```
import pandas as pd

#dataset=pd.read_csv("50_Startups_withYear.csv")
dataset=pd.read_csv("50_Startups.csv")

dataset
```

	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
5	131876.90	99814.71	362861.36	New York	156991.12
6	134615.46	147198.87	127716.82	California	156122.51
7	130298.13	145530.06	323876.68	Florida	155752.60
8	120542.52	148718.95	311613.29	New York	152211.77
9	123334.88	108679.17	304981.62	California	149759.96
10	101913.08	110594.11	229160.95	Florida	146121.95
11	100671.96	91790.61	249744.55	California	144259.40
12	93863.75	127320.38	249839.44	Florida	141585.52
13	91992.39	135495.07	252664.93	California	134307.35
14	119943.24	156547.42	256512.92	Florida	132602.65
15	114523.61	122616.84	261776.23	New York	129917.04
16	78013.11	121597.55	264346.06	California	126992.93
17	94657.16	145077.58	282574.31	New York	125370.37
18	91749.16	114175.79	294919.57	Florida	124266.90
19	86419.70	153514.11	0.00	New York	122776.86
20	76253.86	113867.30	298664.47	California	118474.03
21	78389.47	153773.43	299737.29	New York	111313.02
22	73994.56	122782.75	303319.26	Florida	110352.25
23	67532.53	105751.03	304768.73	Florida	108733.99
24	77044.01	99281.34	140574.81	New York	108552.04
25	64664.71	139553.16	137962.62	California	107404.34
26	75328.87	144135.98	134050.07	Florida	105733.54
27	72107.60	127864.55	353183.81	New York	105008.31
28	66051.52	182645.56	118148.20	Florida	103282.38
29	65605.48	153032.06	107138.38	New York	101004.64
30	61994.48	115641.28	91131.24	Florida	99937.59
31	61136.38	152701.92	88218.23	New York	97483.56
32	63408.86	129219.61	46085.25	California	97427.84
33	55493.95	103057.49	214634.81	Florida	96778.92
34	46426.07	157693.92	210797.67	California	96712.80
35	46014.02	85047.44	205517.64	New York	96479.51
36	28663.76	127056.21	201126.82	Florida	90708.19
37	44069.95	51283.14	197029.42	California	89949.14
38	20229.59	65947.93	185265.10	New York	81229.06
39	38558.51	82982.09	174999.30	California	81005.76
40	28754.33	118546.05	172795.67	California	78239.91
41	27892.92	84710.77	164470.71	Florida	77798.83

```
42   23640.93        96189.63        148001.11  California   71498.49
43   15505.73       127382.30         35534.17    New York   69758.98
44   22177.74       154806.14         28334.72  California   65200.33
45    1000.23       124153.04          1903.93    New York   64926.08
46    1315.46       115816.21        297114.46     Florida   49490.75
47       0.00       135426.92             0.00  California   42559.73
48     542.05        51743.15             0.00    New York   35673.41
49       0.00       116983.80         45173.06  California   14681.40
```

```
dataset=pd.get_dummies(dataset,drop_first=True)
```

```
dataset=dataset.replace({True: 1, False: 0})
```

```
dataset
```

```
    R&D Spend  Administration  Marketing Spend     Profit
State_Florida  \
0   165349.20       136897.80        471784.10  192261.83
0
1   162597.70       151377.59        443898.53  191792.06
0
2   153441.51       101145.55        407934.54  191050.39
1
3   144372.41       118671.85        383199.62  182901.99
0
4   142107.34        91391.77        366168.42  166187.94
1
5   131876.90        99814.71        362861.36  156991.12
0
6   134615.46       147198.87        127716.82  156122.51
0
7   130298.13       145530.06        323876.68  155752.60
1
8   120542.52       148718.95        311613.29  152211.77
0
9   123334.88       108679.17        304981.62  149759.96
0
10  101913.08       110594.11        229160.95  146121.95
1
11  100671.96        91790.61        249744.55  144259.40
0
12   93863.75       127320.38        249839.44  141585.52
1
13   91992.39       135495.07        252664.93  134307.35
0
14  119943.24       156547.42        256512.92  132602.65
1
15  114523.61       122616.84        261776.23  129917.04
0
16   78013.11       121597.55        264346.06  126992.93
```

```
0
17   94657.16       145077.58        282574.31  125370.37
0
18   91749.16       114175.79        294919.57  124266.90
1
19   86419.70       153514.11             0.00  122776.86
0
20   76253.86       113867.30        298664.47  118474.03
0
21   78389.47       153773.43        299737.29  111313.02
0
22   73994.56       122782.75        303319.26  110352.25
1
23   67532.53       105751.03        304768.73  108733.99
1
24   77044.01        99281.34        140574.81  108552.04
0
25   64664.71       139553.16        137962.62  107404.34
0
26   75328.87       144135.98        134050.07  105733.54
1
27   72107.60       127864.55        353183.81  105008.31
0
28   66051.52       182645.56        118148.20  103282.38
1
29   65605.48       153032.06        107138.38  101004.64
0
30   61994.48       115641.28         91131.24   99937.59
1
31   61136.38       152701.92         88218.23   97483.56
0
32   63408.86       129219.61         46085.25   97427.84
0
33   55493.95       103057.49        214634.81   96778.92
1
34   46426.07       157693.92        210797.67   96712.80
0
35   46014.02        85047.44        205517.64   96479.51
0
36   28663.76       127056.21        201126.82   90708.19
1
37   44069.95        51283.14        197029.42   89949.14
0
38   20229.59        65947.93        185265.10   81229.06
0
39   38558.51        82982.09        174999.30   81005.76
0
40   28754.33       118546.05        172795.67   78239.91
0
```

```
41   27892.92        84710.77        164470.71   77798.83
1
42   23640.93        96189.63        148001.11   71498.49
0
43   15505.73       127382.30         35534.17   69758.98
0
44   22177.74       154806.14         28334.72   65200.33
0
45    1000.23       124153.04          1903.93   64926.08
0
46    1315.46       115816.21        297114.46   49490.75
1
47       0.00       135426.92             0.00   42559.73
0
48     542.05        51743.15             0.00   35673.41
0
49       0.00       116983.80         45173.06   14681.40
0
```

```
    State_New York
0                1
1                0
2                0
3                1
4                0
5                1
6                0
7                0
8                1
9                0
10               0
11               0
12               0
13               0
14               0
15               1
16               0
17               1
18               0
19               1
20               0
21               1
22               0
23               0
24               1
25               0
26               0
27               1
28               0
```

```
29                1
30                0
31                1
32                0
33                0
34                0
35                1
36                0
37                0
38                1
39                0
40                0
41                0
42                0
43                1
44                0
45                1
46                0
47                0
48                1
49                0
```

```
dataset.columns
```

```
Index(['R&D Spend', 'Administration', 'Marketing Spend', 'Profit',
       'State_Florida', 'State_New York'],
      dtype='object')
```

```
independent=dataset[['R&D Spend', 'Administration', 'Marketing
Spend','State_Florida', 'State_New York']]
```

```
dependent=dataset[['Profit']]
```

```
independent
```

```
    R&D Spend  Administration  Marketing Spend  State_Florida
State_New York
0   165349.20       136897.80        471784.10               0
1
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
1
4   142107.34        91391.77        366168.42               1
0
5   131876.90        99814.71        362861.36               0
1
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	1
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	1
16	78013.11	121597.55	264346.06	0	0
17	94657.16	145077.58	282574.31	0	1
18	91749.16	114175.79	294919.57	1	0
19	86419.70	153514.11	0.00	0	1
20	76253.86	113867.30	298664.47	0	0
21	78389.47	153773.43	299737.29	0	1
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	1
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	1
28	66051.52	182645.56	118148.20	1	0
29	65605.48	153032.06	107138.38	0	1
30	61994.48	115641.28	91131.24	1	0
31	61136.38	152701.92	88218.23	0	

```
1
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
1
36   28663.76       127056.21        201126.82               1
0
37   44069.95        51283.14        197029.42               0
0
38   20229.59        65947.93        185265.10               0
1
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
1
44   22177.74       154806.14         28334.72               0
0
45    1000.23       124153.04          1903.93               0
1
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
1
49       0.00       116983.80         45173.06               0
0
```

dependent

```
       Profit
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
20  118474.03
21  111313.02
22  110352.25
23  108733.99
24  108552.04
25  107404.34
26  105733.54
27  105008.31
28  103282.38
29  101004.64
30   99937.59
31   97483.56
32   97427.84
33   96778.92
34   96712.80
35   96479.51
36   90708.19
37   89949.14
38   81229.06
39   81005.76
40   78239.91
41   77798.83
42   71498.49
43   69758.98
44   65200.33
45   64926.08
46   49490.75
47   42559.73
48   35673.41
49   14681.40
```

```
#Training code
#Now we need to split the Training and Test data set from parant
dataset. To split, we are using sklearn as class and call function as
model_selection-> train_test_split
from sklearn.model_selection import train_test_split
#input parameters are passed in function train_test_split with
x,y,size for test data(30 percent from x & y) and random state then
assign to variable x_train,x_test,y_train and y_test
```

```
x_train,x_test,y_train,y_test=train_test_split(independent, dependent,
test_size=0.30, random_state=0)

#Model Creation by using linear regression algorithm, here importing
LinearRegression algorithm from sklearn.liner_model
from sklearn.linear_model import LinearRegression
#assign the LinearRegression fucntion without parameters in the
variable regressor
regressor=LinearRegression()
#calling the fit function from LinearRegression by access operator
from variable regressor and pass input train and output train as
parameter
#train model
regressor.fit(x_train,y_train)
```

```
LinearRegression()
```

```
#from above code, weight and bais will be calculated
#slope or weight will be calculated and saved in coef_ of regressor,
we assign this value in weight variable
weight = regressor.coef_
weight
```

```
array([[7.90840255e-01, 3.01968165e-02, 3.10148566e-02,
4.63028992e+02,
        3.04799573e+02]])
```

```
#bais will be calculated and saved in intercept of regressor, we
assign this value in bais variable
bais = regressor.intercept_
bais
```

```
array([42403.87087053])
```

```
#now predict using x_test data and assign to y_pred varaible using
above model created which is present in regressor
y_pred=regressor.predict(x_test)

#import r2_score from sklean.metrics functions
from sklearn.metrics import r2_score
#compare actual test (y_test) vs predicted value on x_test which is
assign in y_pred
r_score=r2_score(y_test,y_pred)

# print the r_score verify it is near to 1 or not. If near to 1, we
can confirm that created model is good else created model is bad
r_score
```

```
0.9358680970046241
```

```
#Save the model created using pickle function, created the sav file to
save the model
```

```
import pickle
filename="finalized_model_Mul_linear.sav"

#save the regressor where model is present in filename using
pickle.dump function
pickle.dump(regressor,open(filename,'wb'))

#read the filename where model is saved in the variable by load
function in pickle
loaded_model=pickle.load(open("finalized_model_Mul_linear.sav",'rb'))
#assign variable to pass the value during runtime
#R&DSpend = int(input("Enter the amount for R&D Spend:"))
#Administration = int(input("Enter the amount for Administration
Spend:"))
#MarketingSpend = int(input("Enter the amount for Marketing Spend:"))
#State_Florida = int(input("Enter the amount for State_Florida:"))
#State_Newyork = int(input("Enter the amount for State_Newyork:"))
#assign the result variable to call the predict function on
loded_module variable where model saved
#result =
loaded_model.predict([[R&DSpend,Administration,MarketingSpend,State_Fl
orida,State_Newyork]])
#result = loaded_model.predict([[2019,1234,4450,4565,1,0]])
result = loaded_model.predict([[1234,4450,4565,1,0]])
```

```
C:\Users\Maheshwaran\anaconda3\Lib\site-packages\sklearn\base.py:464:
UserWarning: X does not have valid feature names, but LinearRegression
was fitted with feature names
  warnings.warn(
```

```
result
```

```
array([[44118.75539065]])
```