Using Python for Modelling & Simulation

https://trinket.io/python or https://repl.it/

CLNandi (Dr)

What is Modelling?

Definition of Modelling - to make a good symbolic representation of any actual system .

Examples:-

(1) A Globe on your office desk is a Model of the Earth

(2) Maps about the Weather

(3) Looking at patterns of disease - eg. COVID-19

Models are used in a number of ways:-

(1) To gain insights about the data/about a situation that would been otherwise difficult to see.

(2) To help make predictions/forecasts about the future.

https://trinket.io/python or https://repl.it/

An Example of Modelling in Real-Life - COVID-19

Recently, at Imperial College, Professors and Researchers used Computer Programs were involved in the modelling of COVID-19 forecasts.

Their model is the one being followed now in the UK.

And they heavy use of the Programming languages Python & R.

They have published some scientific papers about their research.

https://www.imperial.ac.uk/media/imperial-college/medicine/mrc-gida/2020-03-30-COVID19-Report-13.pdf

Note, that they make heavy use of graphs in trying to illustrate their ideas.

The researchers are also in the midst of publishing their programming code.

This will be found here:-

https://github.com/ImperialCollegeLondon/covid19model

Note, the folder the researchers have created a folder called 'Python' to put their Python code in it.

This is called Open-Source Code - Code openly available to the general public and us.

Eventually the public will have enough information to recreate the Imperial Model about Covid-19..

https://trinket.io/python or https://repl.it/

Objectives

Today, We will look at the Building Blocks of Models

And in doing so:-

We will study & write Python Programs modelling various real-life situations.

We will produce some graphs.

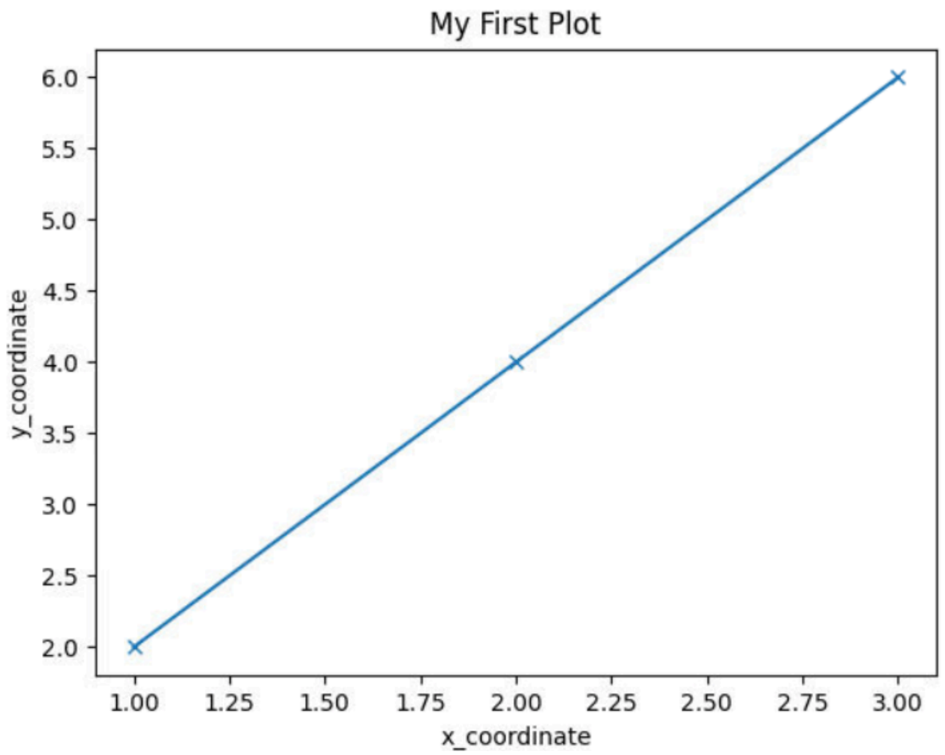
https://trinket.io/python or https://repl.it/

CLNandi (Dr)

Example 1

```
# Example 1 - A Straight Line Graph with Annotation & Marking Points
import matplotlib.pyplot as plt
def create_graph():
    x_numbers = [1, 2, 3]  #Numbers on X axis
    y_numbers = [2, 4, 6]   #Numbers on Y axis

    plt.plot(x_numbers, y_numbers, marker=‘x') #Instruction to Plot

    plt.xlabel(‘x_coordinate')    # Labelling X-axis
    plt.ylabel(‘y_coordinate’)     # Labelling Y-axis

    plt.title("My First Plot”) #Title of Plot
    plt.show().  # Draw the Graph

# Calling the Above Function
create_graph().
```

Output

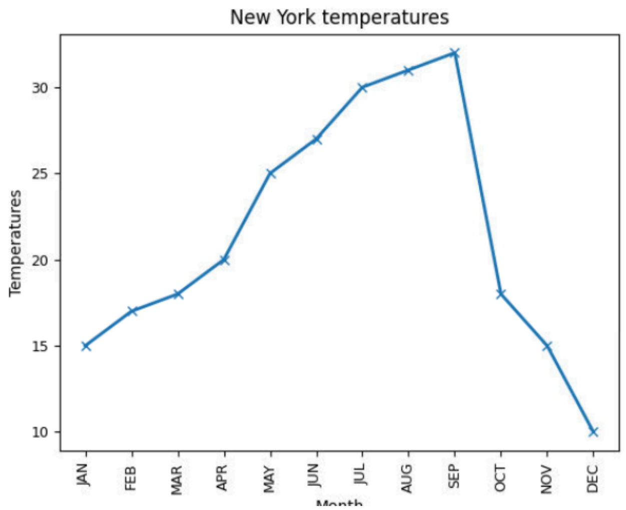


https://trinket.io/python or https://repl.it/

Example 2

```
# Example 2 - Plotting New York Temperatures
import matplotlib.pyplot as plt
def new_graph():

    New_York_Temp = [15, 17, 18,20,25,27,30,31,32,18,10,9] #Temperature data
    Month = [‘JAN','FEB','MAR','APR','MAY','JUN','JUL','AUG','SEP','OCT','NOV','DEC']

    plt.plot(Month, New_York_Temp, marker = ‘x’)  #Instruction to Plot

    plt.xlabel =(“Month”)     #Labelling X-axis
    plt.ylabel(“Temperatures”) #Labelling Y-axis

    plt.title("New York temperatures”) #Displaying Title

    plt.tick_params(axis=“both”,labelsize=14) #Label Sizes 14 Points
    plt.xticks(Month, rotation = “vertical”) #Writing the labels on the x-axis vertically
    plt.show().   #Draw the Graph

new_graph()
```

Output



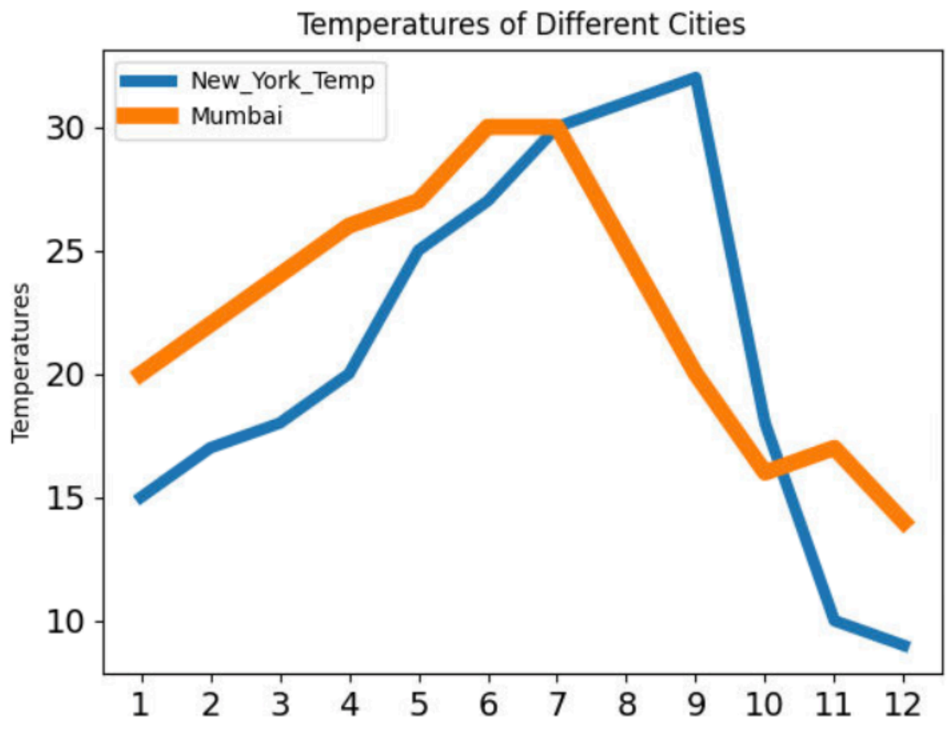
https://trinket.io/python or https://repl.it/

