In [1]:
```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

In [6]:
```python
data = pd.read_csv('students.csv')
data.head()
```

Out[6]:

	Hours	Scores
0	2.5	21
1	5.1	47
2	3.2	27
3	8.5	75
4	3.5	30

In [7]:
```python
data.shape
```

Out[7]: (25, 2)

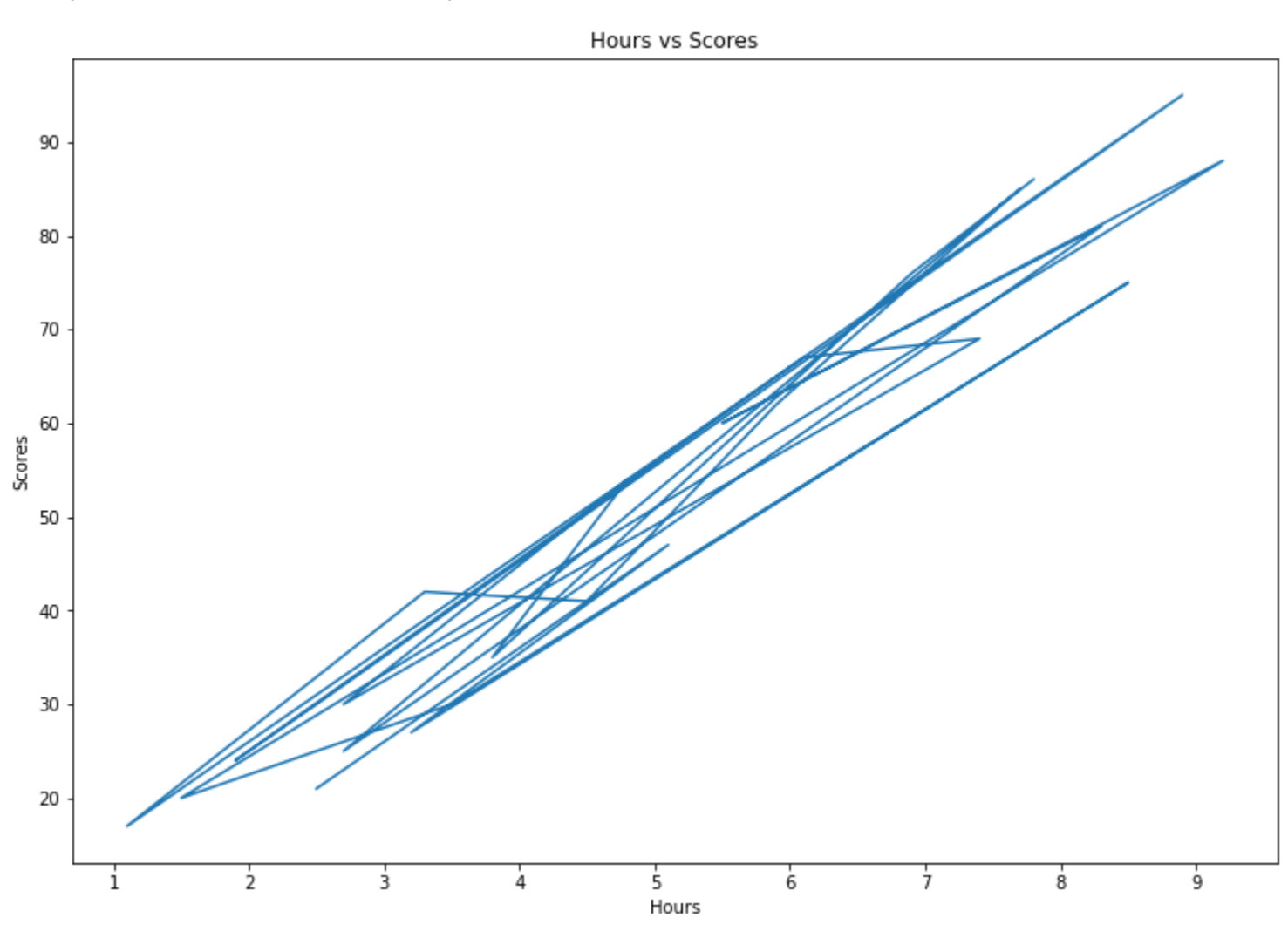
In [8]:
```python
data.dtypes
```

Out[8]:
```
Hours     float64
Scores      int64
dtype: object
```

In [10]:
```python
Hours = data['Hours']
Scores = data['Scores']
```

In [22]:
```python
fig = plt.figure()
myaxes = fig.add_axes([0.1,0.1,1.6,1.6])
myaxes.plot(Hours,Scores)
myaxes.set_xlabel('Hours')
myaxes.set_ylabel('Scores')
myaxes.set_title('Hours vs Scores')
```

Out[22]: Text(0.5, 1.0, 'Hours vs Scores')



In [30]:
```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

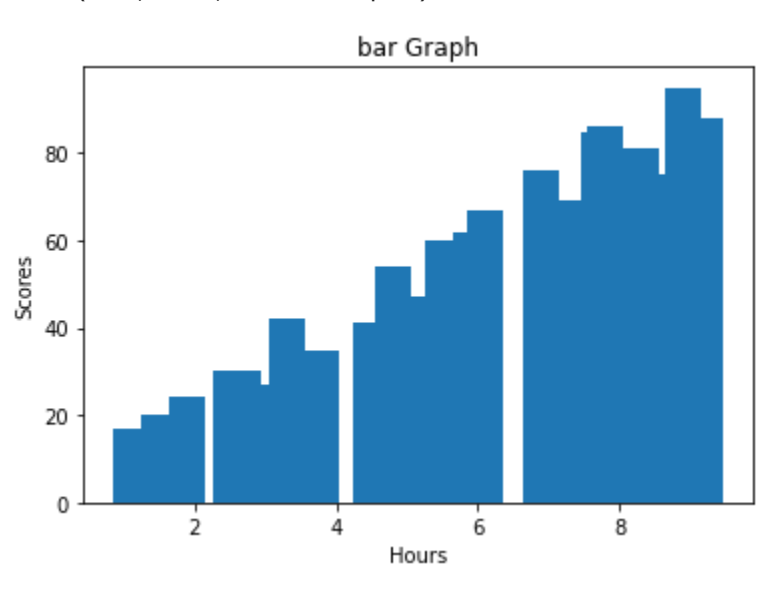
In [63]:
```python
data = pd.read_csv('students.csv')
data.head()
```

Out[63]:

	Hours	Scores
0	2.5	21
1	5.1	47
2	3.2	27
3	8.5	75
4	3.5	30

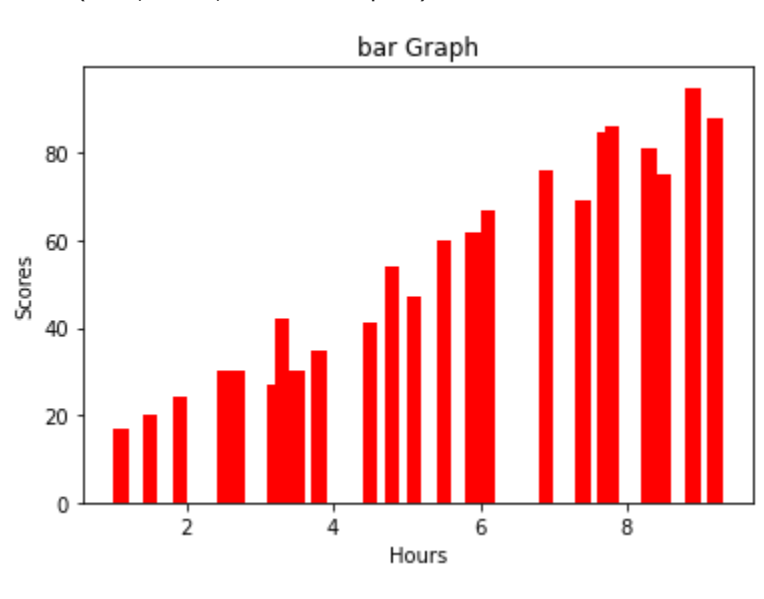
In [88]:
```python
plt.bar(Hours,Scores,width=0.5)
plt.xlabel("Hours")
plt.ylabel("Scores")
plt.title('bar Graph')
```

Out[88]: Text(0.5, 1.0, 'bar Graph')



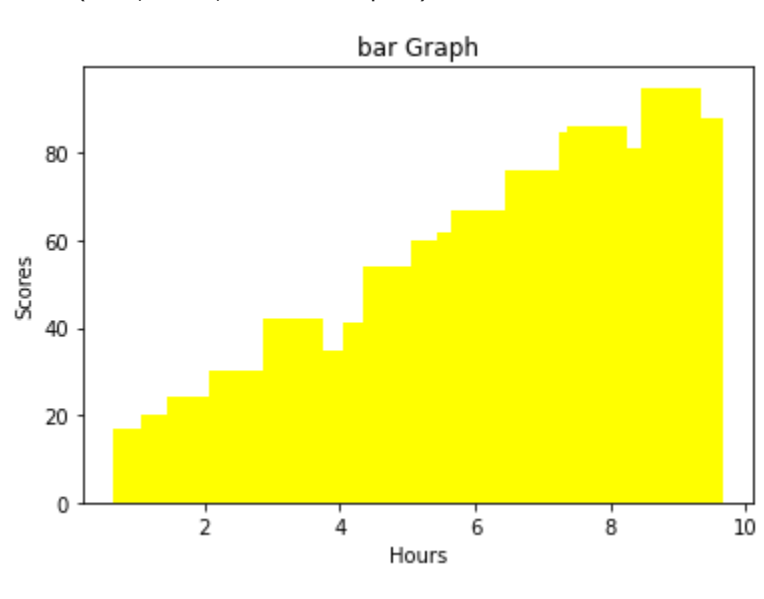
In [89]:
```python
plt.bar(Hours,Scores,width=0.2,color=['red'])
plt.xlabel("Hours")
plt.ylabel("Scores")
plt.title('bar Graph')
```

Out[89]: Text(0.5, 1.0, 'bar Graph')

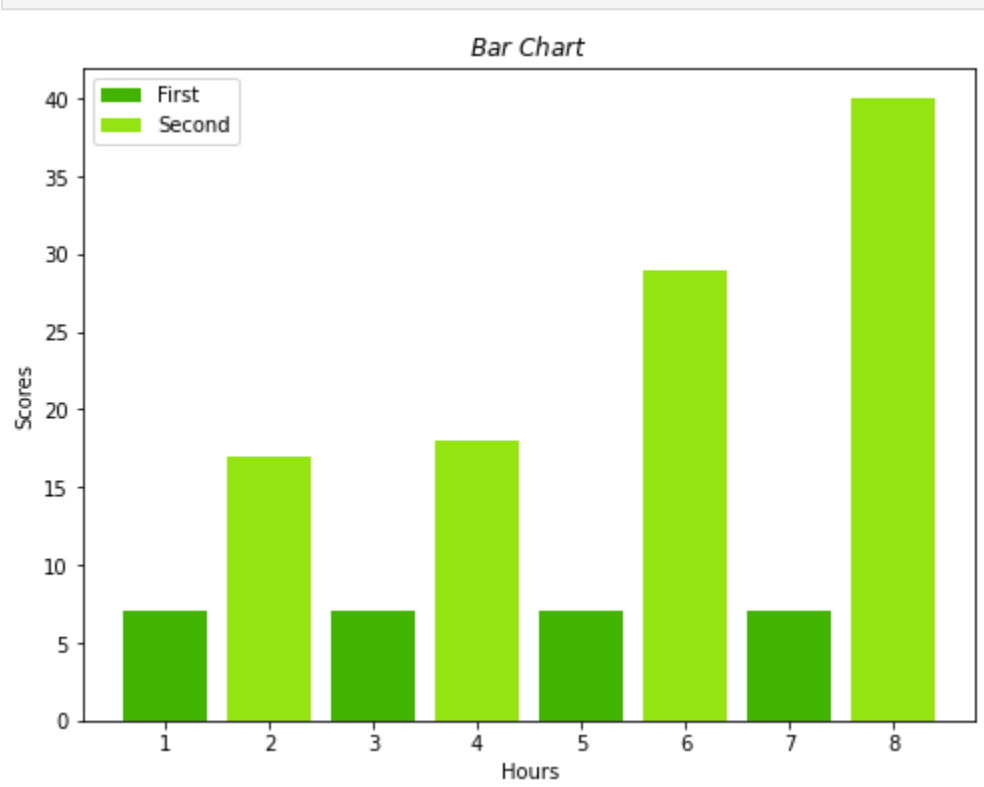


In [99]:
```python
plt.bar(Hours,Scores,width=0.9,color=['yellow'])
plt.xlabel("Hours")
plt.ylabel("Scores")
plt.title('bar Graph')
```

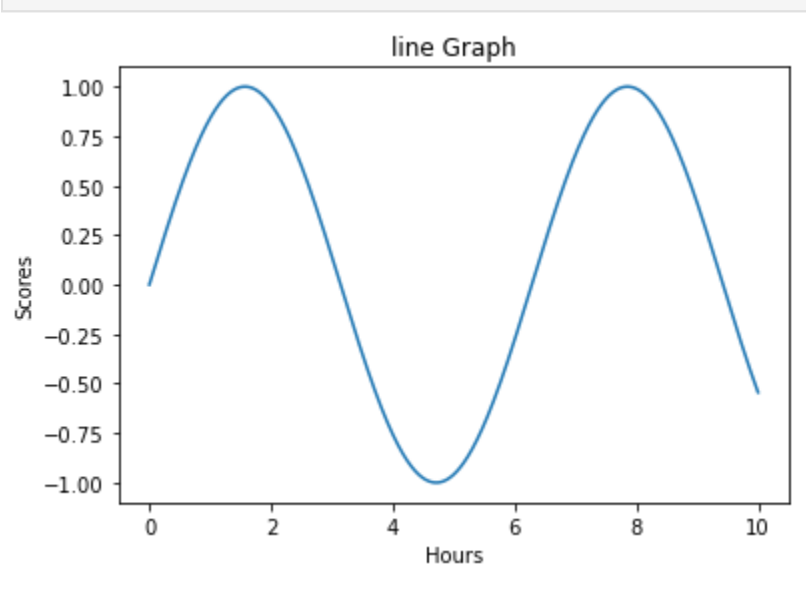
Out[99]: Text(0.5, 1.0, 'bar Graph')



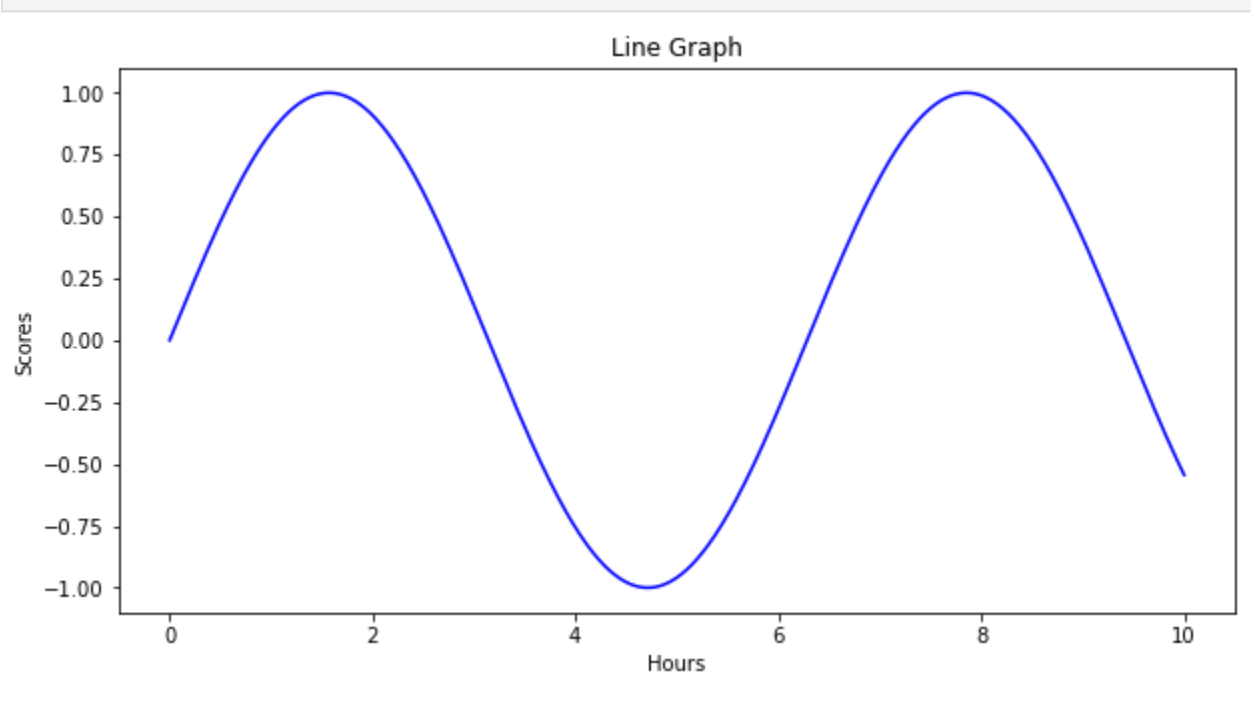
In [160…
```python
#Plotting multiple sets of data
x1= [1,3,5,7]
x2=[2,4,6,8]
y1 = [7,7,7,7]
y2= [17,18,29,40]
plt.figure(figsize=(8,6))
ax = plt.axes()
ax.set_facecolor("white")
plt.bar(x1,y1,label = "First",color = '#42B300') # First set of data
plt.bar(x2,y2,label = "Second",color = '#94E413') # Second set of data
plt.xlabel('$X$')
plt.ylabel('$Y$')
plt.title ('$Bar $ $ Chart$')
plt.legend()
plt.xlabel("Hours")
plt.ylabel("Scores")
plt.show()
```



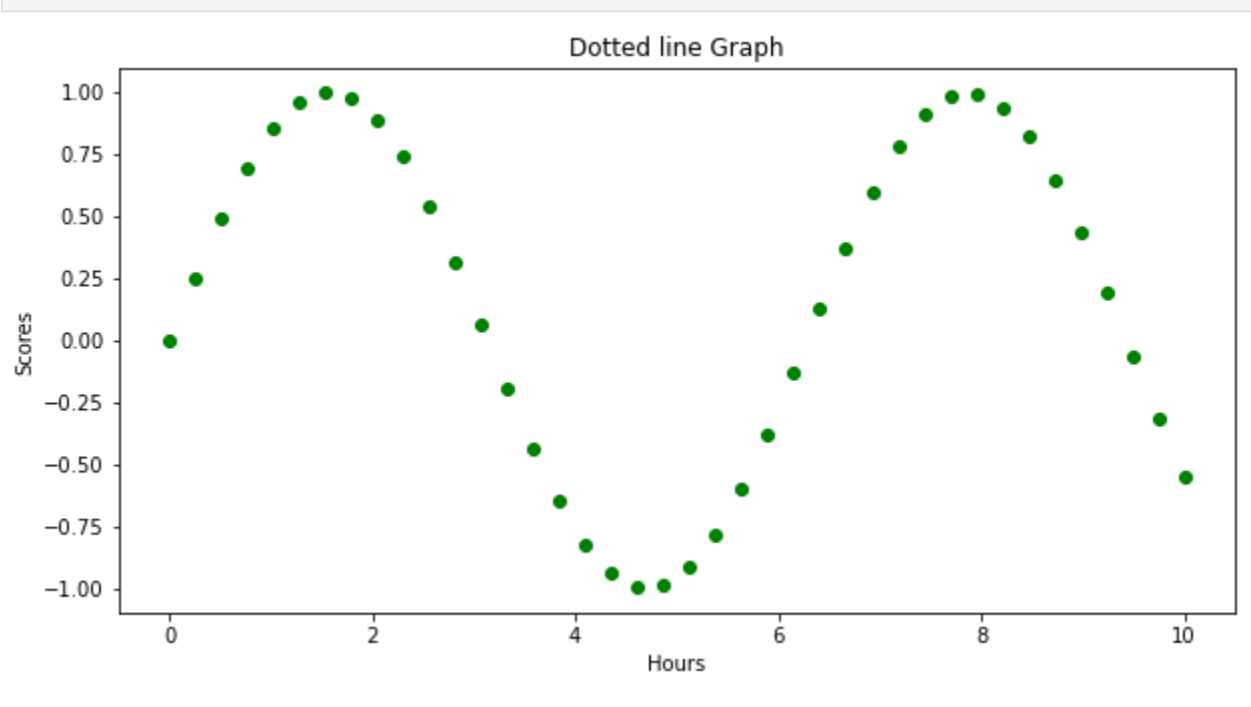
In [167…
```python
x = np.linspace(0, 10, 1000)
y = np.sin(x) # Sine Graph
plt.plot(x,y)
plt.xlabel("Hours")
plt.ylabel("Scores")
plt.title('line Graph')
plt.show()
```



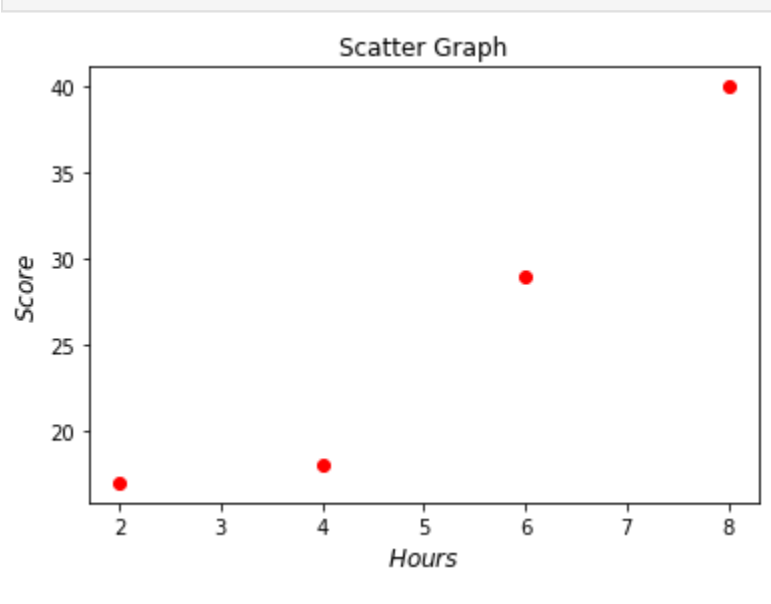
In [168…
```python
# Solid blue line will be plotted using the argument "b-"
plt.figure(figsize=(10,5))
x = np.linspace(0, 10, 1000)
y = np.sin(x) # Sine Graph
plt.plot(x,y,'b-')
plt.xlabel("Hours")
plt.ylabel("Scores")
plt.title('Line Graph')
plt.show()
```



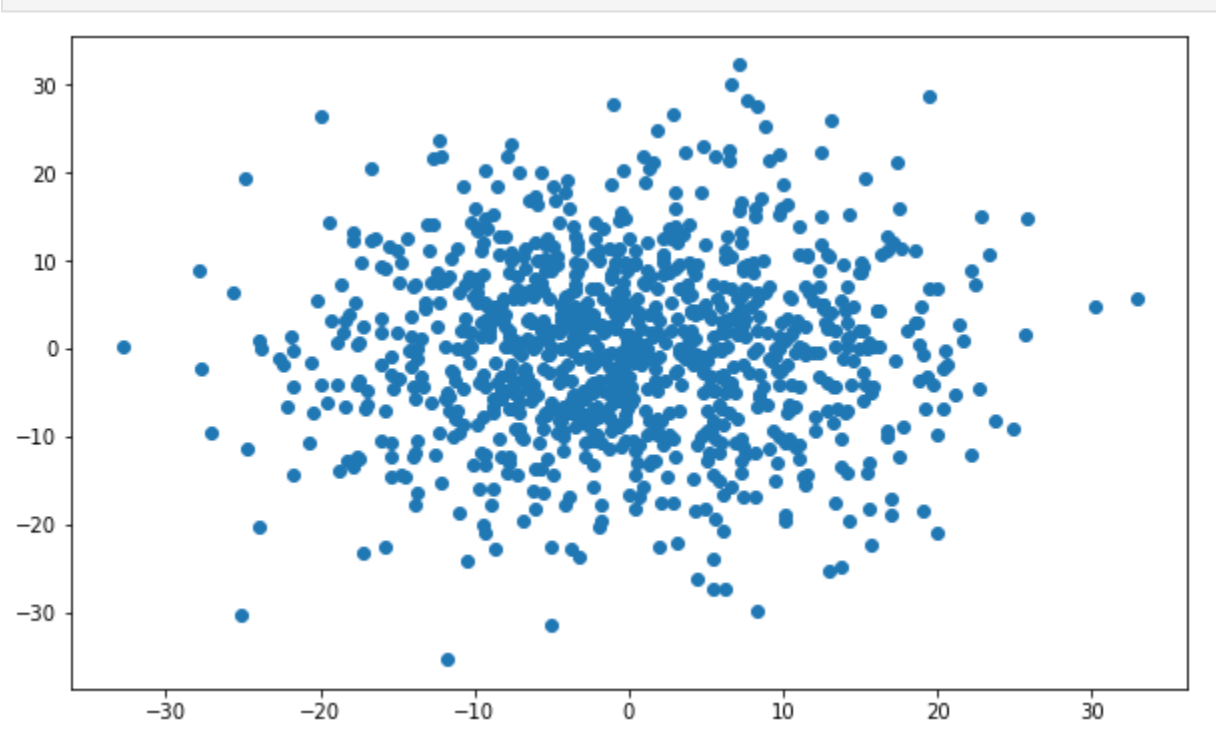
In [170…
```python
plt.figure(figsize=(10,5))
x = np.linspace(0, 10, 40)
y = np.sin(x) # Sine Graph
plt.plot(x,y,'go')
plt.xlabel("Hours")
plt.ylabel("Scores")
plt.title('Dotted line Graph')
plt.show()
```



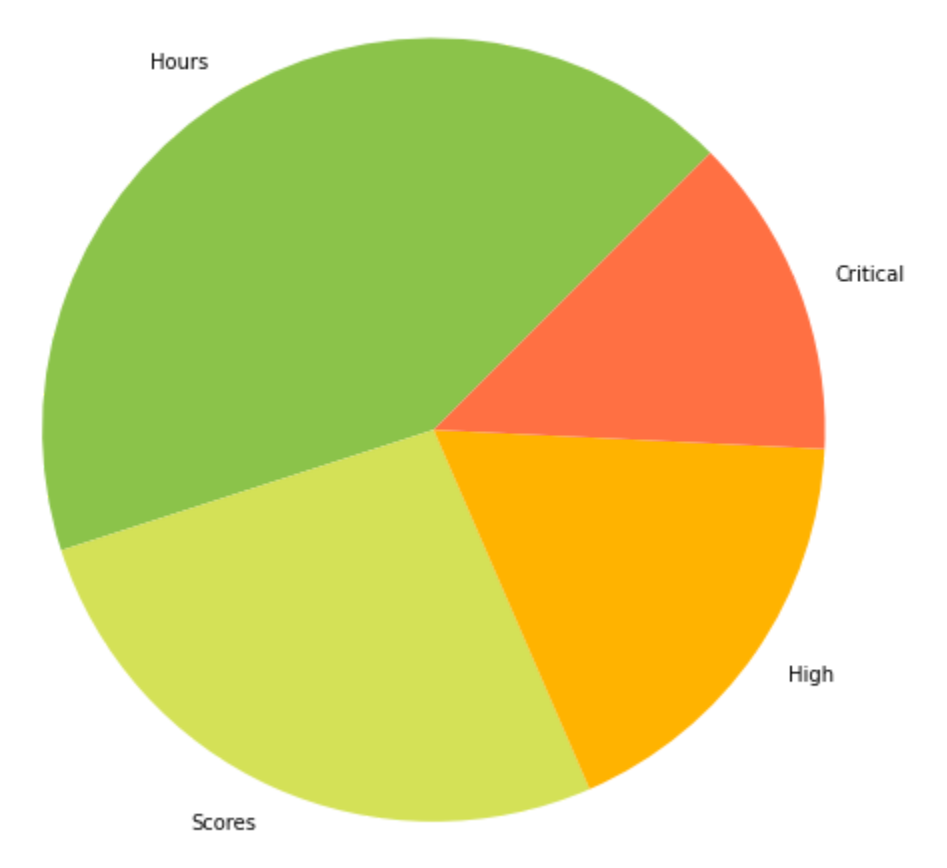
In [172…
```python
plt.scatter(x2,y2 ,color = 'r')
plt.xlabel('$Hours$' , fontsize = 12)
plt.ylabel('$Score$' , fontsize = 12)
plt.title ('Scatter Graph')
plt.show()
```



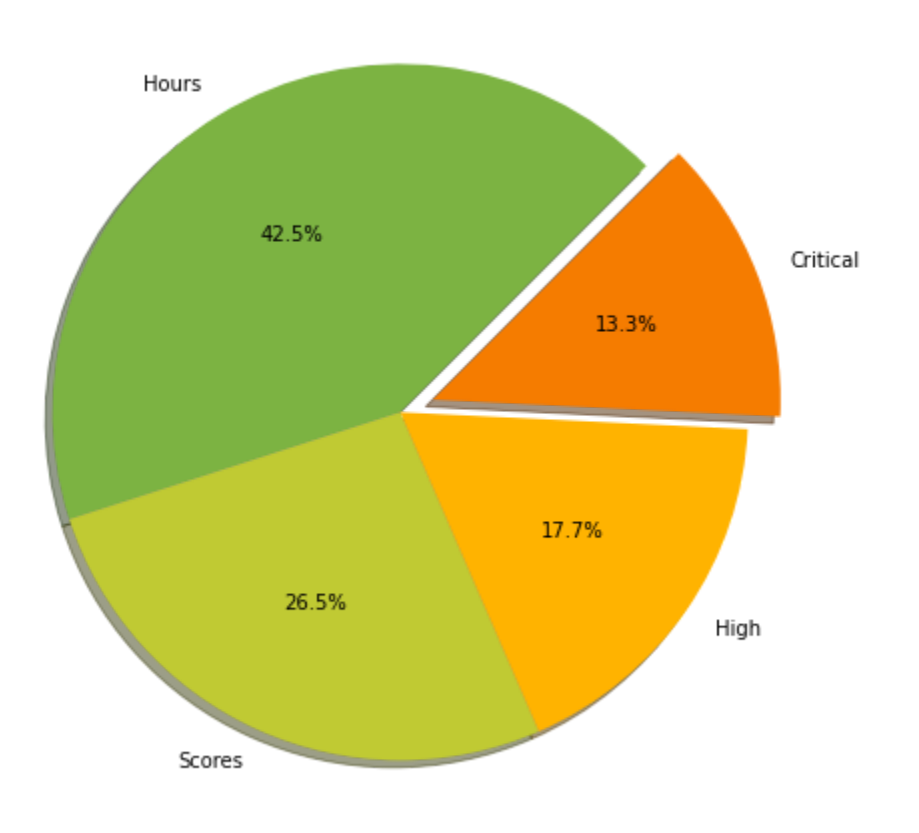
In [173…
```python
plt.figure(figsize=(10,6))
x = np.random.normal(0,10,1000)
y = np.random.normal(0,10,1000)
plt.scatter(x,y)
plt.show()
```



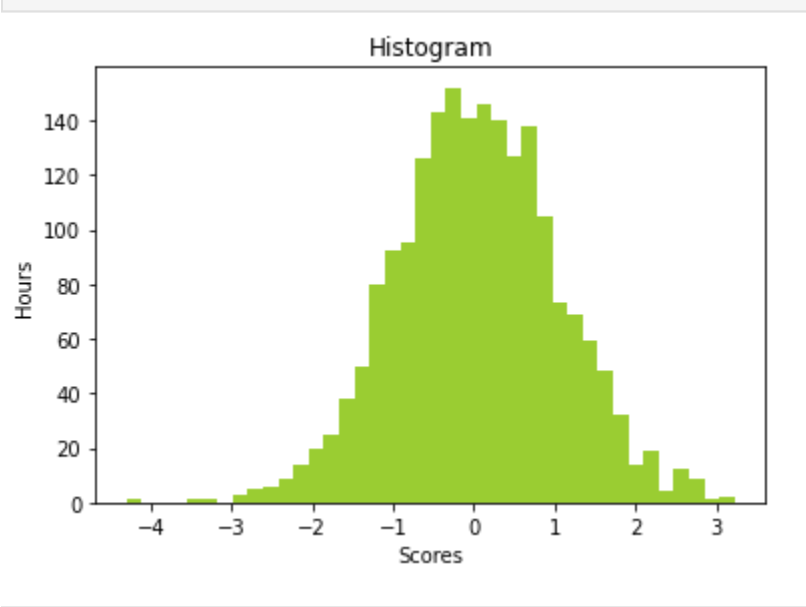
In [175…
```python
plt.figure(figsize=(9,9))
area = [48 , 30 , 20 , 15]
labels = ['Hours' , 'Scores' , 'High' , 'Critical']
colors = ['#8BC34A','#D4E157','#FFB300','#FF7043']
plt.pie (area , labels= labels , colors= colors , startangle=45)
plt.show()
```



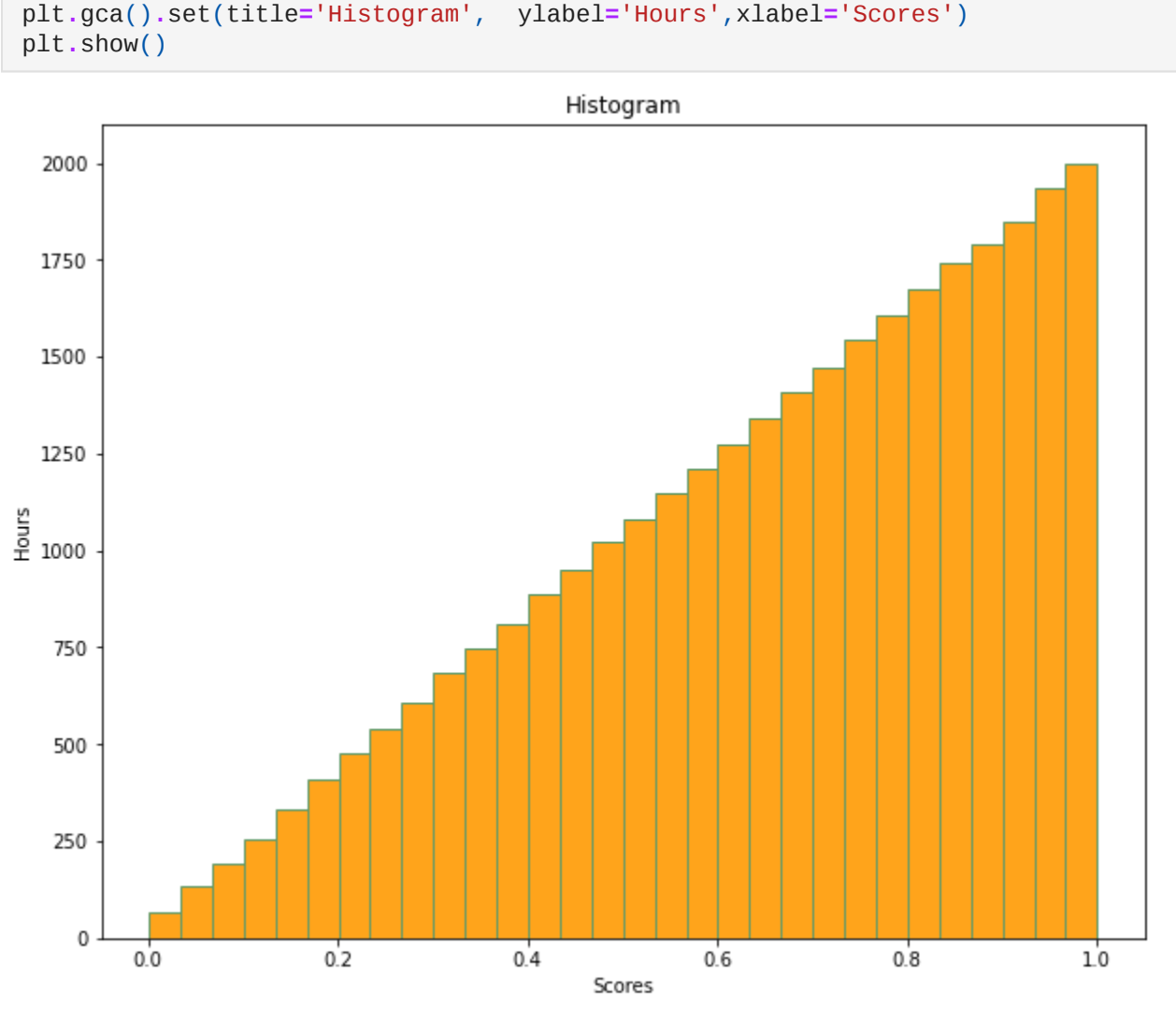
In [176…
```python
plt.figure(figsize=(8,8))
area = [48 , 30 , 20 , 15]
labels = ['Hours' , 'Scores' , 'High' , 'Critical']
colors = ['#7CB342','#C0CA33','#FFB300','#F57C00']
# explode = [0,0,0,0.1] will explode the fourth slice
plt.pie (area , labels= labels , colors= colors , startangle=45 , autopct='%1.1f%%' , shadow=True, explode = (0, 0, 0, 0.1))
plt.show()
```



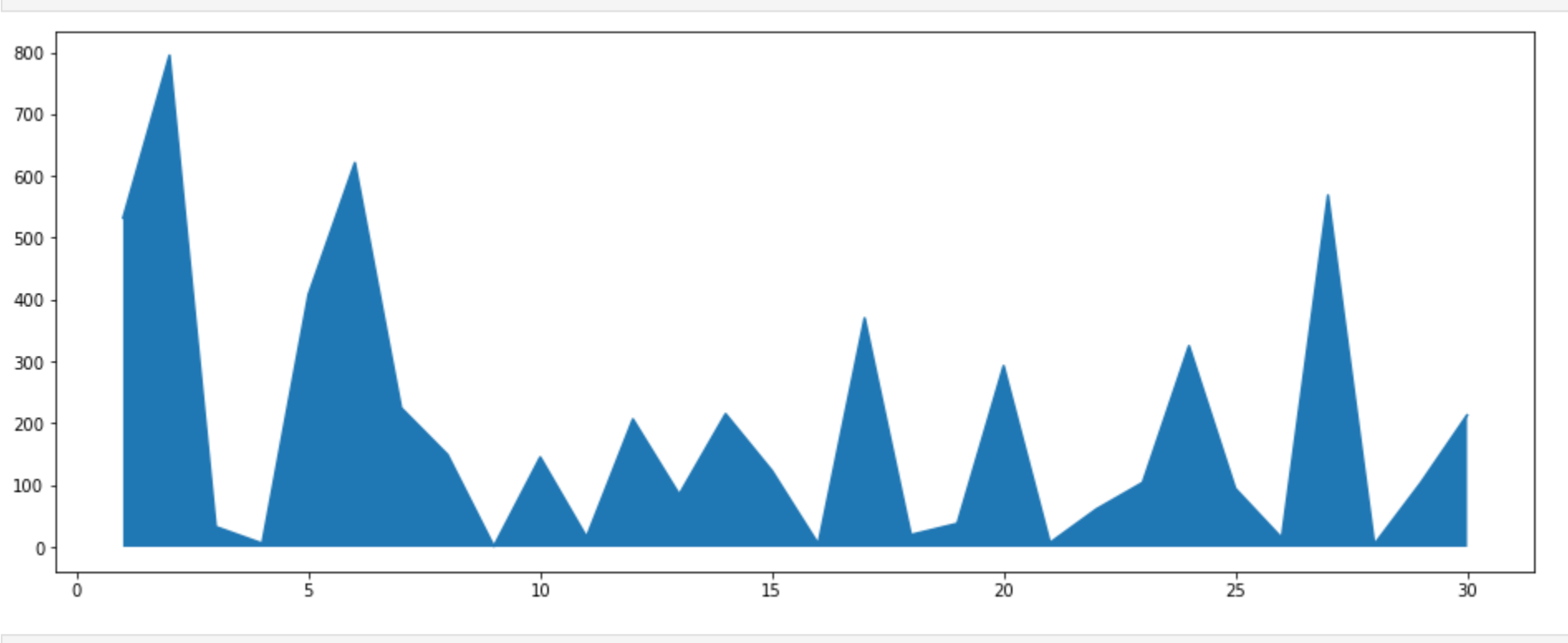
In [177…
```python
x = np.random.normal(size = 2000)
plt.hist(x, bins=40, color='yellowgreen')
plt.gca().set(title='Histogram', ylabel='Hours',xlabel='Scores')
plt.show()
```



In [178…
```python
# Using Histogram to plot a cumulative distribution function
plt.figure(figsize=(10,8))
x = np.random.rand(2000)
plt.hist(x, bins=30 ,color='#ffa41b' , edgecolor="#639a67",cumulative=True)
plt.gca().set(title='Histogram',  ylabel='Hours',xlabel='Scores')
plt.show()
```



In [180…
```python
x = np.arange(1,31)
y = np.random.normal(10,11,size=30)
y = np.square(y)
plt.figure(figsize=(16,6))
plt.plot(x,y)
plt.fill_between(x, y)
plt.show()
```



In [ ]: