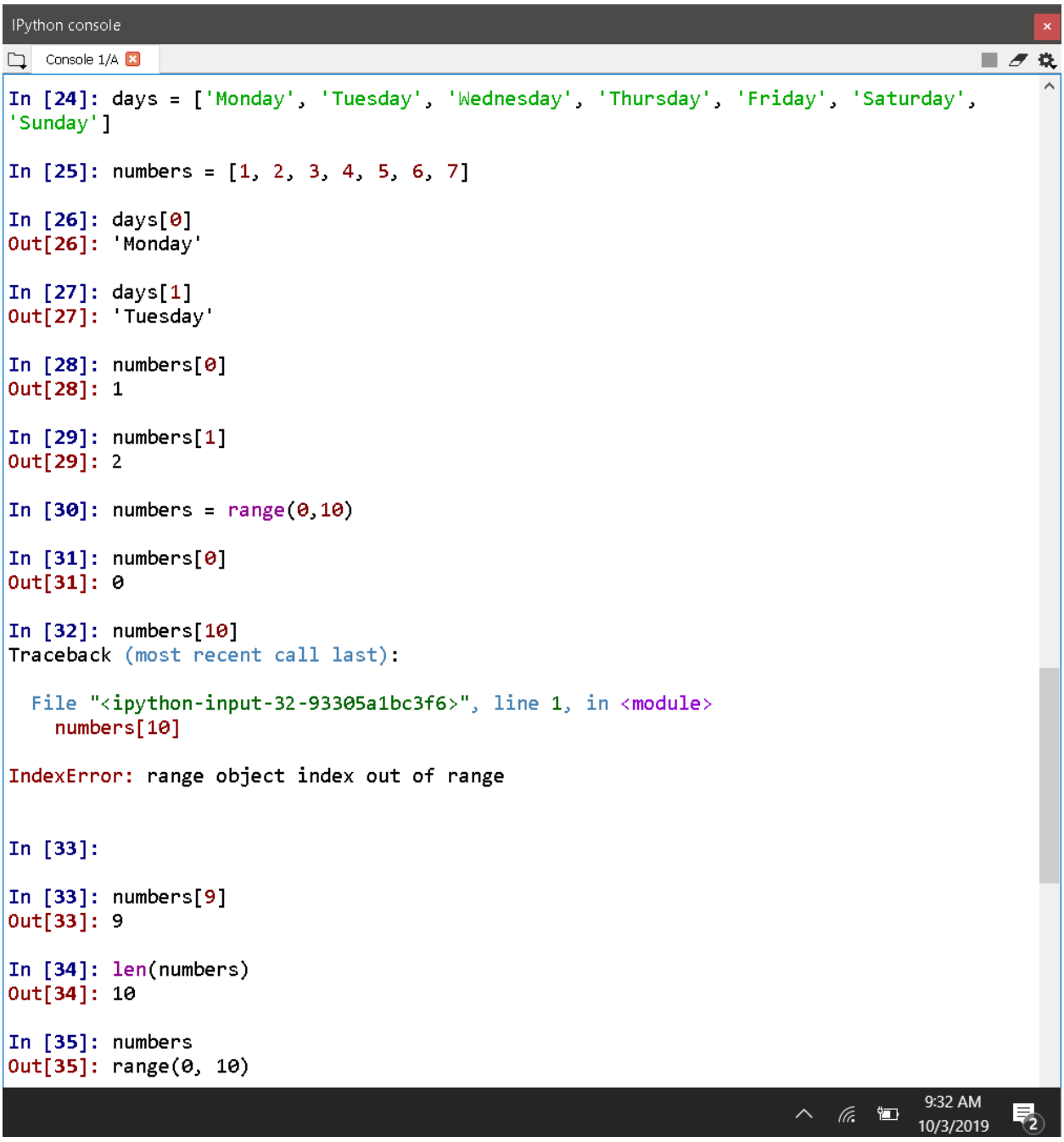

Figure 5: My console (kernel) when performing these operations.

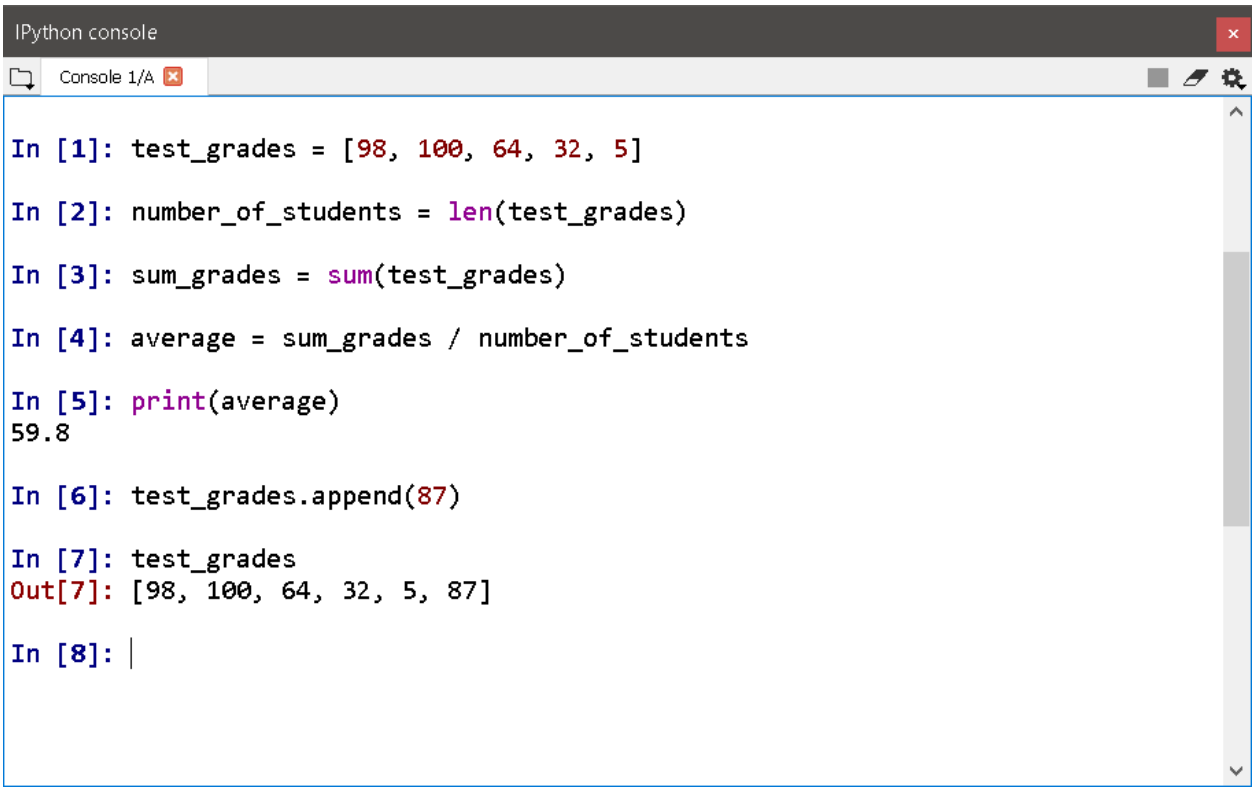

Figure 6: My console (kernel) when performing these operations.

6 Conditional Statements and Loops

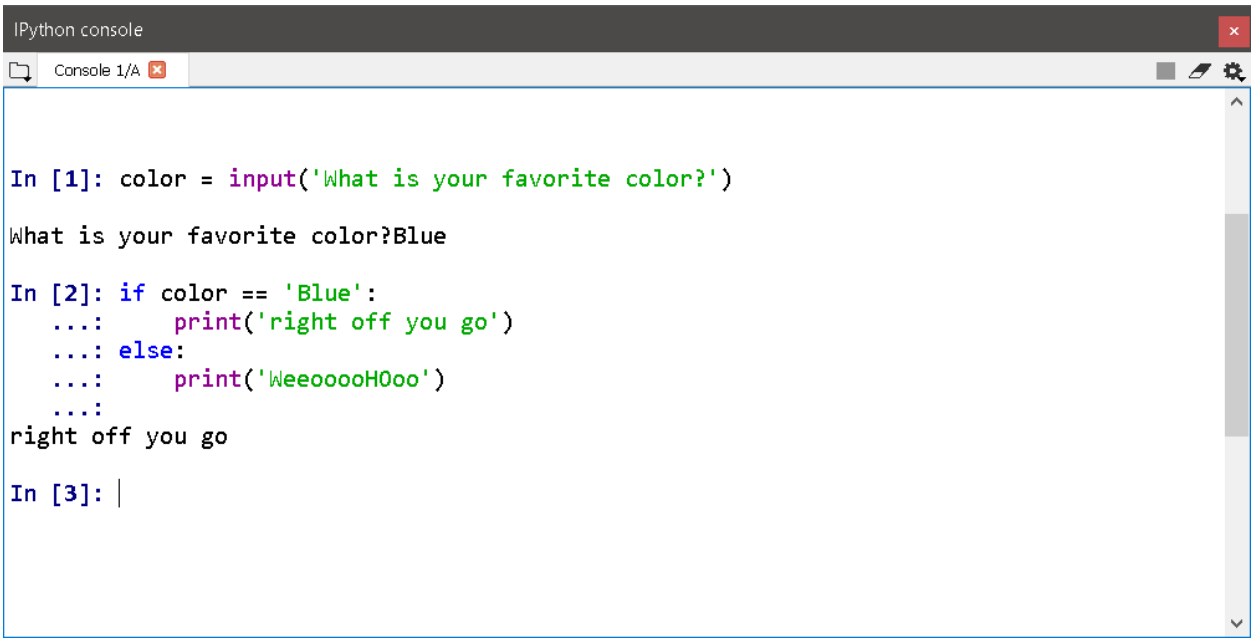

Figure 7: My console (kernel) when performing these operations.

This is our first Python script.

```
# -*- coding: utf-8 -*-
"""
Created on Wed Oct  2 19:38:10 2019

@author: maxhu
"""

print('Hello World!')
```

Let's explore some loops:

```
# -*- coding: utf-8 -*-
"""
Created on Wed Oct  2 19:38:10 2019

@author: maxhu
"""

#This is how you can comment
#Comments are used to explain your code.


#This is an example of a for loop
for n in range(0,11):
    print('The number is: ', n)

```

```
16 print('This was an example of a for loop.')
17
18 #This is an example of a while loop.
19 n = 0
20 while n < 11:
21     print('The number is: ', n)
22     n = n + 1
23 print('This was an example of a while loop.')
24
25 #This is another example
26 n = 0
27 while n < 11:
28     if n % 2 == 0: #if n is even
29         print('The number is: ', n)
30
31     n = n + 1
```

7 Matplotlib

```
1
2 #This is how to import a package into your script
3 import matplotlib.pyplot as plt #The as plt part just lets you call it in a
4                                  #shorter way.
5
6 #Let's make some random data for our x and y axes
7 x_data = range(0,10)
8 y_data = range (35,45)
9
10 #Now, we can plot it with matplotlib
11 plt.plot(x_data, y_data) #This is the as plt part
12
13 #These are our axes label and title
14 plt.ylabel('y Data')
15 plt.xlabel('x Data')
16 plt.title('Data')
```

Let's do a scatter plot instead for discrete data.

```
1
2 #This is how to import a package into your script
3 import matplotlib.pyplot as plt #The as plt part just lets you call it in a
4                                  #shorter way.
5
6 #Let's make some random data for our x and y axes
7 x_data = range(0,10)
8 y_data = range (35,45)
9
10 #Now, we can plot it with matplotlib
11 plt.scatter(x_data, y_data) #This is the as plt part
12
13 #These are our axes label and title
14 plt.ylabel('y Data')
15 plt.xlabel('x Data')
16 plt.title('Data')
```

8 References

A good majority of this I took from Dr. Slaton's Activity 0.2: Programming which was the first activity we did in μcontrollers. So if you end up taking that course then you'll probably run into a lot of this again.