CLNandi (Dr)

Example 3

```
# Example 3 - Comparing Monthly Temperatures, Having more than 1 graph
import matplotlib.pyplot as plt
def Graph_3():
  New_York_Temp = [15, 17, 18,20,25,27,30,31,32,18,10,9]#Temperature data


  Mumbai_Temp = [20,22,24,26,27,30,30,25,20,16,17,14]   #Temperature data


  Month = [1,2,3,4,5,6,7,8,9,10,11,12] #Numbers on X axis


  plt.plot(Month, New_York_Temp, linewidth=5, label="New_York_Temp")
  plt.plot(Month,Mumbai_Temp,linewidth=7, label="Mumbai")



  plt.xlabel =(“Month”)
  plt.ylabel("Temperatures")
  plt.title("Temperatures of Different Cities")
  plt.tick_params(axis="both",labelsize=14)
  plt.xticks(Month)

  plt.legend()
  plt.show()    #Draw Graph

Graph_3()
```

Output



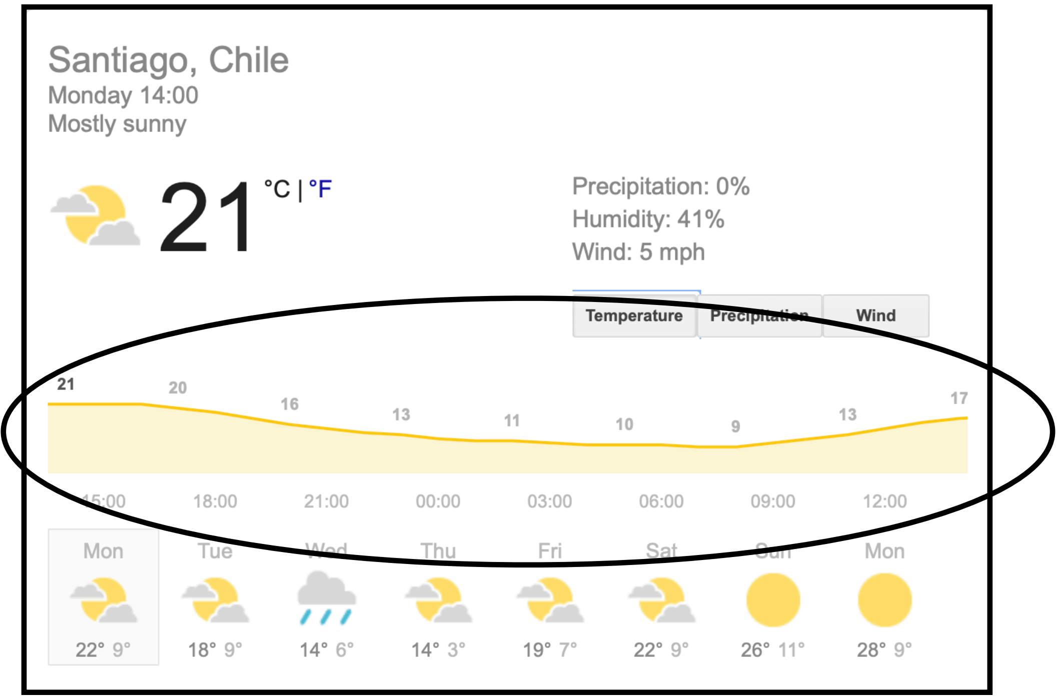
https://trinket.io/python or https://repl.it/

Exercises (1) & (2) - https://trinket.io/python or https://repl.it/

(1) Use a City of your choice and generate a graph with the time of day on the x-axis and the corresponding temperature on the y-axis.

So, for example, let us say you choose the city "Santiago".

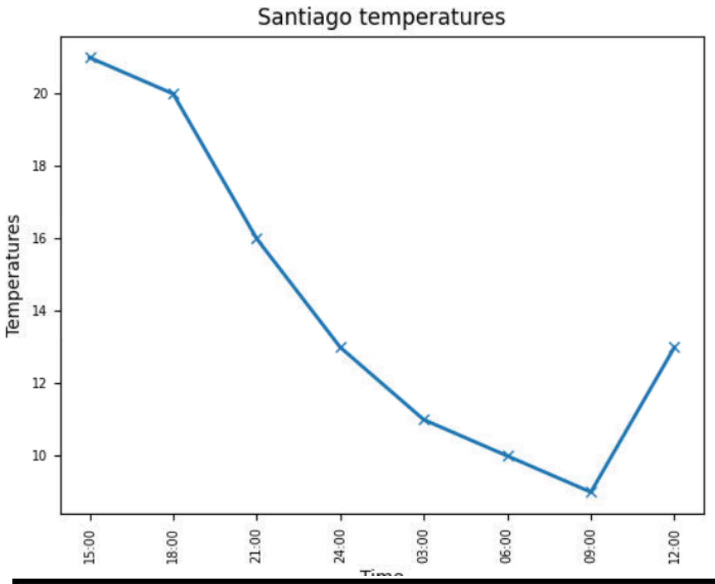
Google "Santiago weather" & you get something like this:-


Santiago, Chile
Monday 14:00
Mostly sunny

21 °C | °F

Precipitation: 0%
Humidity: 41%
Wind: 5 mph

The Output looks something like this:-



(2) Then generate a graph with information about 2 cities

Hints/Help

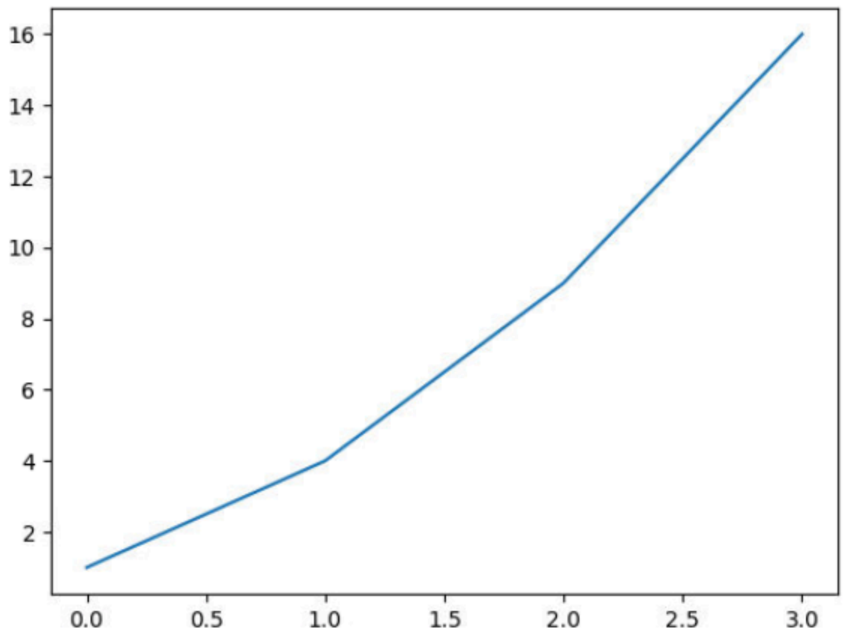
```
# Example 2 - Plotting New York Temperatures
import matplotlib.pyplot as plt
def new_graph():

    New_York_Temp = [15, 17, 18,20,25,27,30,31,32,18,10,9] #Temperature data
    Month = [‘JAN','FEB','MAR','APR','MAY','JUN','JUL','AUG','SEP','OCT','NOV','DEC']

    plt.plot(Month,New_York_Temp, marker = ‘x’)  #Instruction to Plot

    plt.xlabel =(“Month”)     #Labelling X-axis
    plt.ylabel(“Temperatures”) #Labelling Y-axis

    plt.title("New York temperatures”) #Displaying Title

    plt.tick_params(axis=“both”,labelsize=14) #Label Sizes 14 Points
    plt.xticks(Month, rotation = “vertical”) #Writing the labels on the x-axis vertically
    plt.show().   #Draw the Graph

new_graph()
```

```
# Example 3 - Comparing Monthly Temperatures, Having more than 1 graph
import matplotlib.pyplot as plt
def Graph_3():
  New_York_Temp = [15, 17, 18,20,25,27,30,31,32,18,10,9]
  Mumbai_Temp = [20,22,24,26,27,30,30,25,20,16,17,14]

  Month = [1,2,3,4,5,6,7,8,9,10,11,12] #Numbers on X axis

  plt.plot(Month, New_York_Temp, linewidth=5, label="New_York_Temp")
  plt.plot(Month,Mumbai_Temp,linewidth=7, label="Mumbai")

  plt.xlabel =(“Month”)
  plt.ylabel("Temperatures")
  plt.title("Temperatures of Different Cities")
  plt.tick_params(axis="both",labelsize=14)
  plt.xticks(Month)

  plt.legend()
  plt.show()    #Draw Graph

Graph_3()
```

Example 4

```
# Plotting Square Numbers
import matplotlib.pyplot as plt

def Square_Graph():
    squares = [1,4,9,16]
    plt.plot(squares)
    plt.show()

Square_Graph()
```

Output

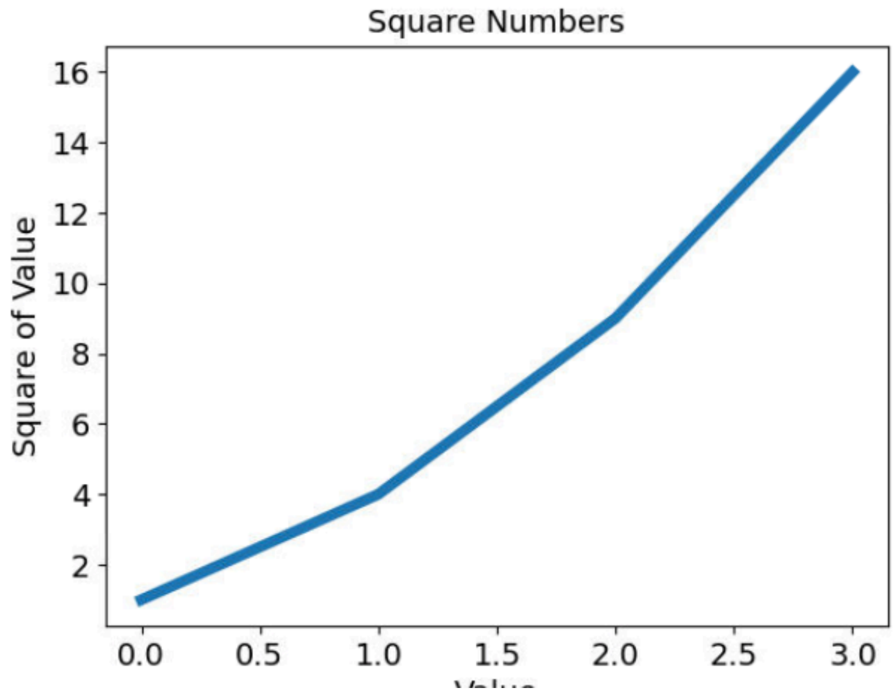


https://trinket.io/python or https://repl.it/

Example 5

```
# Plotting Square Numbers with Labels
import matplotlib.pyplot as plt

def Square_Graph_labels():
   squares = [1,4,9,16]

  plt.plot(squares, linewidth=5)
  plt.title("Square Numbers”,fontsize=14)

  plt.xlabel("Value",fontsize=14)
  plt.ylabel("Square of Value",fontsize=14)
  plt.tick_params(axis="both", labelsize=14)

    plt.show()
Square_Graph_labels()
```

Output

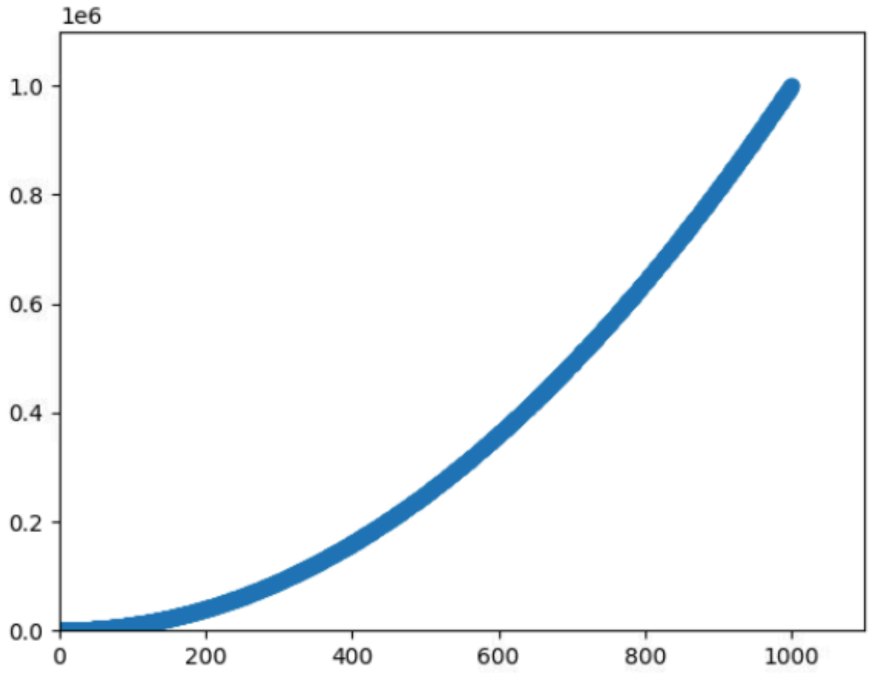


https://trinket.io/python or https://repl.it/

Example 6

```
# Plotting Squares by Automatically Generating the Numbers
import matplotlib.pyplot as plt
def Example6_Square_Automatic():

  x_values = list(range(1,1001)) #Automatically generating x values
  y_values = [x**2 for x in x_values] # Then generating y values

  plt.scatter(x_values,y_values, s=40) # Plotting Points, note use of scatter here

  plt.axis([0,1100, 0, 1100000]) # Defining x and y axis
  plt.show() # Allows Plot to be shown

Example6_Square_Automatic() #Calling the function
```

Output

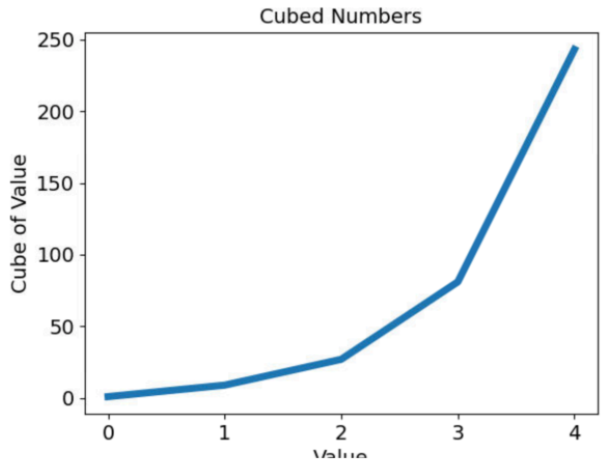


https://trinket.io/python or https://repl.it/

Exercises (3) & (4)

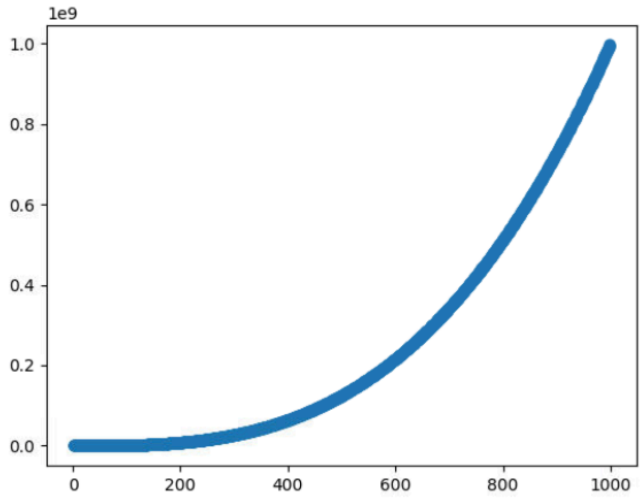
(3) Write a Program which will Plot the first 5 Cubed Numbers

So, you will get an output which looks something like this:-



(4) Now, Write a Program which will plot the 1st 1000 Cubed Numbers.

So, you will get an output which looks something like this:-



Hints/Help

```
# Plotting Square Numbers with Labels
import matplotlib.pyplot as plt

def Square_Graph_labels():
   squares = [1,4,9,16]

  plt.plot(squares, linewidth=5)
  plt.title("Square Numbers”,fontsize=14)

  plt.xlabel("Value",fontsize=14)
  plt.ylabel("Square of Value",fontsize=14)
  plt.tick_params(axis="both", labelsize=14)

    plt.show()
Square_Graph_labels()
```

```
# Plotting Squares by Automatically Generating the Numbers
import matplotlib.pyplot as plt
def Example6_Square_Automatic():

  x_values = list(range(1,1001)) #Automatically generating x values
  y_values = [x**2 for x in x_values] # Then generating y values

  plt.scatter(x_values,y_values, s=40) # Plotting Points, note use of scatter here

  plt.axis([0,1100, 0, 1100000]) # Defining x and y axis
  plt.show() # Allows Plot to be shown

Example6_Square_Automatic() #Calling the function
```

https://trinket.io/python or https://repl.it/

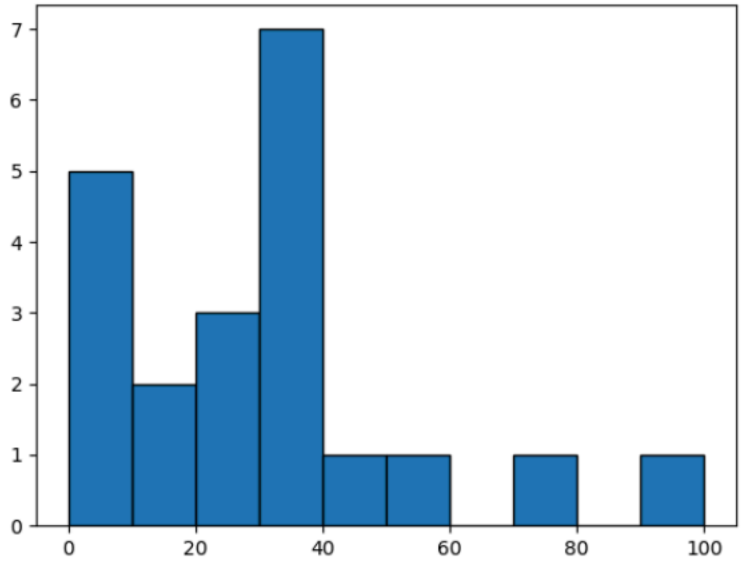
Example 7

Blank for now

Example 8

```
# Example 8 - Histograms
import numpy as np
import matplotlib.mlab as mlab
import matplotlib.pyplot as plt

def Example8():
  x = [21,22,23,4,5,6,77,8,9,10,31,32,33,34,35,36,37,18,49,50,100] #The Data
  b = [0,10,20,30,40,50,60,70,80,90,100] #The Intervals in the Histogram

  plt.hist(x,bins=b,edgecolor=‘black’) #Commands for constructing histogram
  plt.show()  # Displaying the Histogram

Example8() #Calling the function
```

Output

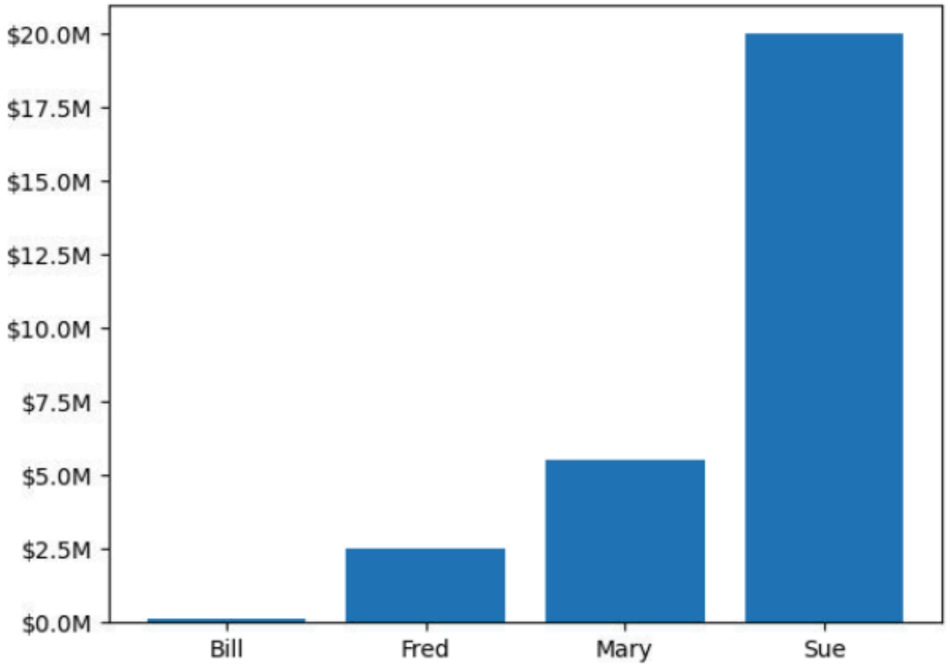


https://trinket.io/python or https://repl.it/

Example 9

```
# Constructing a Bar Chart
from matplotlib.ticker import FuncFormatter
import matplotlib.pyplot as plt
import numpy as np




def millions(x, pos):
    'The two args are the value and tick position'
    return '$%1.1fM' % (x * 1e-6)


def Example9():
  x = np.arange(4)
  money = [1.5e5, 2.5e6, 5.5e6, 2.0e7]
  formatter = FuncFormatter(millions)

  fig, ax = plt.subplots()
  ax.yaxis.set_major_formatter(formatter)
  plt.bar(x, money)
  plt.xticks(x, ('Bill', 'Fred', 'Mary', 'Sue'))
  plt.show()

Example9()
```

Output



https://trinket.io/python or https://repl.it/

That’s all for now folks!!

https://trinket.io/python or https://repl.it/