HR's Hiring_Salary_Predictor(PROJECT)

```
# THIS PROJECT HELPS HR TO PREDICT THE SALARY OF NEWLY HIRED
PERSON
```

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn import linear_model
import warnings
warnings.filterwarnings("ignore", category=UserWarning)
warnings.simplefilter(action='ignore',category=FutureWarning)
```

In [2]:
```
df=pd.read_csv("hiring.csv")
```

In [3]:
```
df
```

Out[3]:

	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 [4]:
```
import math
df.experience=df.experience.fillna('zero')
test_medians=math.floor(df['test_score(out of 10)'].median())
df['test_score(out of 10)']=df['test_score(out of 10)'].fillna(test
```

In [6]:
```
from word2number import w2n
df['experience'] = df['experience'].apply(w2n.word_to_num)
df
```

Out[6]:

	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 [7]:
```
reg=linear_model.LinearRegression()
reg.fit(df[['experience','test_score(out of 10)','interview_score(o
```

Out[7]:
```
LinearRegression
LinearRegression()
```

In [8]:
```
reg.predict([[2,9,6]])
```

Out[8]:
```
array([53205.96797671])
```

In [9]:
```
reg.predict([[12,10,10]])
```

Out[9]:
```
array([92002.18340611])
```

In [10]:
```
import pickle
# save model
with open('linearRegressionModel_Salary_Prediction','wb') as f:
    pickle.dump(reg,f)
```

In [11]:
```
# save model
with open('linearRegressionModel_Salary_Prediction','rb') as f:
    load_model=pickle.load(f)
```

In [14]:
```
pred=load_model.predict([[2, 10, 4]]) # (experience, test score, in
print(pred)
```

```
[50641.19359534]
```