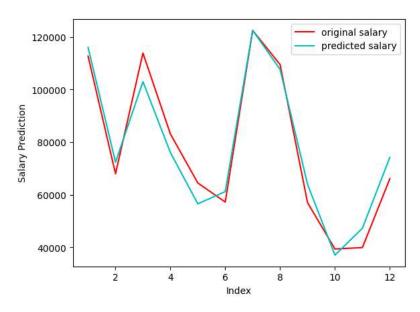
LinearRegression()

y\_pred=reg.predict(X\_test)

print(f"Intercept {reg.intercept\_}")
print(f"Coefficient {reg.coef\_}")

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2 score
import matplotlib.pyplot as plt
df=pd.read_csv('Salary_Data.csv')
df.head()
        YearsExperience Salary
      0
                     1.1 39343.0
                     1.3 46205.0
      1
      2
                     1.5 37731.0
                     2.0 43525.0
      3
                     2.2 39891.0
df.shape
     (30, 2)
df.isnull().sum()
     YearsExperience
                        0
     Salary
                        0
     dtype: int64
df.describe()
             YearsExperience
                                    Salary
                                  30.000000
      count
                   30.000000
                              76003.000000
      mean
                    5.313333
       std
                    2.837888
                              27414.429785
                              37731.000000
                    1.100000
      min
      25%
                    3.200000
                               56720.750000
      50%
                    4.700000
                              65237.000000
      75%
                    7.700000
                             100544.750000
                   10.500000 122391.000000
      max
X = df.drop(['Salary'], axis=1)
y = df['Salary']
X_train, X_test, y_train, y_test = train_test_split(X, y, train_size=0.6, random_state=42)
reg=LinearRegression()
reg.fit(X_train, y_train)
     ▼ LinearRegression
```

```
Intercept 26806.601413653698
     Coefficient [9284.87835182]
rscore=r2_score(y_test, y_pred)
print(rscore)
     0.9532338376280692
c = [i for i in range (1,len(y_test)+1, 1)]
     [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]
y_test
           112635.0
     27
            67938.0
     15
     23
           113812.0
     17
            83088.0
     8
            64445.0
     9
            57189.0
     28
           122391.0
     24
           109431.0
     12
            56957.0
            39343.0
     0
            39891.0
     4
     16
            66029.0
     Name: Salary, dtype: float64
plt.plot(c, y_test, color="r")
plt.plot(c, y_pred, color="c")
plt.xlabel("Index")
plt.ylabel("Salary Prediction")
plt.legend(['original salary', 'predicted salary'])
plt.show()
```



• ×