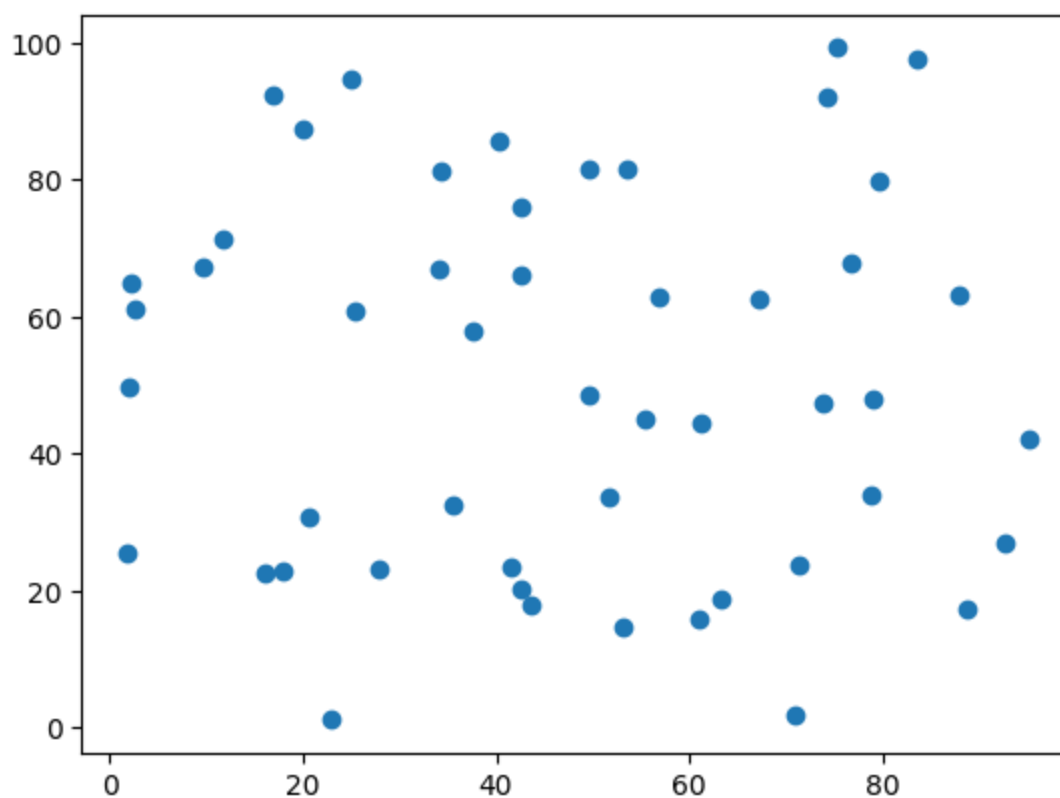


```
In [6]: import numpy as np
import matplotlib.pyplot as plt
X_data = np.random.random(50) * 100
Y_data = np.random.random(50) * 100
```

```
In [8]: ## SCATTER PLOT
#(x,y,c = color, s= size , marker = circle/stars ect , alpha = transparency)
plt.scatter(X_data , Y_data)
plt.show()
```



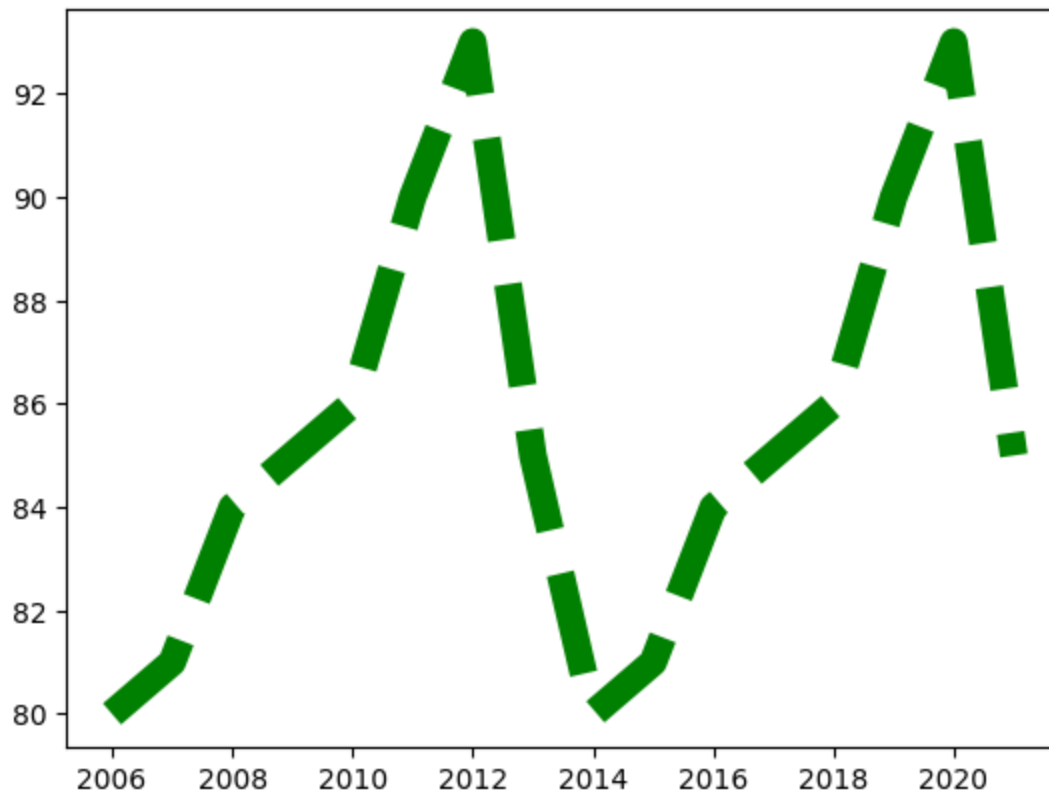
```
In [10]: #LINE PLOT

years = [2006 + x for x in range(16)]
weights = [80 , 81 , 84 , 85, 86 , 90 , 93 , 85 ,80 , 81 , 84 , 85, 86 , 90 , 93 ,

# plt.plot automaticly defaults to a line graph

plt.plot(years , weights , c = 'g' , lw = 10, linestyle = '--')
plt.plot()
```

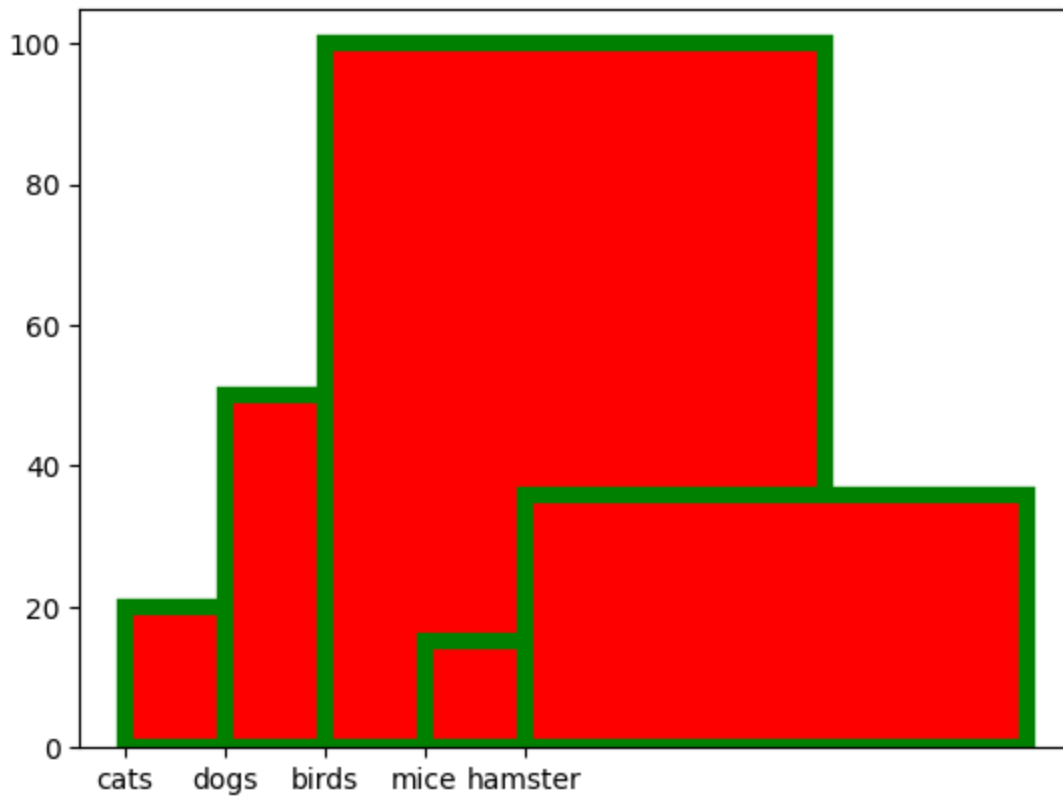
Out[10]: []



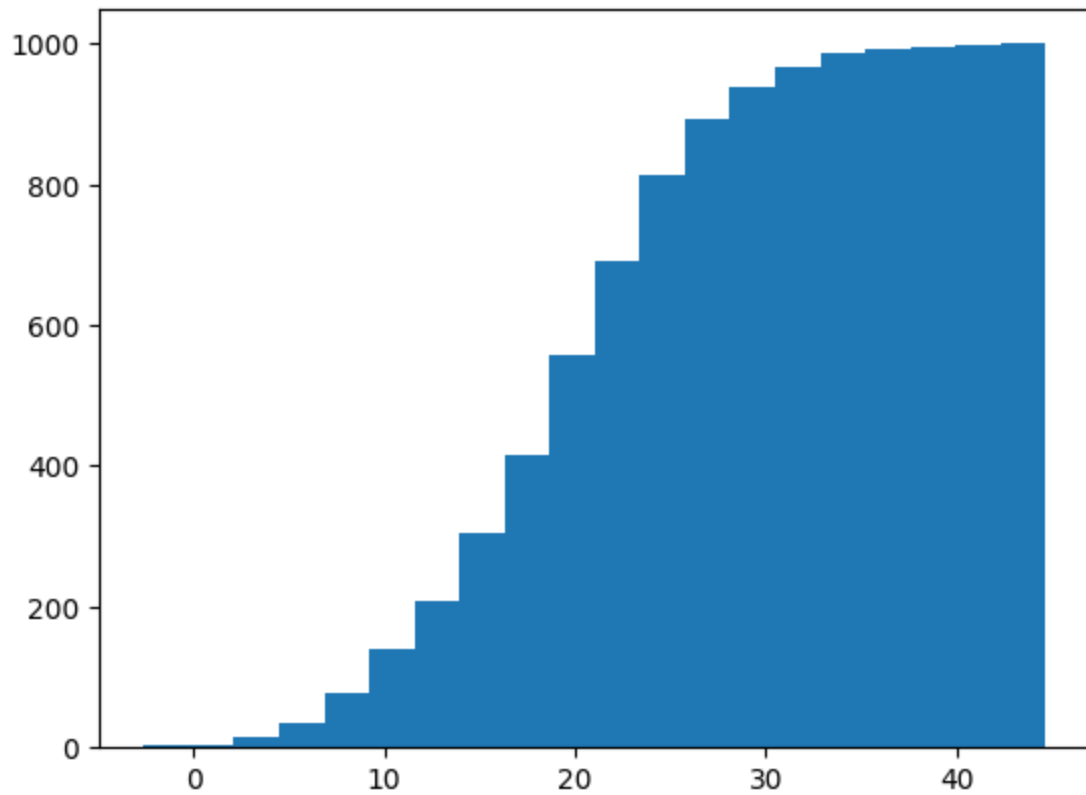
```
In [16]: #BAR PLOTS
pets = ['cats' , 'dogs' , 'birds' , 'mice' , 'hamster']
votes = [20, 50, 100, 15 , 36]

plt.bar(pets, votes , color = 'r' , align = 'edge', width = 5, edgecolor = 'g' , lw
plt.plot()
```

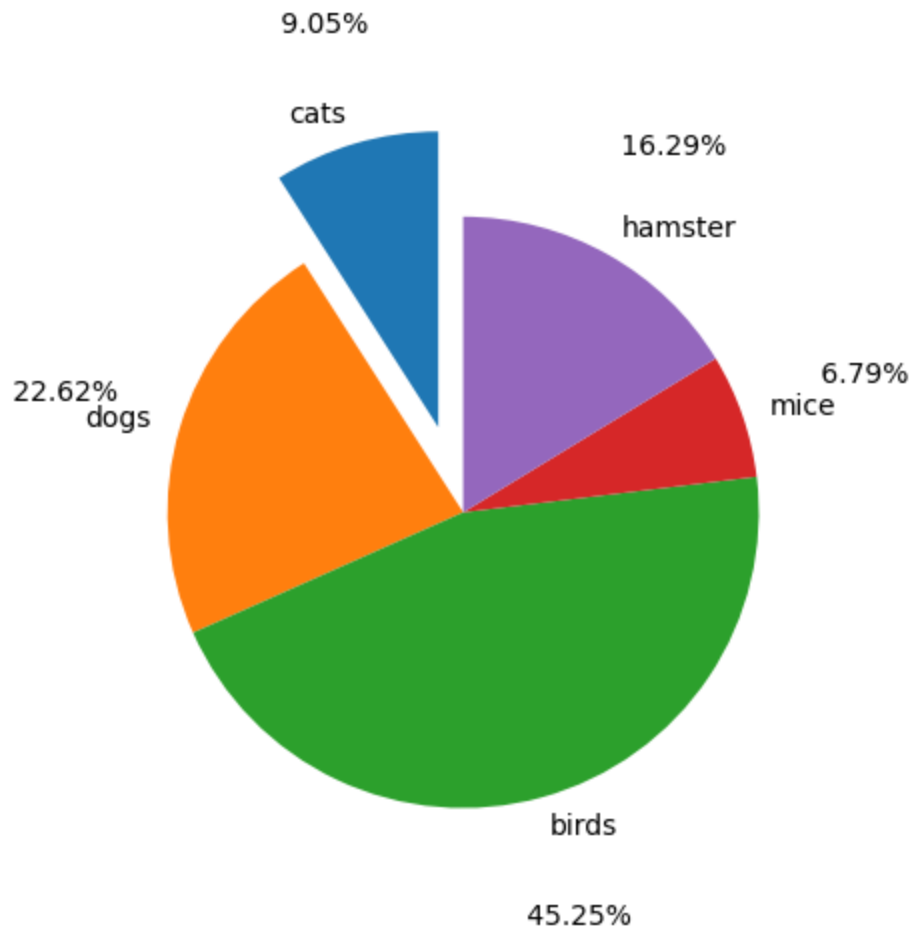
Out[16]: []



```
In [22]: # Histogram
# normal distribution, mean of 20 , SD 1.5 , 1000 people
ages = np.random.normal(20 , 7.5 , 1000)
# can specify bins
plt.hist(ages,
         bins= 20 , cumulative = True )
plt.show()
```

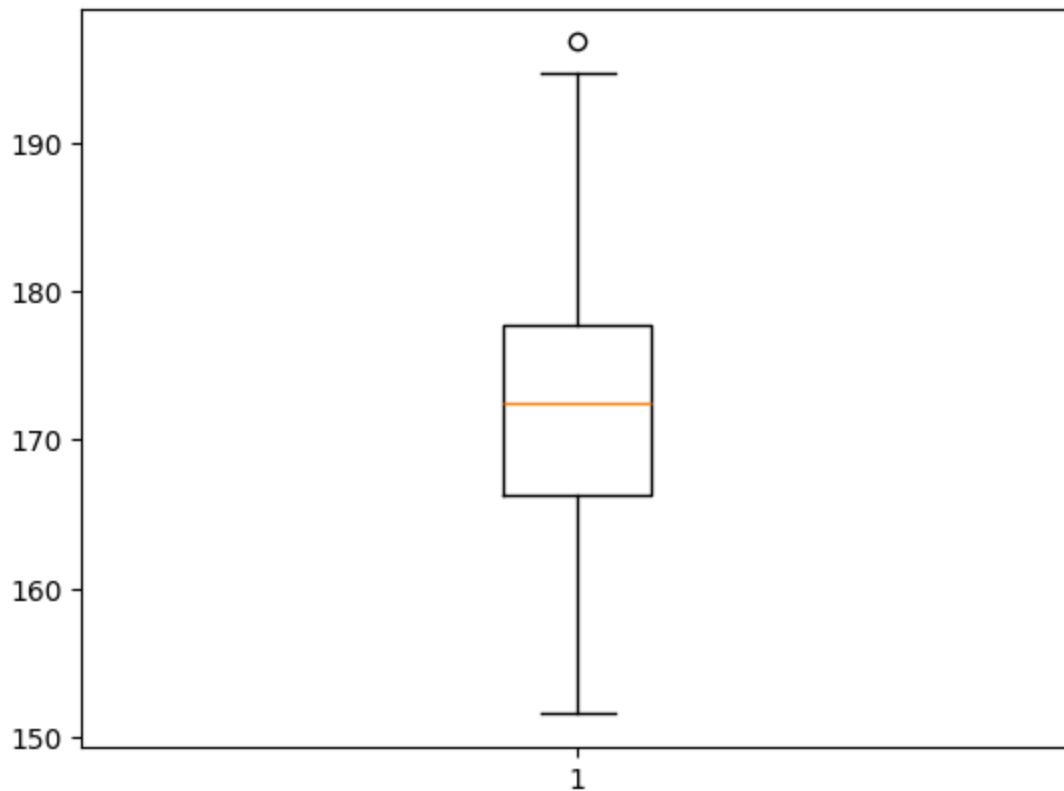


```
In [30]: #PIE CHART
