In [1]:
```
!pip install pandas
```

```
Requirement already satisfied: pandas in c:\users\megha\anaconda3\lib\site-packages (2.1.4)
Requirement already satisfied: numpy<2,>=1.23.2 in c:\users\megha\anaconda3\lib\site-packages (from pandas) (1.26.4)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\megha\anaconda3\lib\site-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in c:\users\megha\anaconda3\lib\site-packages (from pandas) (2023.3.post1)
Requirement already satisfied: tzdata>=2022.1 in c:\users\megha\anaconda3\lib\site-packages (from pandas) (2023.3)
Requirement already satisfied: six>=1.5 in c:\users\megha\anaconda3\lib\site-packages (from python-dateutil>=2.8.2->pandas) (1.16.0)
```

In [2]:
```
import pandas
print(pandas.__version__)
```

```
2.1.4
```

In [4]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [6]:
```
plt.rcParams['figure.figsize']=(19,8)
```

In [11]:
```
import warnings
warnings.filterwarnings('ignore')
```

In [7]:
```
import pandas as pd
salary_df=pd.read_csv("C:\\Users\\megha\\OneDrive\\Documents\\Salary.csv")
df=pd.DataFrame(salary_df)
df
```

Out[7]:

	YearsExperience	Salary
0	1.1	39343
1	1.3	46205
2	1.5	37731
3	2.0	43525
4	2.2	39891
5	2.9	56642
6	3.0	60150
7	3.2	54445
8	3.2	64445
9	3.7	57189
10	3.9	63218
11	4.0	55794
12	4.0	56957
13	4.1	57081
14	4.5	61111
15	4.9	67938
16	5.1	66029
17	5.3	83088
18	5.9	81363
19	6.0	93940
20	6.8	91738
21	7.1	98273
22	7.9	101302
23	8.2	113812
24	8.7	109431
25	9.0	105582
26	9.5	116969
27	9.6	112635
28	10.3	122391
29	10.5	121872
30	11.2	127345
31	11.5	126756
32	12.3	128765
33	12.9	135675
34	13.5	139465

In [8]:
```
salary_df.shape
```

Out[8]:
```
(35, 2)
```

In [9]:
```
salary_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 35 entries, 0 to 34
Data columns (total 2 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   YearsExperience  35 non-null     float64
 1   Salary           35 non-null     int64  
dtypes: float64(1), int64(1)
memory usage: 692.0 bytes
```

In [10]:
```
salary_df.head()
```

Out[10]:

	YearsExperience	Salary
0	1.1	39343
1	1.3	46205
2	1.5	37731
3	2.0	43525
4	2.2	39891

In [11]:
```
salary_df.describe()
```

Out[11]:

	YearsExperience	Salary
count	35.000000	35.000000
mean	6.308571	83945.600000
std	3.618610	32162.673003
min	1.100000	37731.000000
25%	3.450000	57019.000000
50%	5.300000	81363.000000
75%	9.250000	113223.500000
max	13.500000	139465.000000

In [13]:
```
import matplotlib.pyplot as plt
plt.scatter(data=salary_df, x='YearsExperience', y='Salary', s=200)
plt.title("Salary based on the years of experience")
plt.xlabel("Years of Experience")
plt.ylabel("Salary")
plt.show()
```

In [17]:
```
x=salary_df.loc[:,'YearsExperience'].values
y=salary_df.loc[:,'Salary'].values
```

In [18]:
```
from sklearn.model_selection import train_test_split
```

In [19]:
```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=0)
```

In [20]:
```
x_train.shape,x_test.shape,y_train.shape,y_test.shape
```

Out[20]:
```
((28,), (7,), (28,), (7,))
```

In [21]:
```
x_train.reshape(-1,1).shape
```

Out[21]:
```
(28, 1)
```

In [22]:
```
from sklearn.linear_model import LinearRegression
```

In [23]:
```
reg_model=LinearRegression()
```

In [24]:
```
reg_model.fit(x_train.reshape(-1,1),y_train.reshape(-1,1))
```

Out[24]:
```
LinearRegression()
```

In [25]:
```
reg_model.coef_
```

Out[25]:
```
array([[8629.79240044]])
```

In [26]:
```
reg_model.intercept_
```

Out[26]:
```
array([29945.05652018])
```

In [27]:
```
reg_model.score(x_train.reshape(-1,1),y_train.reshape(-1,1))
```

Out[27]:
```
0.9638371903672509
```

In [28]:
```
y_predicted=reg_model.predict(x_test.reshape(-1,1))
```

In [29]:
```
y_predicted
```

Out[29]:
```
array([[120057.87672477],
       [ 88127.64484315],
       [ 73456.99776241],
       [118331.91824468],
       [ 97620.41648363],
       [ 71731.03928232],
       [ 63101.24688189]])
```

In [30]:
```
y_test
```

Out[30]:
```
array([121872,  91738,  66029, 122391, 101302,  67938,  63218],
      dtype=int64)
```

In [32]:
```
from sklearn.metrics import r2_score
```

In [33]:
```
r_square=r2_score(y_test.reshape(-1,1),y_predicted)
```

In [34]:
```
r_square
```

Out[34]:
```
0.9708090245443415
```

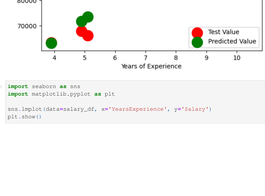
In [35]:
```
print(f"accuracy={r_square:.2%}")
```

```
accuracy=97.08%
```

In [36]:
```
reg_model.score(x_test.reshape(-1,1),y_test.reshape(-1,1))
```

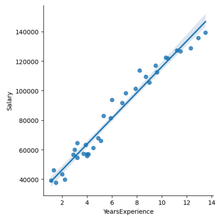
Out[36]:
```
0.9708090245443415
```

In [38]:
```
plt.scatter(x=x_test,y=y_test,color='red',s=300)
plt.scatter(x=x_test,y=y_predicted,color='green',s=300)
plt.title("Salary Test Data Vs Predicted")
plt.xlabel("Years of Experience")
plt.ylabel("Salary")
plt.legend(["Test Value","Predicted Value"],loc="lower right")
plt.show()
```



In [40]:
```
import seaborn as sns
import matplotlib.pyplot as plt

sns.lmplot(data=salary_df, x='YearsExperience', y='Salary')
plt.show()
```


Salary
YearsExperience
140000
120000
100000
80000
60000
40000
2
4
6
8
10
12
14

In [ ]: