In [1]:

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

In [2]:

```
#importing Dataset
df = pd.read_csv("50_Startups.csv")
```

In [3]:

```
#View The Data
df.head()
```

Out[3]:

	R&D Spend	Administration	Marketing Spend	State	Profit
0	165349.20	136897.80	471784.10	New York	192261.83
1	162597.70	151377.59	443898.53	California	191792.06
2	153441.51	101145.55	407934.54	Florida	191050.39
3	144372.41	118671.85	383199.62	New York	182901.99
4	142107.34	91391.77	366168.42	Florida	166187.94

In [4]:

```
#View The Data Info
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 50 entries, 0 to 49
Data columns (total 5 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   R&D Spend        50 non-null     float64
 1   Administration   50 non-null     float64
 2   Marketing Spend  50 non-null     float64
 3   State            50 non-null     object 
 4   Profit           50 non-null     float64
dtypes: float64(4), object(1)
memory usage: 2.1+ KB
```

In [5]:

```
#View The Shape of Data
df.shape
```

Out[5]:

```
(50, 5)
```

In [6]:

```
#Check if There is Any NULl Values in Data
df.isnull().sum()
```

Out[6]:

```
R&D Spend          0
Administration     0
Marketing Spend    0
State              0
Profit             0
dtype: int64
```

In [7]:
```
#Defining Features & Label of Data
X = df.iloc[:, :-1]
y = df.iloc[:, 4]
```

In [8]:
```
X
```

Out[8]:

	R&D Spend	Administration	Marketing Spend	State
0	165349.20	136897.80	471784.10	New York
1	162597.70	151377.59	443898.53	California
2	153441.51	101145.55	407934.54	Florida
3	144372.41	118671.85	383199.62	New York
4	142107.34	91391.77	366168.42	Florida
5	131876.90	99814.71	362861.36	New York
6	134615.46	147198.87	127716.82	California
7	130298.13	145530.06	323876.68	Florida
8	120542.52	148718.95	311613.29	New York
9	123334.88	108679.17	304981.62	California
10	101913.08	110594.11	229160.95	Florida

In [9]:
```
y
```

Out[9]:
```
0     192261.83
1     191792.06
2     191050.39
3     182901.99
4     166187.94
5     156991.12
6     156122.51
7     155752.60
8     152211.77
9     149759.96
10    146121.95
11    144259.40
12    141585.52
13    134307.35
14    132602.65
15    129917.04
16    126992.93
17    125370.37
18    124266.90
19    122776.86
```

In [10]:
```
df['State'].unique()
```

Out[10]:
```
array(['New York', 'California', 'Florida'], dtype=object)
```

In [11]:

```
#Encoding Categorical Data
#Encoding The Independent Variable
#Import LabelEncoder Scikit-Learn Library to Handle the Categorical Data

from sklearn.preprocessing import LabelEncoder

LE = LabelEncoder()
X.iloc[:, 3] = LE.fit_transform(X.iloc[:, 3])
```

In [12]:

```
X
```

Out[12]:

	R&D Spend	Administration	Marketing Spend	State
0	165349.20	136897.80	471784.10	2
1	162597.70	151377.59	443898.53	0
2	153441.51	101145.55	407934.54	1
3	144372.41	118671.85	383199.62	2
4	142107.34	91391.77	366168.42	1
5	131876.90	99814.71	362861.36	2
6	134615.46	147198.87	127716.82	0
7	130298.13	145530.06	323876.68	1
8	120542.52	148718.95	311613.29	2
9	123334.88	108679.17	304981.62	0
10	101913.08	110594.11	229160.95	1
11	100671.96	91790.61	249744.55	0
12	93863.75	127320.38	249839.44	1
13	91992.39	135495.07	252664.93	0
14	119943.24	156547.42	256512.92	1
15	114523.61	122616.84	261776.23	2
16	78013.11	121597.55	264346.06	0
17	94657.16	145077.58	282574.31	2
18	91749.16	114175.79	294919.57	1
19	86419.70	153514.11	0.00	2
20	76253.86	113867.30	298664.47	0
21	78389.47	153773.43	299737.29	2
22	73994.56	122782.75	303319.26	1
23	67532.53	105751.03	304768.73	1
24	77044.01	99281.34	140574.81	2
25	64664.71	139553.16	137962.62	0
26	75328.87	144135.98	134050.07	1
27	72107.60	127864.55	353183.81	2
28	66051.52	182645.56	118148.20	1
29	65605.48	153032.06	107138.38	2
30	61994.48	115641.28	91131.24	1
31	61136.38	152701.92	88218.23	2
32	63408.86	129219.61	46085.25	0
33	55493.95	103057.49	214634.81	1
34	46426.07	157693.92	210797.67	0
35	46014.02	85047.44	205517.64	2
36	28663.76	127056.21	201126.82	1
37	44069.95	51283.14	197029.42	0

	R&D Spend	Administration	Marketing Spend	State
38	20229.59	65947.93	185265.10	2
39	38558.51	82982.09	174999.30	0
40	28754.33	118546.05	172795.67	0
41	27892.92	84710.77	164470.71	1
42	23640.93	96189.63	148001.11	0
43	15505.73	127382.30	35534.17	2
44	22177.74	154806.14	28334.72	0
45	1000.23	124153.04	1903.93	2
46	1315.46	115816.21	297114.46	1
47	0.00	135426.92	0.00	0
48	542.05	51743.15	0.00	2
49	0.00	116983.80	45173.06	0

In [13]:

```
#Import OneHotEncoder Scikit-Learn Library to Handle the Categorical Data

from sklearn.preprocessing import OneHotEncoder

OHE = OneHotEncoder(categories = 'auto', sparse_output = False, drop = 'first')
X["State"] = OHE.fit_transform(X[["State"]])
```

In [14]:

```
X
```

Out[14]:

	R&D Spend	Administration	Marketing Spend	State
0	165349.20	136897.80	471784.10	0.0
1	162597.70	151377.59	443898.53	0.0
2	153441.51	101145.55	407934.54	1.0
3	144372.41	118671.85	383199.62	0.0
4	142107.34	91391.77	366168.42	1.0
5	131876.90	99814.71	362861.36	0.0
6	134615.46	147198.87	127716.82	0.0
7	130298.13	145530.06	323876.68	1.0
8	120542.52	148718.95	311613.29	0.0
9	123334.88	108679.17	304981.62	0.0
10	101913.08	110594.11	229160.95	1.0
11	100671.96	91790.61	249744.55	0.0
12	93863.75	127320.38	249839.44	1.0
13	91992.39	135495.07	252664.93	0.0
14	119943.24	156547.42	256512.92	1.0
15	114523.61	122616.84	261776.23	0.0
16	78013.11	121597.55	264346.06	0.0
17	94657.16	145077.58	282574.31	0.0
18	91749.16	114175.79	294919.57	1.0
19	86419.70	153514.11	0.00	0.0
20	76253.86	113867.30	298664.47	0.0
21	78389.47	153773.43	299737.29	0.0
22	73994.56	122782.75	303319.26	1.0
23	67532.53	105751.03	304768.73	1.0
24	77044.01	99281.34	140574.81	0.0
25	64664.71	139553.16	137962.62	0.0
26	75328.87	144135.98	134050.07	1.0
27	72107.60	127864.55	353183.81	0.0
28	66051.52	182645.56	118148.20	1.0
29	65605.48	153032.06	107138.38	0.0
30	61994.48	115641.28	91131.24	1.0
31	61136.38	152701.92	88218.23	0.0
32	63408.86	129219.61	46085.25	0.0
33	55493.95	103057.49	214634.81	1.0
34	46426.07	157693.92	210797.67	0.0
35	46014.02	85047.44	205517.64	0.0
36	28663.76	127056.21	201126.82	1.0
37	44069.95	51283.14	197029.42	0.0

	R&D Spend	Administration	Marketing Spend	State
38	20229.59	65947.93	185265.10	0.0
39	38558.51	82982.09	174999.30	0.0
40	28754.33	118546.05	172795.67	0.0
41	27892.92	84710.77	164470.71	1.0
42	23640.93	96189.63	148001.11	0.0
43	15505.73	127382.30	35534.17	0.0
44	22177.74	154806.14	28334.72	0.0
45	1000.23	124153.04	1903.93	0.0
46	1315.46	115816.21	297114.46	1.0
47	0.00	135426.92	0.00	0.0
48	542.05	51743.15	0.00	0.0
49	0.00	116983.80	45173.06	0.0

In [15]:

```
#Spliting Data into Train Test

from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=2)
```

In [16]:

```
X_train.shape
```

Out[16]:

```
(35, 4)
```

In [17]:

```
y_test.shape
```

Out[17]:

```
(15,)
```

In [18]:

```
#Import Linear Regression
from sklearn.linear_model import LinearRegression

regressor = LinearRegression()
```

In [19]:

```
#Fit Data into Linear Regression
regressor.fit(X_train, y_train)
```

Out[19]:

```
▾ LinearRegression
LinearRegression()
```

In [20]:

```
#Predicting The Test Set Results
y_pred = regressor.predict(X_test)
y_pred
```

Out[20]:

```
array([ 72429.48912957,  47181.51953034,  95756.870945  , 157311.2829052 ,
       127996.57073699, 192714.64646446,  64080.83170966,  53374.82743239,
        87619.15326602, 108532.24950067, 116684.51835304,  55577.64825096,
       129942.44900368, 126712.21201849, 114791.66860359])
```

In [21]:

```
y_test
```

Out[21]:

```
36     90708.19
47     42559.73
28    103282.38
9     149759.96
13    134307.35
0     192261.83
44     65200.33
46     49490.75
39     81005.76
23    108733.99
24    108552.04
48     35673.41
17    125370.37
12    141585.52
27    105008.31
Name: Profit, dtype: float64
```

In [22]:

```
#Accuracy SCore Of the Model
regressor.score(X_test, y_test)
```

Out[22]:

```
0.9479214681245989
```

In [ ]: