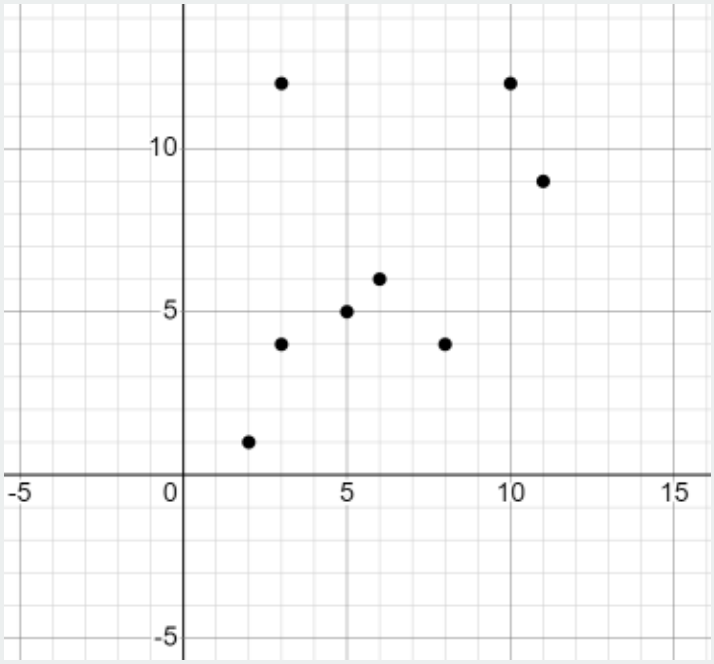
Question -1: Consider the time series data given below: **0.5 Marks**

X_i	8	3	2	10	11	3	6	5
Y_i	4	12	1	12	9	4	6	1

i. Use the least square method to determine the equation of line of best fit for the data. Then plot the line.

Plot the points on graph



x	8	3	2	10	11	3	6	5
y	4	12	1	12	9	4	6	1
X(mean)	6							
Y(men)	6.125							

i	xi	yi	xi-X	yi-Y	(xi-X)(yi-Y)	(xi-X)^2		
1	8	4	2	-2.125	-4.25	4		
2	3	12	-3	5.875	-17.625	9		
3	2	1	-4	-5.125	20.5	16		
4	10	12	4	5.875	23.5	16		
5	11	9	5	2.875	14.375	25		
6	3	4	-3	-2.125	6.375	9		
7	6	6	0	-0.125	0	0		
8	5	1	-1	-5.125	5.125	1		
Sum					48	80		

Calculate the slope.

$$m = \frac{\sum_{i=1}^{n}(x_i - \overline{X})(y_i - \overline{Y})}{\sum_{i=1}^{n}(x_i - \overline{X})^2} = \frac{48}{80} = 0.6$$

Calculate the y -intercept.

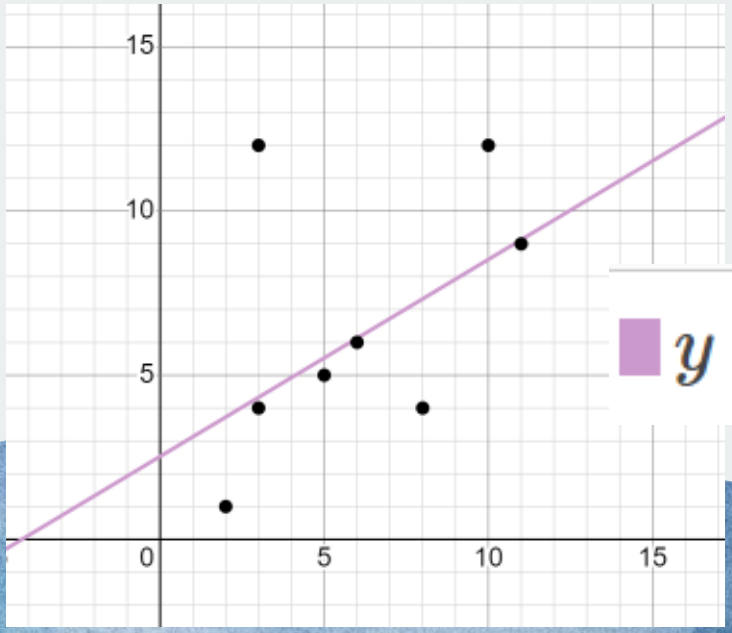
Calculate the y -intercept.

Use the formula to compute the y -intercept.

$$b = \overline{Y} - m\overline{X}$$

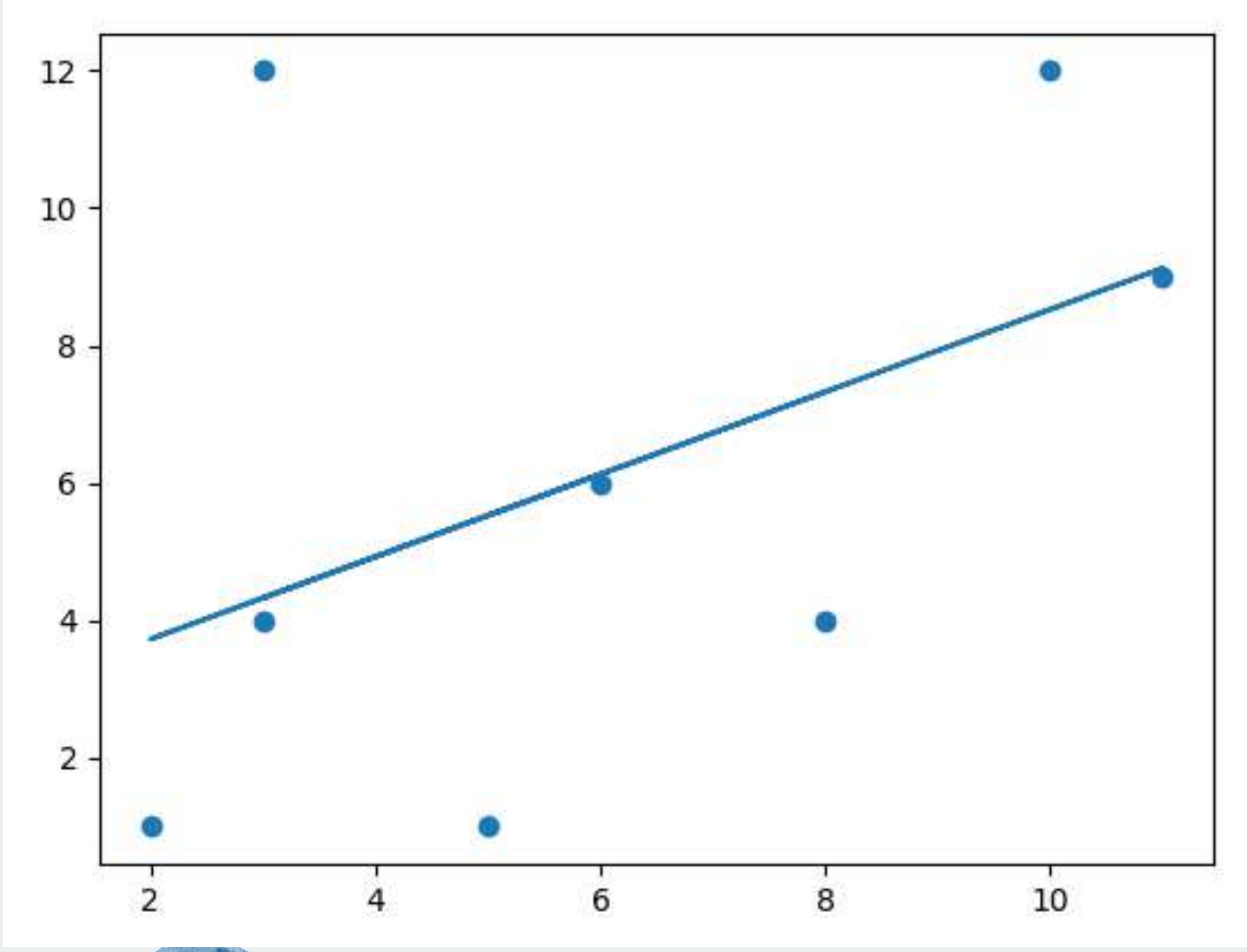
= (6.125-0.6*6)

=2.525



$y = 0.6x + 2.525$

```python
import numpy as np
import matplotlib.pyplot as plt
x=np.array([8,3,2,10,11,3,6,5])
y=np.array([4,12,1,12,9,4,6,1])
m,c=np.polyfit(x,y,1)
plt.scatter(x,y)
plt.plot(x,x*m+c)
plt.show()
```



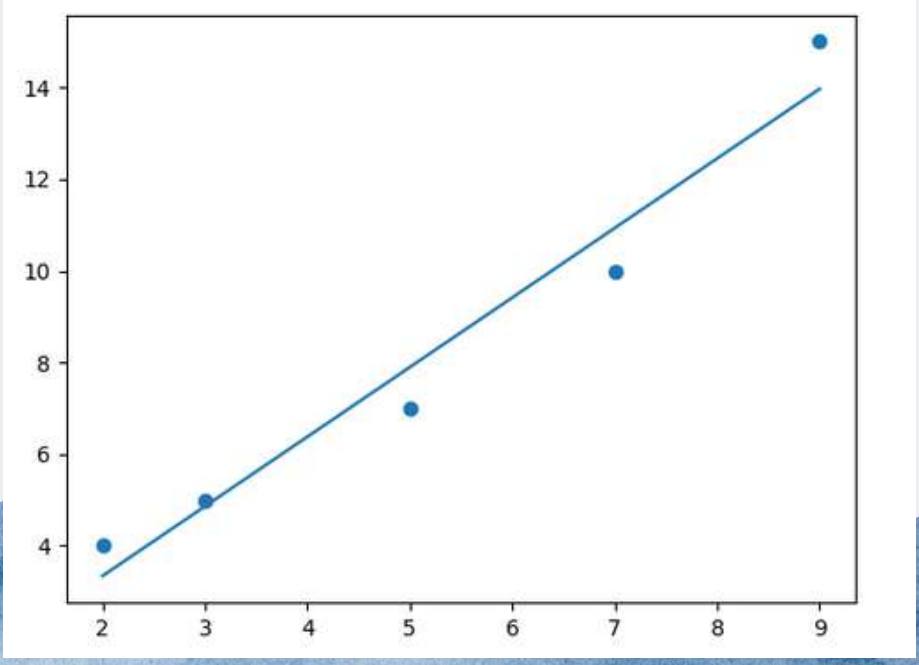
Question -2: Consider the example below where someone named Sam found how many hours of sunshine vs how many ice creams were sold at his shop from Monday to Friday: 0.5 Marks

Hours of sunshine	2	3	5	7	9
Ice cream sold	4	5	7	10	15

i. Use the least square method to determine the equation of line of best fit for the data.
ii. Plot the best fit line. and estimate if there is a 8 hours of sunshine how many ice cream would be sold tomorrow.

```
from array import array
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
a=np.array([2,3,5,7,9])
b=np.array([4,5,7,10,15])
x = np.mean(a)
y = np.mean(b)
n= len(a)
numer = 0
denom = 0
```

```
for i in range(n):
    numer += (a[i] - x) * (b[i] - y)
    denom += (a[i] - x) ** 2
m = numer / denom
c = b - (m * x)
print(m)
c,d = np.polyfit(a,b,1)
plt.scatter(a,b)
plt.plot(a,a*c+d)
plt.show()
```

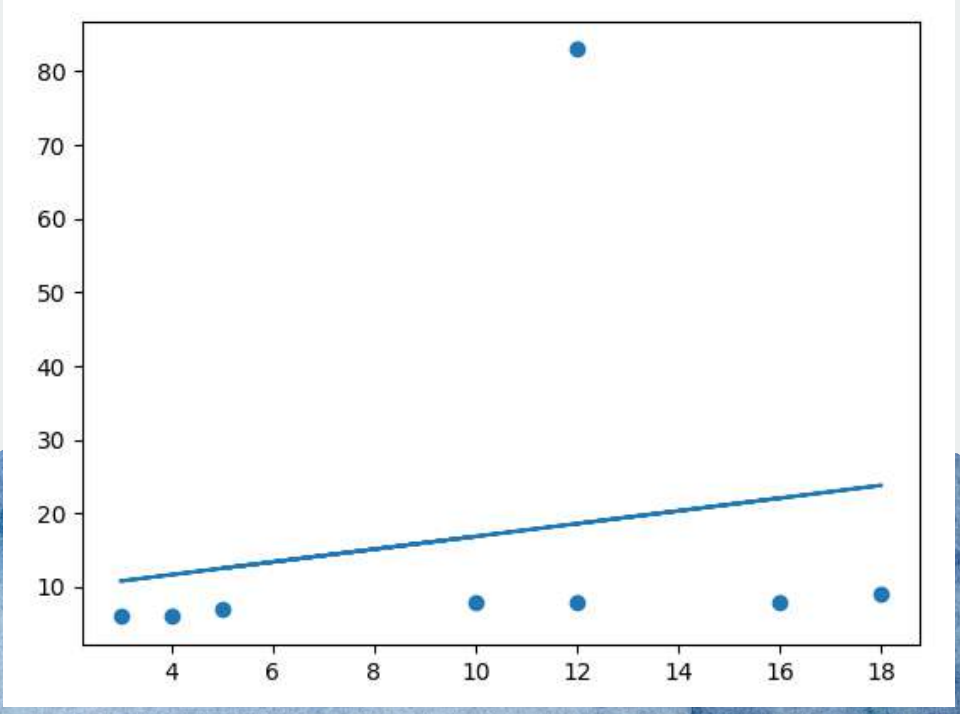


Question -3: Consider the following example: working years of faculty and their performance rating is given out of 10. **0.5 Marks**

Exp.	16	12	18	4	3	10	5	12
Performance	8	8	9	6	6	8	7	83

i. Find out the slope and intercept by least square method.
ii. Estimate the performance for a faculty with 25 years of experience.

```
import numpy as np
import matplotlib.pyplot as plt
exp=np.array([16,12,18,4,3,10,5,12])
performance=np.array([8,8,9,6,6,8,7,83])
c,d=np.polyfit(exp,performance,1)
plt.scatter(exp,performance)
plt.plot(exp,exp*c+d)
plt.show()
s=c*25+d
print(s)
```

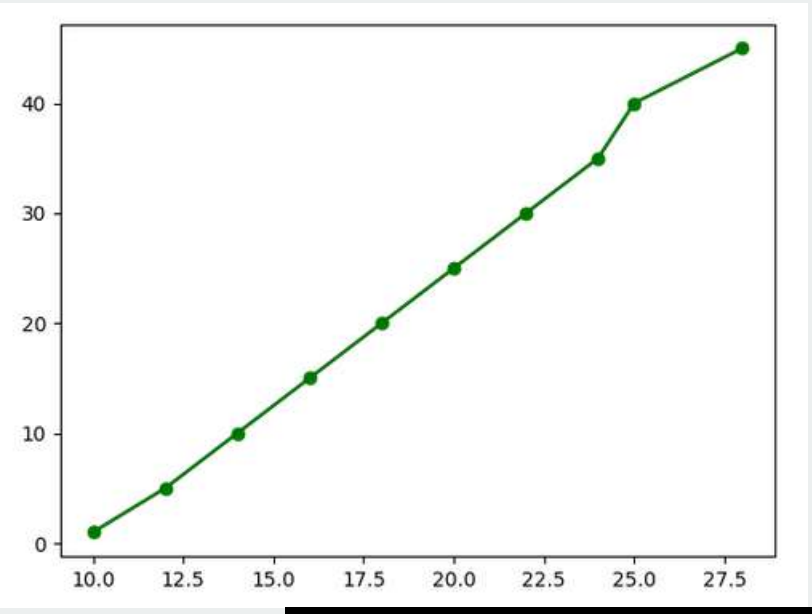


Question -4: Consider the following 10 randomly generated data point pairs. **0.5 Marks**

X	10	12	14	16	18	20	22	24	25	28
Y	1	5	10	15	20	25	30	35	40	45

i. Write a python program using Scikit-learn module to find out the Intercept and Slope by Least Square regression.

ii. Plot the Data and Regression Line




```
-0.000302
```

```python
import numpy as np
from sklearn.linear_model import LinearRegression
import matplotlib.pyplot as plt
a = np.array([10, 12, 14, 16, 18, 20, 22, 24, 25, 28])
b = np.array([1, 5, 10, 15, 20, 25, 30, 35, 40, 45])
c2 = []
d2= []
n = 10
c = np.mean(a)
d = np.mean(b)
for i in range(len(a)):
    c2.append(a[i])
for j in range(len(b)):
    d2.append(b[j])
print(-(c+d/2)*.00001)
plt.plot(a,b,'black')
SS_xy = np.sum(b*a) - n*d*c
SS_xx = np.sum(a*a) - n*c*c
plt.plot(a,b,'g')
plt.scatter(c2,d2, color = "green")
plt.show()
```