In [1]:

```
#Importing Libraries
import pandas as pd
import numpy as np
```

In [2]:

```
# Loading Data
data_df = pd.read_csv('C:/Users/DELL/Desktop/AI ML _ SCM/Assignment 2/Assignment2_Dataset.c
```

In [3]:

```
# Checking if loaded data is correct or not
data_df.head()
```

Out[3]:

	S.No	District	Number of Positive Cases	Positivity Rate	Total Achievement towards 1st Dose of Covishield and Covaxin	Achievement towards 2nd Dosage Covishield 18+
0	1	Ariyalur	16	1.6	196727	6169
1	2	Chennai	183	0.8	2542245	89809
2	3	Coimbatore	205	1.8	1170289	19173
3	4	Cuddalore	57	1.3	612752	6333
4	5	Dharmapuri	21	1.1	446398	4172

In [43]:

```
# removing column number of positive cases from x axis
x=data_df.drop(["District","Number of Positive Cases","Positivity Rate"],axis=1).values
y=data_df['Number of Positive Cases'].values
```

In [44]:

```
print(x)
```

```
[[      1  196727    6169]
 [      2 2542245   89809]
 [      3 1170289   19173]
 [      4  612752    6333]
 [      5  446398    4172]
 [      6  324812    8830]
 [      7  697172    3814]
 [      8  336941    2982]
 [      9  298154    2478]
 [     10  573879   19062]
 [     11  306036   13602]
 [     12  593762   14321]
 [     13  740458   10588]
 [     14  214557    5517]
 [     15  210928    4452]
 [     16  575070   13487]
 [     17  402618   15358]
 [     18  263807    4106]
 [     19  263975    3366]
 [     20  757730   13380]
 [     21  390138   11124]
 [     22  619647   11976]
 [     23  310232   12471]
 [     24  789546   29945]
 [     25  365717    8459]
 [     26  712324    8504]
 [     27  479825   11228]
 [     29  358852    6480]
 [     30  396416    5905]
 [     31  455226    3222]
 [     32  262771    9034]]
```

In [45]:

```
print(y)
```

```
[ 16 183 205  57  21   8 152  24  31  25  13  20  14  14  48  50  36  25
  20  79  21  95   5  52  15  76  58  39  27  43   6]
```

In [56]:

```
# Splitting Data
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.8,random_state=0)
```

In [57]:

```
# Training the Model
from sklearn.linear_model import LinearRegression
ml=LinearRegression()
ml.fit(x_train,y_train)
```

Out[57]:

```
LinearRegression()
```

In [58]:

```
# Prediction Model and Printing
y_pred=ml.predict(x_test)
print(y_pred)
```

```
[  12.25016519  104.81442514   53.86368367   25.81770859   88.52637723
   84.33147265   56.39401299   32.89594385   68.20038641   19.48957693
   28.7703103    36.64017963   53.34576348   59.13066512   73.88795611
   57.68152783 -161.91484295   85.61834381   54.50351347   42.92220239
   72.61616238   93.36011674   57.68526558   14.14186315   50.54786354]
```

In [59]:

```
# Predicting value of positive cases using model
ml.predict([[1,196727,6169]])
# Predicted model   --> 22.
# According to data --> 16.
```

Out[59]:

```
array([22.08287011])
```

In [60]:

```
# Evaluating the model using R2 value
from sklearn.metrics import r2_score
r2_score(y_test,y_pred)
# Conclusion no correlation between data and parameters
```

Out[60]:

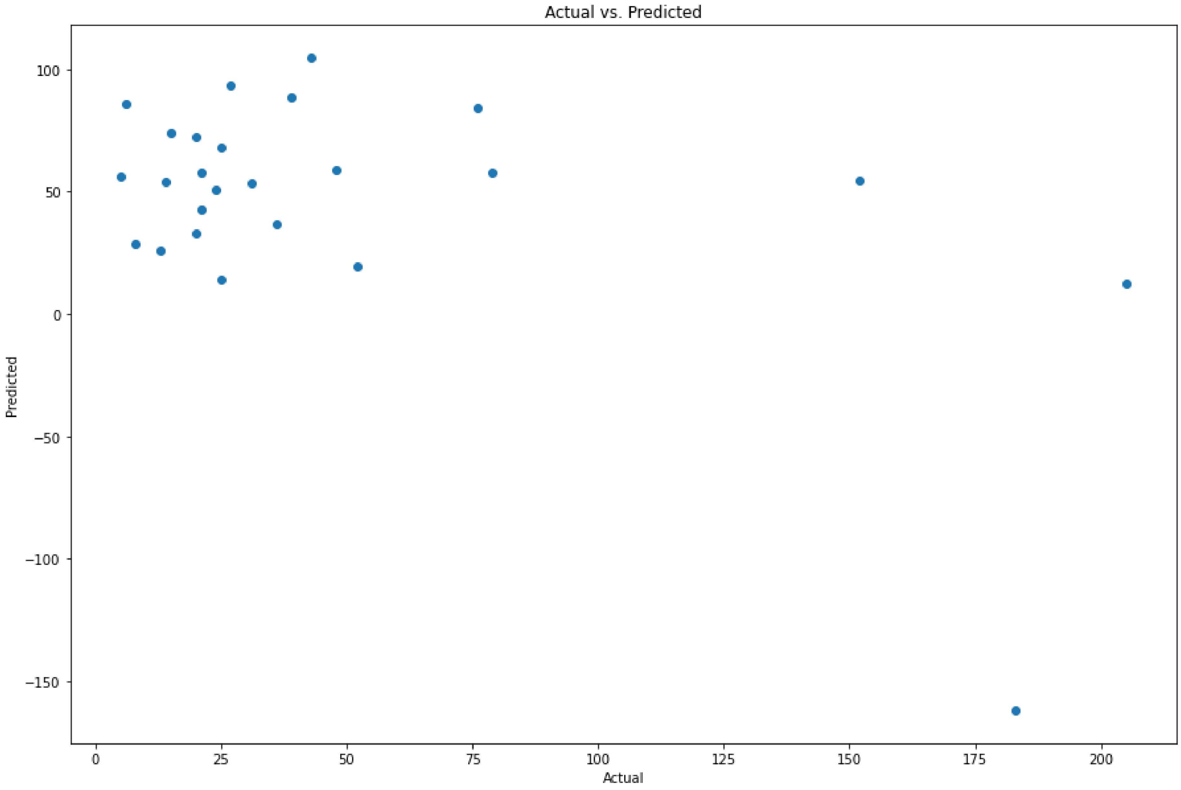
```
-1.8750765503305127
```

In [62]:

```
import matplotlib.pyplot as plt
plt.figure(figsize=(15,10))
plt.scatter(y_test,y_pred)
plt.xlabel('Actual')
plt.ylabel('Predicted')
plt.title('Actual vs. Predicted')
```

Out[62]:

```
Text(0.5, 1.0, 'Actual vs. Predicted')
```



In [63]:

```
pred_y_df=pd.DataFrame({'Actual Value':y_test,'Predicted Value':y_pred, 'Difference': y_tes
pred_y_df[0:31]
```

Out[63]:

	Actual Value	Predicted Value	Difference
0	205	12.250165	192.749835
1	43	104.814425	-61.814425
2	14	53.863684	-39.863684
3	13	25.817709	-12.817709
4	39	88.526377	-49.526377
5	76	84.331473	-8.331473
6	5	56.394013	-51.394013
7	20	32.895944	-12.895944
8	25	68.200386	-43.200386
9	52	19.489577	32.510423
10	8	28.770310	-20.770310
11	36	36.640180	-0.640180
12	31	53.345763	-22.345763
13	48	59.130665	-11.130665
14	15	73.887956	-58.887956
15	21	57.681528	-36.681528
16	183	-161.914843	344.914843
17	6	85.618344	-79.618344
18	152	54.503513	97.496487
19	21	42.922202	-21.922202
20	20	72.616162	-52.616162
21	27	93.360117	-66.360117
22	79	57.685266	21.314734
23	25	14.141863	10.858137
24	24	50.547864	-26.547864

In [ ]: