In [59]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sn
from sklearn import linear_model
```

In [143…
```
df = pd.read_csv("Dataset4.csv")
df
```

Out[143…

	experience	test_score(out of 10)	interview_score(out of 10)	salary($)
0	NaN	8.0	9	50000
1	NaN	8.0	6	45000
2	five	6.0	7	60000
3	two	10.0	10	65000
4	seven	9.0	6	70000
5	three	7.0	10	62000
6	ten	NaN	7	72000
7	eleven	7.0	8	80000

In [157…
```
df['test_score(out of 10)'].median()
```

Out[157… 8.0

In [168…
```
df['test_score(out of 10)'].fillna(df['test_score(out of 10)'].median())
```

Out[168…
```
0     8.0
1     8.0
2     6.0
3    10.0
4     9.0
5     7.0
6     8.0
7     7.0
Name: test_score(out of 10), dtype: float64
```

In [170…
```
df
```

Out[170…

	experience	test_score(out of 10)	interview_score(out of 10)	salary($)
0	zero	8.0	9	50000
1	zero	8.0	6	45000
2	five	6.0	7	60000
3	two	10.0	10	65000
4	seven	9.0	6	70000
5	three	7.0	10	62000
6	ten	8.0	7	72000
7	eleven	7.0	8	80000

In [172…
```
df['experience'] = df['experience'].fillna('zero')
```

In [174…
```
df
```

Out[174…

	experience	test_score(out of 10)	interview_score(out of 10)	salary($)
0	zero	8.0	9	50000
1	zero	8.0	6	45000
2	five	6.0	7	60000
3	two	10.0	10	65000
4	seven	9.0	6	70000
5	three	7.0	10	62000
6	ten	8.0	7	72000
7	eleven	7.0	8	80000

In [176…
```
!pip install word2number
# i used this package to change word number to integer numbers ( example: " two " will bw printed as 2)
from word2number import w2n
```

```
Requirement already satisfied: word2number in c:\users\mypc\anaconda3\lib\site-packages (1.1)
```

In [178…
```
def convert_word_to_number(word_number):
    try:
        return w2n.word_to_num(word_number)
    except ValueError:
        return None
df['experience'] = df['experience'].apply(convert_word_to_number)
```

In [180…
```
df
```

Out[180…

	experience	test_score(out of 10)	interview_score(out of 10)	salary($)
0	0	8.0	9	50000
1	0	8.0	6	45000
2	5	6.0	7	60000
3	2	10.0	10	65000
4	7	9.0	6	70000
5	3	7.0	10	62000
6	10	8.0	7	72000
7	11	7.0	8	80000

In [182…
```
df.isna().sum()
```

Out[182…
```
experience                    0
test_score(out of 10)         0
interview_score(out of 10)    0
salary($)                     0
dtype: int64
```

In [184…
```
reg = linear_model.LinearRegression()
reg.fit(df.drop(columns=['salary($)']), df['salary($)'])
```

Out[184…
```
LinearRegression()
```

In [186…
```
reg.coef_
```

Out[186… array([2812.95487627, 1845.70596798, 2205.24017467])

In [188…
```
reg.intercept_
```

Out[188… 17737.26346433768

In [190…
```
reg.predict([[2,9,6]])
```

```
C:\Users\MyPc\anaconda3\Lib\site-packages\sklearn\base.py:493: UserWarning: X does not have valid feature names, but LinearRegression was fitted with feature names
  warnings.warn(
```

Out[190… array([53205.96797671])

In [192…
```
reg.predict([[12,10,10]])
```

```
C:\Users\MyPc\anaconda3\Lib\site-packages\sklearn\base.py:493: UserWarning: X does not have valid feature names, but LinearRegression was fitted with feature names
  warnings.warn(
```

Out[192… array([92002.18340611])

In [194…
```
2812.95487627*12 + 1845.70596798 * 10 + 2205.24017467 * 10 +17737.26346433768
```

Out[194… 92002.18340607767

In [ ]: