```
import numpy as np

import pandas as pd

import matplotlib.pyplot as plt


# Load the dataset

dataset = pd.read_csv(r'D:\Samsom - All Data\Naresh IT Institute\New
folder\Salary_Data.csv')


# Check the shape of the dataset

print("Dataset Shape:", dataset.shape) # (30, 2)


# Feature selection (independent variable x and dependent variable)

x = dataset.iloc[:, :-1] # Years of experience (Independent Variable)

y = dataset.iloc[:, -1] # Salary (Dependent variable)


# Split the dataset into training and testing sets (80% training)

from sklearn.model_selection import train_test_split

x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.2, random_state=0)


# Reshape x_train and x_test into 2D arrays if they are single

x_train = x_train.values.reshape(-1, 1)

x_test = x_test.values.reshape(-1, 1)


# Predicting the results for the test set

from sklearn.linear_model import LinearRegression

from sklearn.model_selection import train_test_split

regressor = LinearRegression()

regressor.fit(x_train, y_train)
```

```
y_pred = regressor.predict(x_test)

# Compare predicted and actual salaries from the test set
comparison = pd.DataFrame({'Actual': y_test, 'Predicted': y_pred})
print(comparison)

# Visualizing the Training set results
plt.scatter(x_test, y_test, color = 'red') # Real salary
plt.plot(x_train, regressor.predict(x_train), color = 'blue')
plt.title('Salary vs Experience (Training set)')
plt.xlabel('Years of Experience')
plt.ylabel('Salary')
plt.show()

m_slope = regressor.coef_
print(m_slope)

c_intercept = regressor.intercept_
print(c_intercept)

y_12 = m_slope*12+c_intercept
print(y_12)
```