Post-Lab Week 1 Submission

Lab 1A

1. **Hello World, name, and Arduino status:** Screenshot the cells and outputs of your Python notebook where you printed “Hello World!”, your name, and whether you have your Arduino. Insert this screenshot below.

```
print("Hello World, my name is Sergio Gonzalez. I have an arduino.")
```

```
Hello World, my name is Sergio Gonzalez. I have an arduino.
```

2. **Kinetic Energy Calculations:** Screenshot the cells and outputs of your Python notebook where you calculated the kinetic energy of a 5 kg object moving at 10 m/s, made a comment, and calculated the natural log of this kinetic energy. Insert this screenshot below.

```
import numpy as np
m = 5
v = 10
E = (1/2)*m*v**2
#Step 1: Set variable m to be the mass of the object in kg
#Step 2: Set variable v to be the velocity of the object in m/s
#Step 3: Write the Python equivalent of k=1/2mv^2
print(E)
print(np.log(E))
```

```
250.0
5.521460917862246
```

3. **Resistance measurements:** Complete the following table with your resistance multimeter measurements you took in lab 1A.

Stated Resistance (Ω)	Measured Resistance (Ω)
220Ω	218Ω

1000**Ω**	992**Ω**
10000**Ω**	10000**Ω**

4. **Voltage measurements:** Complete the following table with your voltage multimeter measurements you took in lab 1A.

Measurement	Measured Voltage (V)
5V pin	4.88V
3.3V pin	3.25V

Lab 1B

1. **Problem 1:** Screenshot the cells and outputs of your Python notebook for problem 1. Insert this screenshot below.

```
import numpy as np
positions = np.array([[2., 4.],
              [1., 3.],
              [7., 2.],
              [1., 8.],
              [9., 2.]])
# Separate out the x coordinates here
xCoor = positions[:,0]
print("X-Coordinates:")
print(xCoor)
# Separate out the y coordinates here
yCoor = positions[:,1]
print("Y-Coordinates:")
print(yCoor)
# Calculate the distances from the origin
distances = [];
for i in range(len(xCoor)):
    xTemp = xCoor[i]
    yTemp = yCoor[i]
    dTemp = float(np.sqrt(yTemp**2 + xTemp**2))
    distances.append(dTemp)
# Print the distances
print("Distances:")
print(distances)
```

```
X-Coordinates:
[2. 1. 7. 1. 9.]
Y-Coordinates:
[4. 3. 2. 8. 2.]
Distances:
[4.47213595499958, 3.1622776601683795, 7.280109889280518, 8.06225774829855, 9.219544457292887]
```

2. **Problem 2 (setup):** Screenshot the cells and outputs of your Python notebook for problem 2 where you mounted your drive, read in the practice linear data, and extracted the data for the x and y axes. Insert this screenshot below.

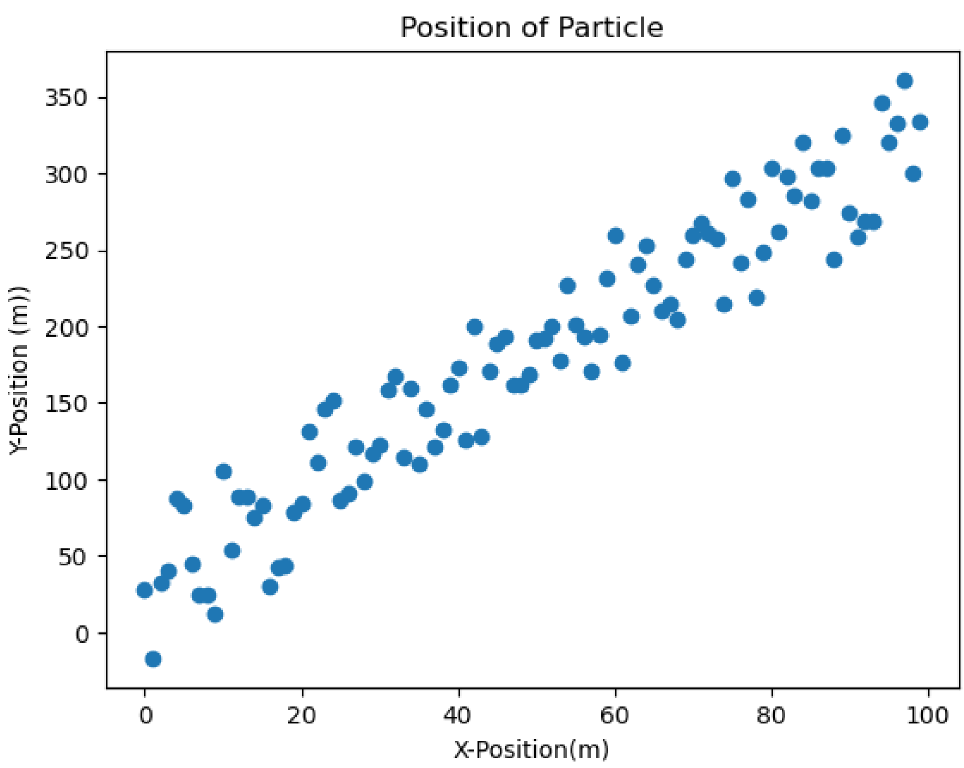
```python
# read_in_array = np.loadtxt('file_path')
import numpy as np
read_in_array = np.loadtxt('practicelineardata.txt')
```

```python
#Header Lines
import numpy as np
read_in_array = np.loadtxt('practicelineardata.txt')
# The first column is the x axis, second is y axis
# Take all of the elements in 0th column
xAxis = read_in_array[:,0]

# Take all of the elements in the 1st column

yAxis = read_in_array[:,1]
```

3. **Problem 2 (scatterplot):** Screenshot the cells and outputs of your Python notebook for problem 2 where you plotted the practice linear data in a scatter plot, labeled the axes, and gave the plot a title. Insert this screenshot below.

```
import matplotlib.pyplot as plt
read_in_array = np.loadtxt('practicelineardata.txt')
# The first column is the x axis, second is y axis
# Take all of the elements in 0th column
xAxis = read_in_array[:,0]

# Take all of the elements in the 1st column

yData = read_in_array[:,1]
# Make the scatter plot and label the data
plt.scatter(xAxis, yData)

# Label the x axis
plt.xlabel("X-Position(m)")

# Label the y axis
plt.ylabel("Y-Position (m))")

# Give the plots a title
plt.title("Position of Particle")
```

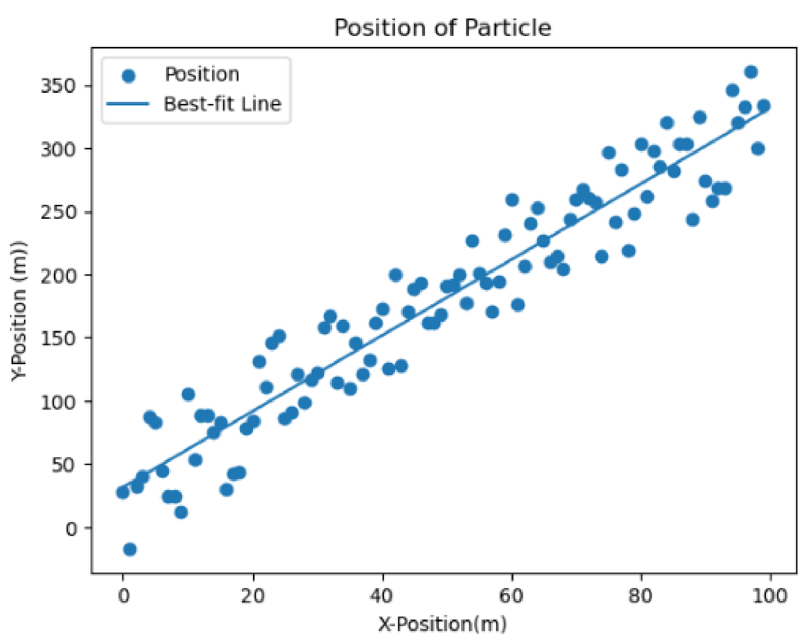
```
Text(0.5, 1.0, 'Position of Particle')
```



4. **Problem 2 (best-fit line):** Screenshot the cells and outputs of your Python notebook for problem 2 where you generated a best-fit line for the practice linear data, plotted this best-fit line with the scatterplot of the data, and created labels to describe the elements of the plot. Insert this screenshot below.

```
# Copy all of your code from the previous code cell and place it here
import matplotlib.pyplot as plt
import numpy as np
read_in_array = np.loadtxt('practicelineardata.txt')
# The first column is the x axis, second is y axis
# Take all of the elements in 0th column
xAxis = read_in_array[:,0]

# Take all of the elements in the 1st column

yData = read_in_array[:,1]
# Make the scatter plot and label the data
plt.scatter(xAxis, yData, label = "Position")

# Label the x axis
plt.xlabel("X-Position(m)")

# Label the y axis
plt.ylabel("Y-Position (m))")

# Give the plots a title
plt.title("Position of Particle")


# # np.polyfit() requires at least 3 arguments or inputs
coeff_linear = np.polyfit(xAxis, yData, 1)
# The x-axis data, the y-axis data and the degree of the polynomial that we are trying to fit - save
# The zeroth element is the slope and first element is the intercept
slope = coeff_linear[0]
interscept =coeff_linear[1]

# plt.plot() generates a connected line - plot the best fit line and label this line
plt.plot(xAxis, slope*xAxis + interscept, label = "Best-fit Line")
# Remember to use plt.legend() to generate plot labels
plt.legend()
```

```
<matplotlib.legend.Legend at 0x780692ac7a40>
```



5. **Problem 3:** Screenshot the cells and outputs of your Python notebook for problem 3. Insert this screenshot below.

```python
#General Set Up
import numpy as np
import matplotlib.pyplot as plt

# x-array
# x-array is time
time = np.linspace(0, 5, 100)
# y-array
Height_0 = 500
Height = []
for i in range(len(time)):
    Height_0 -= (0.5)*(9.8)*(time[i])**2
    if Height_0 < 0:
       Height_0 = 0
    Height.append(Height_0)
# Make the continuous plot
plt.plot(time,Height,label="Height")
plt.legend()
# Label the x axis
plt.xlabel ("Time (s)")
# Label the y axis
plt.ylabel ("Height (m)")
# Give the plot a title
plt.title("Height versus time")
```

```
Text(0.5, 1.0, 'Height versus time')
```