**Tasks and Analysis for Gay Lussac’s Law Experiment**

Before or after the experiment you will be required to add or remove some pieces of python code to a python script (gaylussac1.py) that **{\displaystyle {P}\propto {T},}**uses the temperature and pressure sensors to record the changing temperature and pressure in the jar. The tasks are as follows:

1. **Conditionals (IF else statements)**

Inside the loop, add a block of code that checks the temperature value:

If the temperature is greater than or equal to 28; set all the LED lights to red

If the temperature is less than 28; set all the LED lights to green

**Hint:** To set all the LEDs to a specific color, use **sense.set\_pixels(pixel\_list)** depending on the color of choice. Such as green\_list or red\_list.

1. **For loop**

Change the while loop above to a For loop that will loop only 3 times instead of endlessly.

1. **While loop**

Change the for loop in the program to a while loop and make it loop only 5 times.

**Hint**: use a counter such as (**i=0**)

1. **Functions**

Write a function in the beginning of the script called pressure\_monitor(prezzure) which will take the pressure value as a parameter. If the pressure value is greater than 1000 milibars, the sense hat shows a message on the LED screen.

**Hint:** To show message on the LED screen, use **sense.show\_message(“ ”)**

1. **Graph your recorded temperature and pressure values**

Create a new python script to graph you recorded values

Here’s a simplified example of how matplotlib could be used to graph:

**#imports the necessary modules to run the script**

**Import matplotlib**

**matplotlib.use("TKAgg")**

**Import matplotlib.pyplot as plt**

**# initiliazing the data values to be plotted**

**time\_values = [0.12,0.32,0.47,0.60,0.71]**

**accel\_values = [4,7,3,4,5]**

**# use the plot function to plot accel values against time values**

**plt.plot (time\_values, accel\_values) # x,y axis respectively**

**#labels the axis respectively**

**plt.xlabel('time (s)')**

**plt.ylabel('acceleration (Gs)')**

**plt.title('About as simple as it gets') #gives the title of the graph**

**plt.show() # shows the plotted graph n screen**