1- Simple Linear Regression

Step-1: Import Dataset

In [ ]:

```
import pandas as pd
df = pd.read_csv("ml_data_salary.csv")
```

In [ ]:

```
df.head()
```

Out[ ]:

	age	distance	YearsExperience	Salary
0	31.1	77.75	1.1	39343
1	31.3	78.25	1.3	46205
2	31.5	78.75	1.5	37731
3	32.0	80.00	2.0	43525
4	32.2	80.50	2.2	39891

In [ ]:

```
df1 = df.iloc[:, 2:4]
df1.head()
```

Out[ ]:

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

In [ ]:

```
df1.to_csv("ml_data_salary_new.csv")
```

Step-2: Splitting Dataset into Training data and Testing data

In [ ]:

```
x = df[["YearsExperience"]]
y = df["Salary"]
```

In [ ]:

```
x.head()
```

Out[ ]:

	YearsExperience
0	1.1
1	1.3
2	1.5
3	2.0
4	2.2

In [ ]:

```
y.head()
```

Out[ ]:

```
0    39343
1    46205
2    37731
3    43525
4    39891
Name: Salary, dtype: int64
```

In [ ]:

```
# import llibrary and Split Data
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=
```

Step-3 Fit Linear Regression Model

In [ ]:

```
from sklearn.linear_model import LinearRegression
model=LinearRegression().fit(x_train, y_train)
model
```

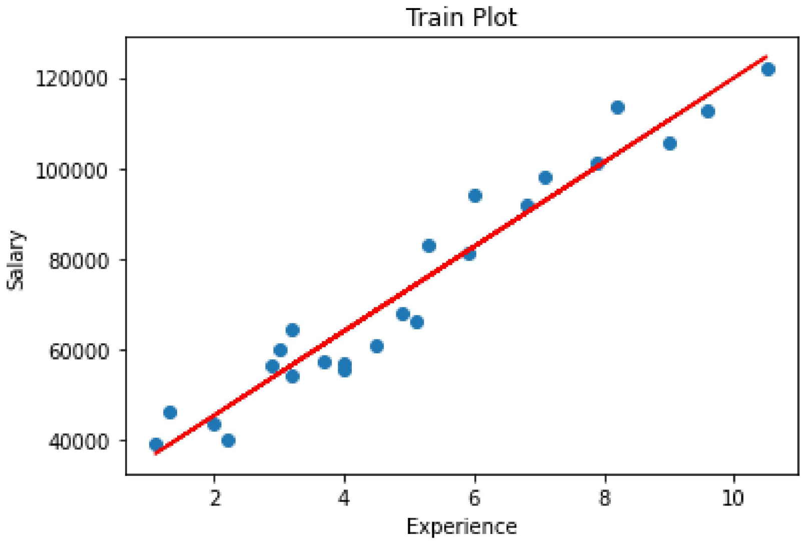
Out[ ]:

```
▾ LinearRegression
LinearRegression()
```

Step-4 Plotting

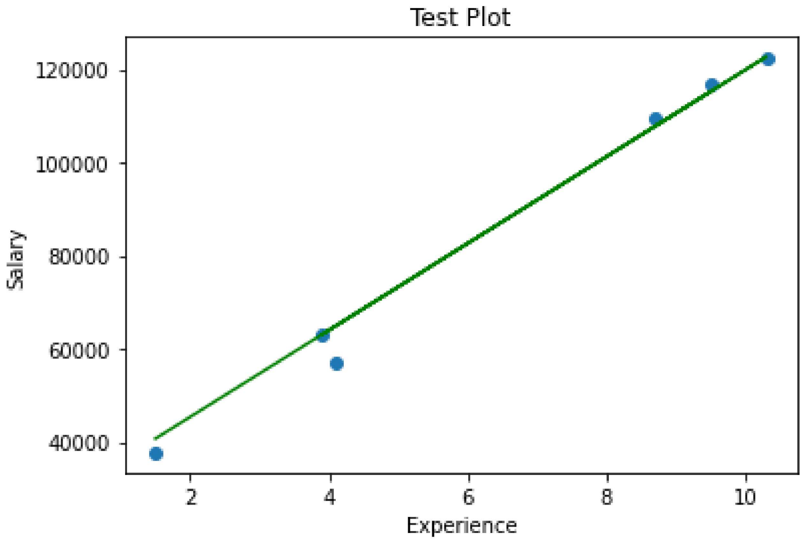
In [ ]:

```
import matplotlib.pyplot as plt
plt.scatter(x_train, y_train)
plt.plot(x_train, model.predict(x_train), color="red")
plt.xlabel("Experience")
plt.ylabel("Salary")
plt.title("Train Plot")
plt.show()
```



In [ ]:

```
import matplotlib.pyplot as plt
plt.scatter(x_test, y_test)
plt.plot(x_test, model.predict(x_test), color="green")
plt.xlabel("Experience")
plt.ylabel("Salary")
plt.title("Test Plot")
plt.show()
```



Step-5 Testing or evaluating your model

In [ ]:

```
# Model fitness score of testing Data Model
model.score(x_test,y_test)
```

Out[ ]:

```
0.988169515729126
```

In [ ]:

```
# Model fitness score of training Data Model
model.score(x_train,y_train)
```

Out[ ]:

```
0.9411949620562126
```

Step-6 Pridiction of Unknown Values

Salary pridiction with 5 years experience.

```
In [ ]: model.predict([[5]])
```

```
c:\Users\Dell\miniconda3\envs\streamlit_env\lib\site-packages\sklearn\base.py:450: Us
erWarning: X does not have valid feature names, but LinearRegression was fitted with
feature names
  warnings.warn(
```

```
Out[ ]: array([73342.97478427])
```

Salary pridiction with multiple values of years experience.

```
In [ ]: model.predict([[5],[10],[15],[25]])
```

```
c:\Users\Dell\miniconda3\envs\streamlit_env\lib\site-packages\sklearn\base.py:450: Us
erWarning: X does not have valid feature names, but LinearRegression was fitted with
feature names
  warnings.warn(
```

```
Out[ ]: array([ 73342.97478427, 119905.85041792, 166468.72605157, 259594.47731886])
```

Values from from test data

```
In [ ]: model.predict(x_test)
```

```
Out[ ]: array([ 40748.96184072, 122699.62295594,  64961.65717022,  63099.14214487,
               115249.56285456, 107799.50275317])
```

```
In [ ]: x = ([10],[20],[30],[40])
        model.predict(x)
```

```
c:\Users\Dell\miniconda3\envs\streamlit_env\lib\site-packages\sklearn\base.py:450: Us
erWarning: X does not have valid feature names, but LinearRegression was fitted with
feature names
  warnings.warn(
```

```
Out[ ]: array([119905.85041792, 213031.60168521, 306157.3529525 , 399283.1042198 ])
```