pets = ['cats' , 'dogs' , 'birds', 'mice', 'hamster']
votes = [20, 50, 100, 15 , 36]
explodes = [ 0.3 ,0,0,0,0]
plt.pie(votes , labels = pets , explode =explodes , autopct = '% 0.2f%%' , pctdista
plt.show()
```



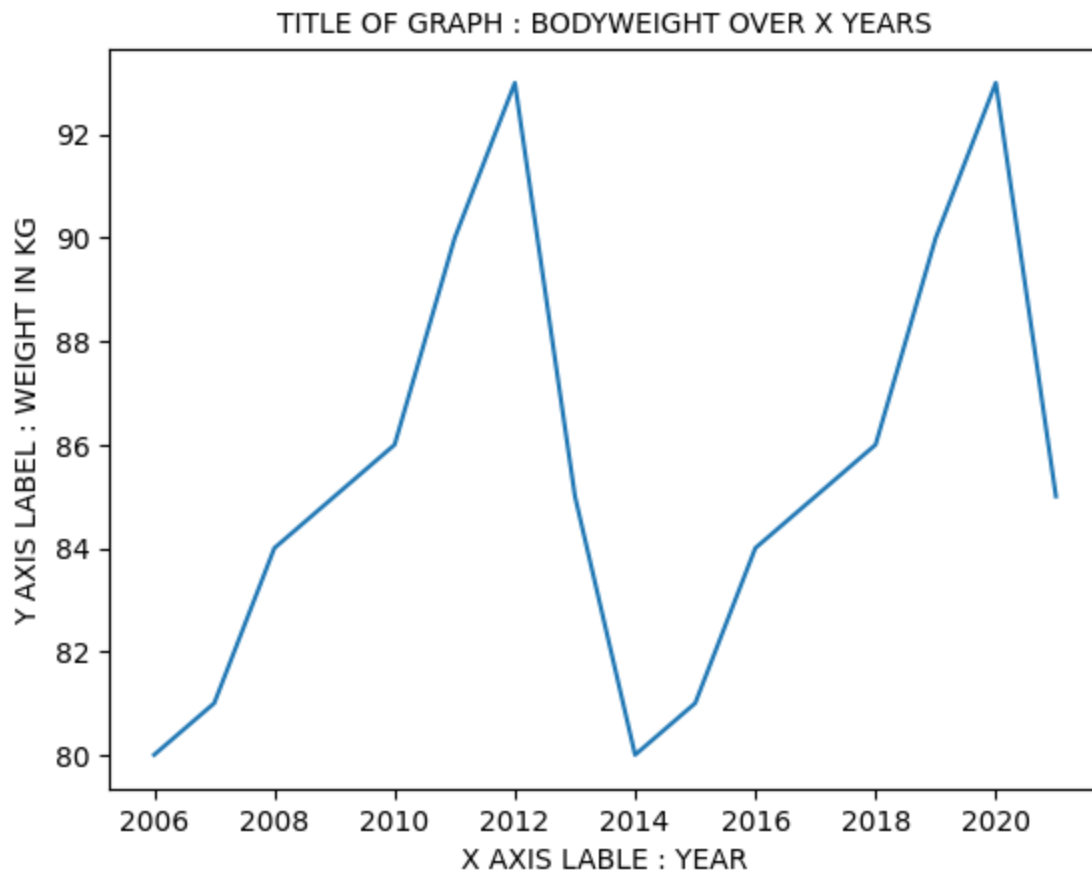
```
In [31]: ##BOX PLOT
heights = np.random.normal(172 ,8 ,300 )
plt.boxplot(heights)
```

```
Out[31]: {'whiskers': [<matplotlib.lines.Line2D at 0x25083f21c90>,
<matplotlib.lines.Line2D at 0x25080d60e10>],
'caps': [<matplotlib.lines.Line2D at 0x2508416edd0>,
<matplotlib.lines.Line2D at 0x2508416e590>],
'boxes': [<matplotlib.lines.Line2D at 0x2508415e4d0>],
'medians': [<matplotlib.lines.Line2D at 0x2508416f850>],
'fliers': [<matplotlib.lines.Line2D at 0x25080d89990>],
'means': []}
```



```
In [37]: ## plot customizations
plt.plot(years , weights )
plt.title('TITLE OF GRAPH : BODYWEIGHT OVER X YEARS' , fontsize = 10) #fontname can
plt.xlabel('X AXIS LABLE : YEAR')
plt.ylabel('Y AXIS LABEL : WEIGHT IN KG')
plt.plot()
```

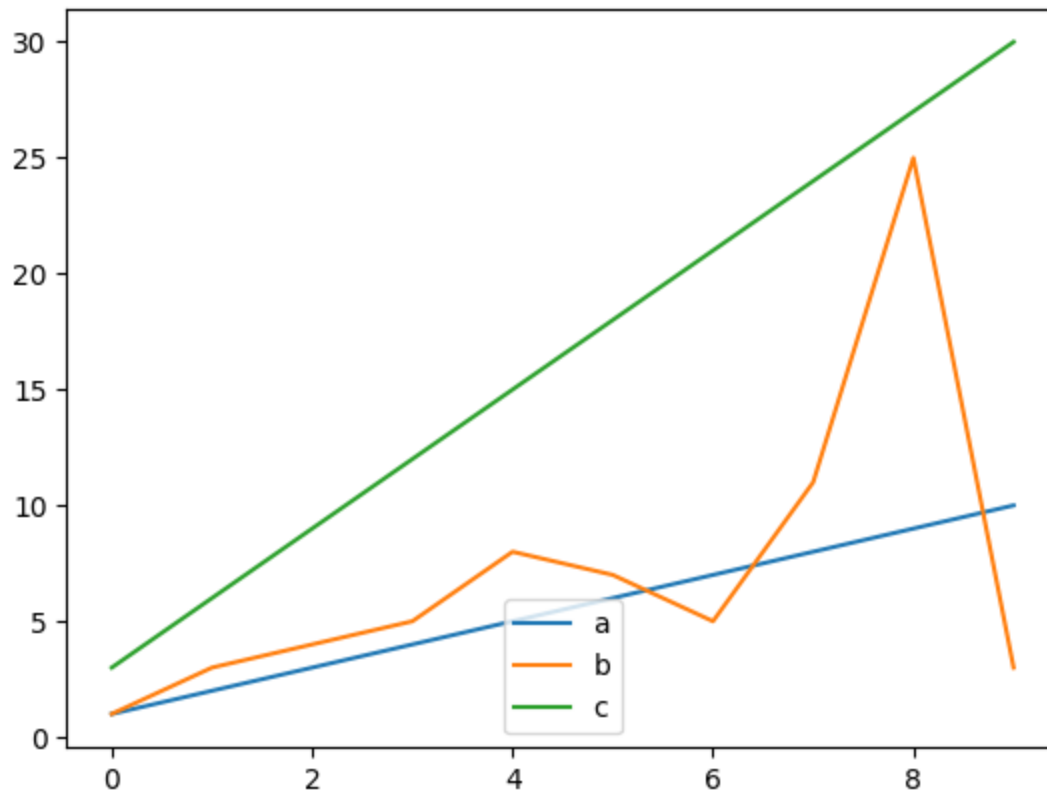
Out[37]: []



```
In [41]: ## MULTIPLE PLOTS
a= [1,2,3,4,5,6,7,8,9,10]
b = [1,3,4,5,8,7,5,11,25,3]
c = [3,6,9,12,15,18,21,24,27,30]

plt.plot(a, label = "a")
plt.plot(b, label = "b")
plt.plot(c, label = "c")
plt.legend(loc = 'lower center')

plt.show()
```



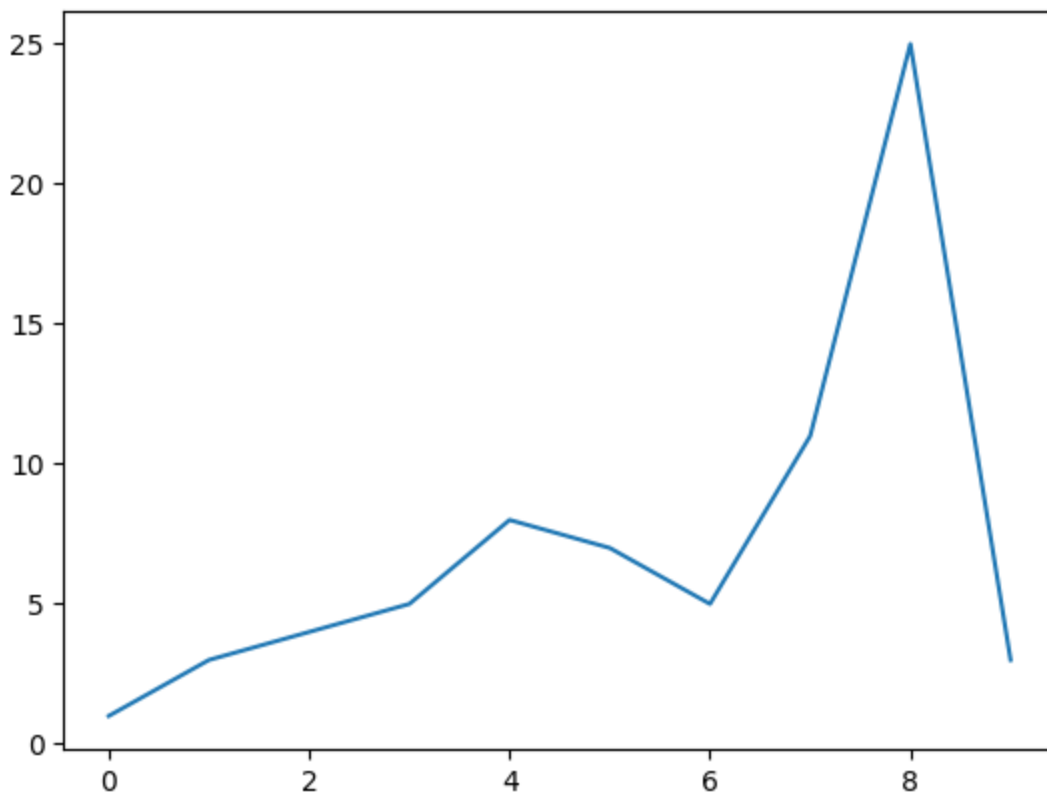
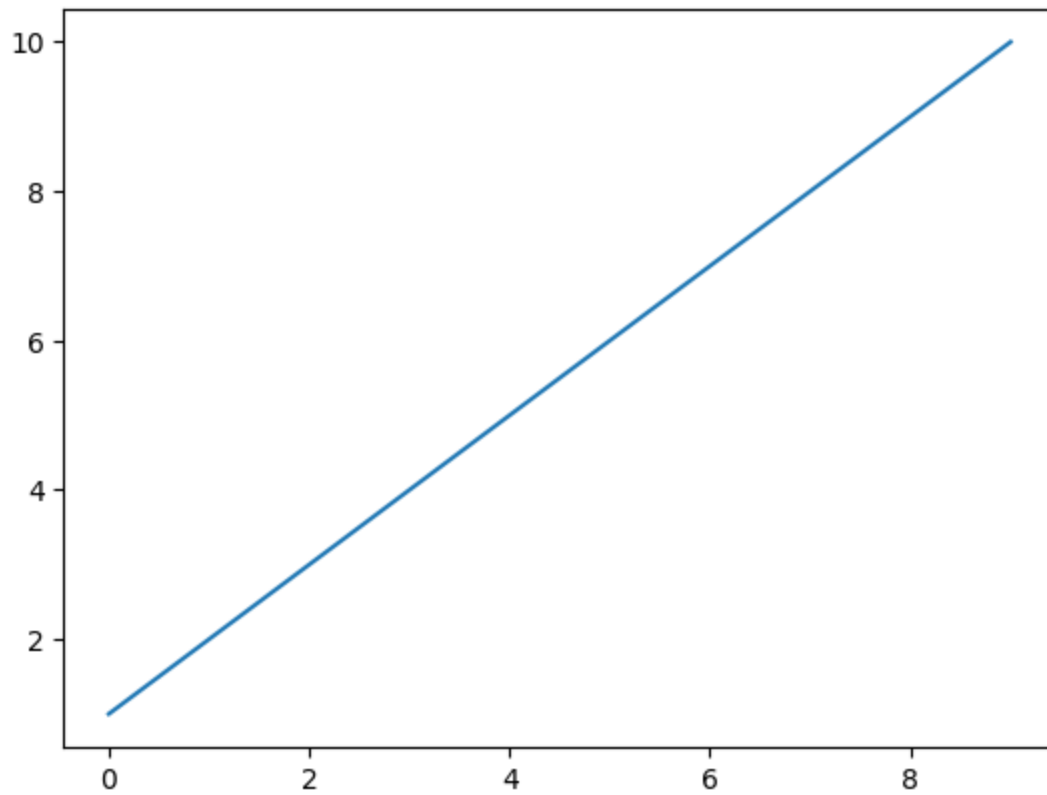
In [44]: *#MULTIPLE FIGURES AT ONCE*

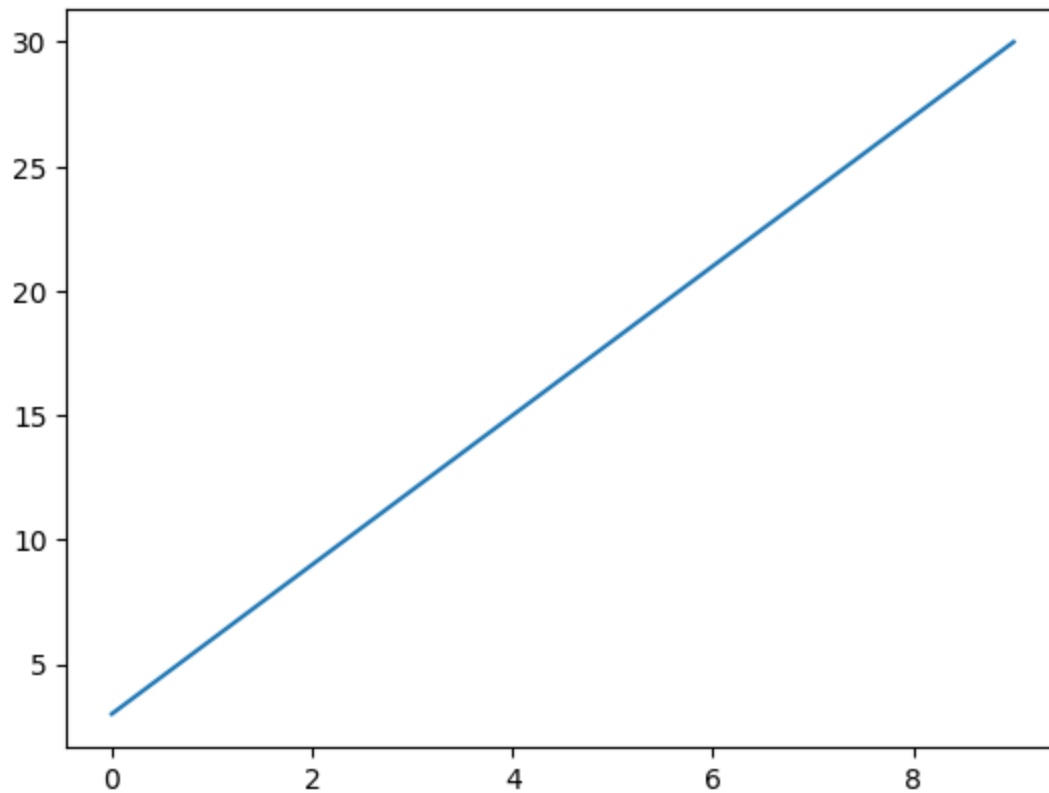
```
plt.figure(1)
plt.plot(a)

plt.figure(2)
plt.plot(b)

plt.figure(3)
plt.plot(c)

plt.show()
```



```
In [50]: ##SUBPLOTS

fig, axs = plt.subplots(2,2)

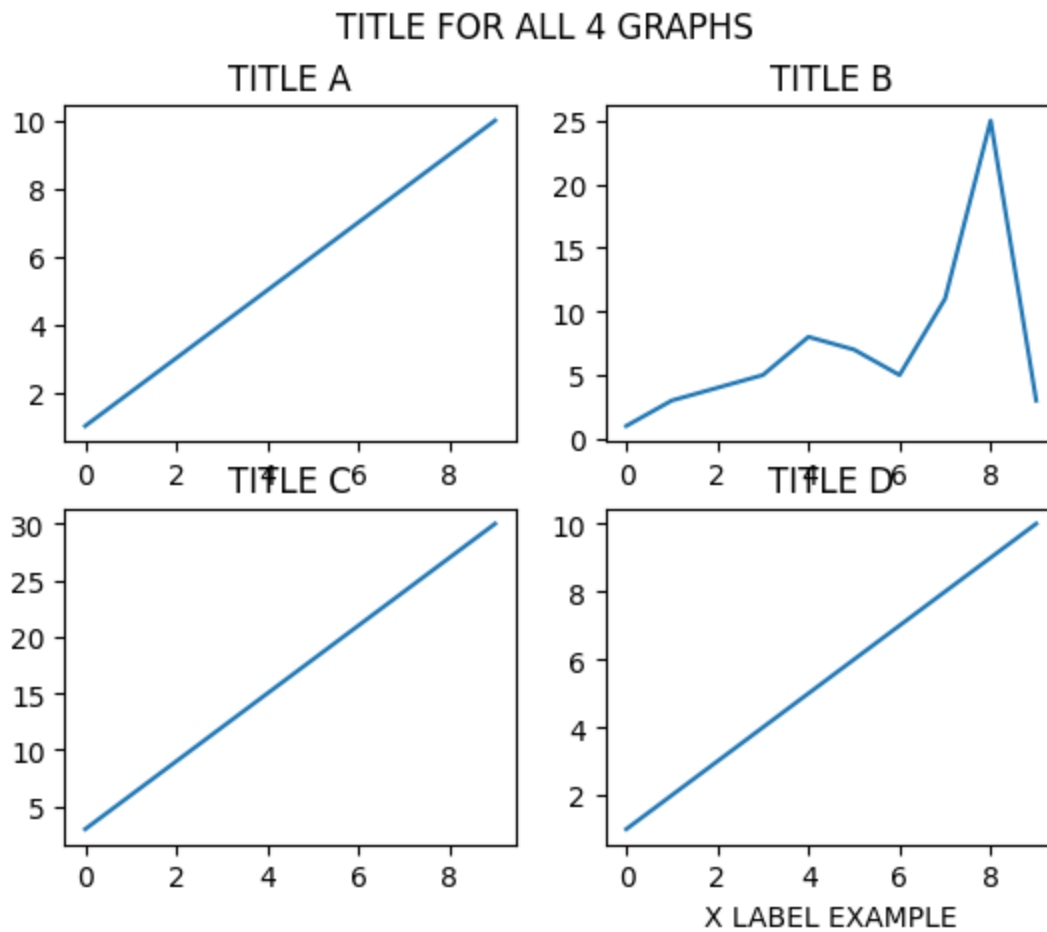
axs[0,0].plot(a)
axs[0,0].set_title('TITLE A')

axs[0,1].plot(b)
axs[0,1].set_title('TITLE B')

axs[1,0].plot(c)
axs[1,0].set_title('TITLE C')

axs[1,1].plot(a)
axs[1,1].set_title('TITLE D')
axs[1,1].set_xlabel('X LABEL EXAMPLE')

fig.suptitle('TITLE FOR ALL 4 GRAPHS')
plt.show()
```



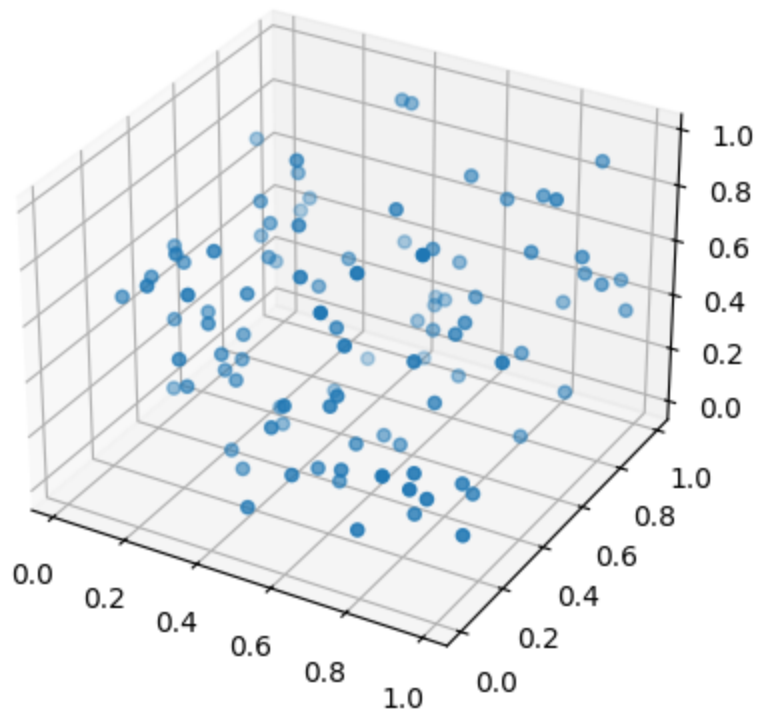
```
In [53]: ax = plt.axes(projection = '3d')

x = np.random.random(100)
y = np.random.random(100)
z = np.random.random(100)

ax.scatter(x,y,z)
ax.set_title('3D PLOT')
plt.show()

#ABLE TO LOOK AT 3d MODEL; not in notebook
```

3D PLOT



In []: