```
#importing the dataset
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

dataset = pd.read_csv("Salary_Data.csv")
dataset.head()
```

| | YearsExperience | Salary |
|---|-----------------|---------|
| 0 | 1.1 | 39343.0 |
| 1 | 1.3 | 46205.0 |
| 2 | 1.5 | 37731.0 |
| 3 | 2.0 | 43525.0 |
| 4 | 2.2 | 39891.0 |

```
#data preprocessing
X= dataset.iloc[:,:-1].values #independent variable array
y = dataset.iloc[:,1].values #dependent varible vector
# splitting the dataset
from sklearn.model selection import train test split
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=1/3,random_state=0)
# fitting the regression model
from sklearn.linear model import LinearRegression
regressor = LinearRegression()
regressor.fit(X_train,y_train) #actually produces the linear eqn for the data
     LinearRegression()
# predicting the test set results
y_pred = regressor.predict(X_test)
y_pred
y_test
     array([ 37731., 122391., 57081., 63218., 116969., 109431., 112635.,
             55794., 83088., 101302.])
X test
     array([[ 1.5],
```

```
[10.3],
[ 4.1],
[ 3.9],
[ 9.5],
[ 8.7],
[ 9.6],
[ 4. ],
[ 5.3],
[ 7.9]])
```

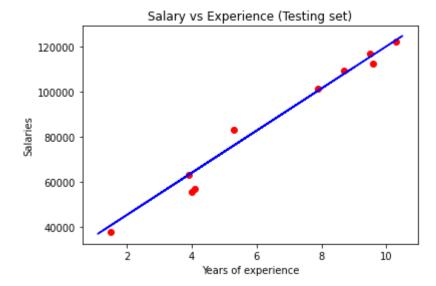
plt.show() # specifies end of graph

```
# visualizing the results
#plot for the TRAIN
plt.scatter(X_train, y_train, color='red') # plotting the observation line
plt.plot(X_train, regressor.predict(X_train), color='blue') # plotting the regression line
plt.title("Salary vs Experience (Training set)") # stating the title of the graph

plt.xlabel("Years of experience") # adding the name of x-axis
plt.ylabel("Salaries") # adding the name of y-axis
```



```
#plot for the TEST
plt.scatter(X_test, y_test, color='red')
plt.plot(X_train, regressor.predict(X_train), color='blue') # plotting the regression line
plt.title("Salary vs Experience (Testing set)")
plt.xlabel("Years of experience")
plt.ylabel("Salaries")
plt.show()
```



✓ 0s completed at 19:06

×