EXPERIMENT NO 3 DTATA SET 2

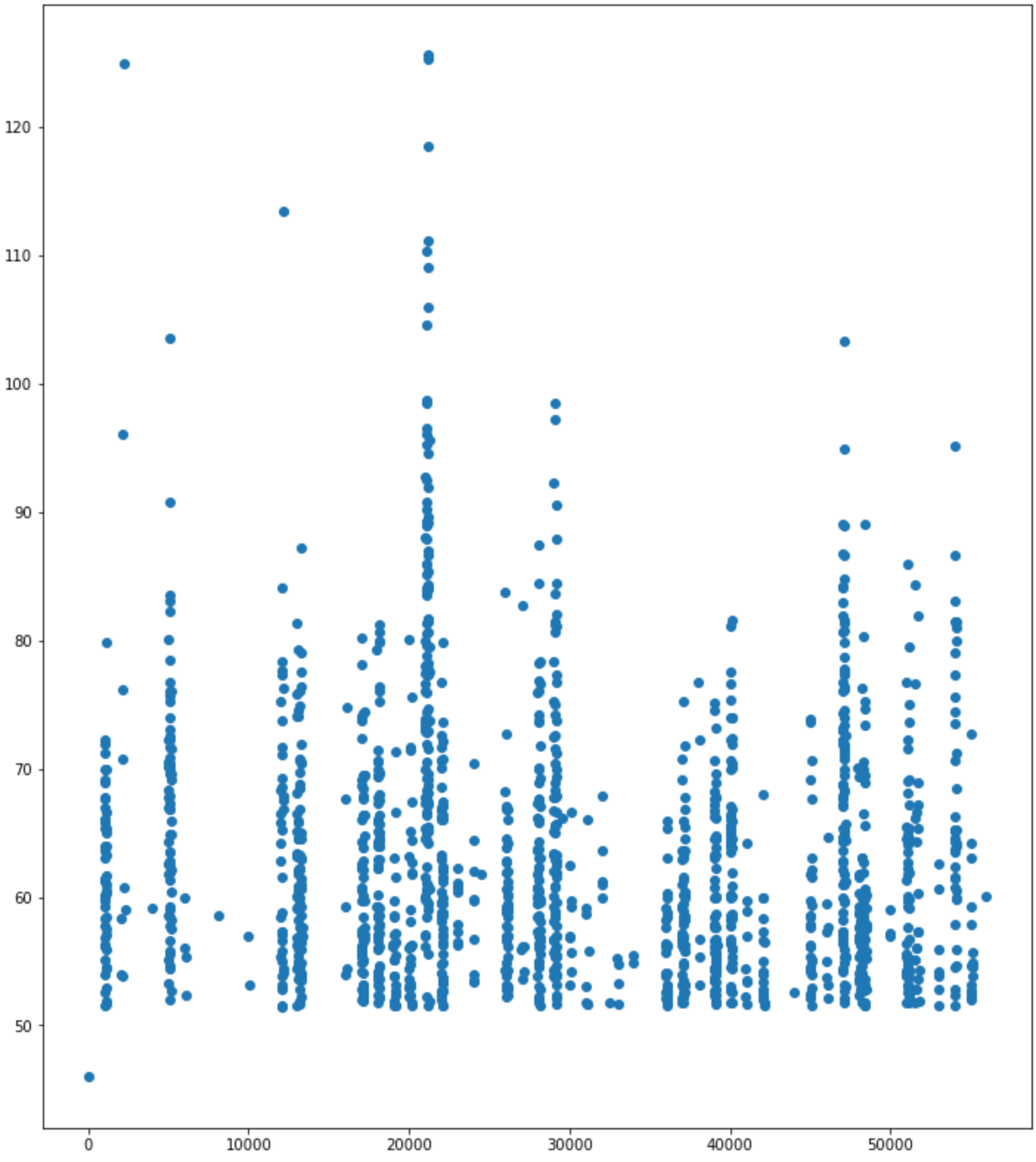
SIMPLE LINEAR REGRATION

IMPORT THE REQUIRED LIBRARIES

In [4]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

plt.rcParams["figure.figsize"] = (12.0, 14.0)

data = pd.read_csv('deathupdated.csv')
```

PLOT THE DATA ACCORDING TO FEAURES

In [5]:
```
x=data.iloc[:,0]
y=data.iloc[:,1]

plt.scatter(x,y)
plt.show()
```

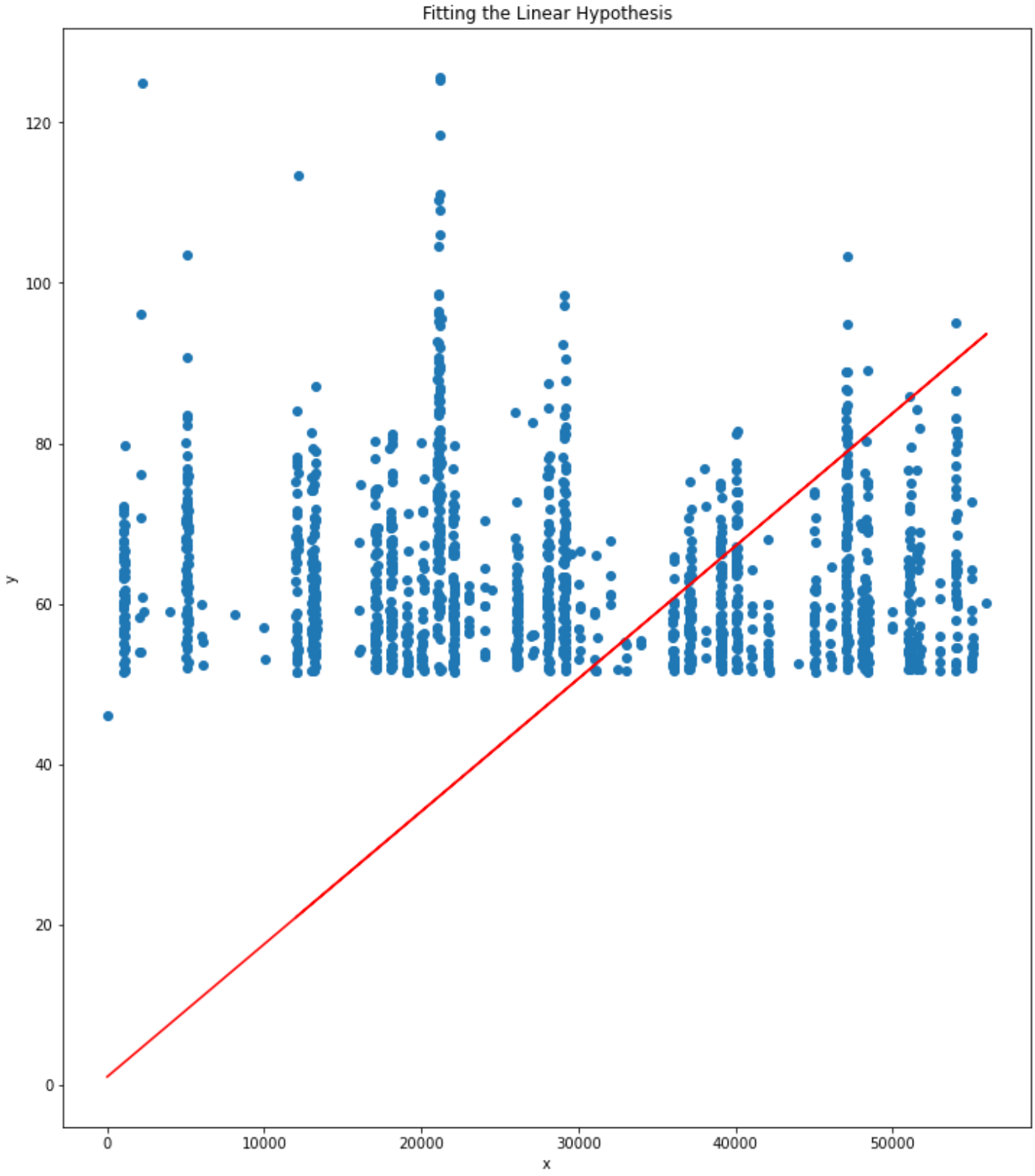


INITIALIZE THE VALUES OF THETA AND LEARNING RATE

In [7]:

```
theta0 = 1.000
theta1 = 0.000
alpha = 0.000000000001

m = float(len(X))# no of samples in data set get it
epsilon = 0.001
CostOld = 9999
diff = 1
iteration = 1
```

```
Iteration =  4095
theta0 =  1.0000000999674783 ,theta1 =  0.0016545220613892865
```




```
pred PREDICTED FOR fipe  2185 AS pred =  4.62
```

PREDICT THE HYPOTHESIS FOR GIVEN DATA

In [ ]:

```
while abs(diff) > epsilon:#diff is toooo small then stop itteration
    #automatic convergence test ACT some threshold should be  there .. that is epsilon
    Y_pred = theta1 * X + theta0#h theta(x)=o0+o1x) and prediction for all vector X sample
    Cost = (1 / m) * sum((Y_pred - Y) ** 2)#X&Y capital is all vector not single entity
    #   print(Cost)
```

```
    D_theta1 = (1 / m) * sum(X * (Y_pred - Y))# partial derivative wrt to theta1
    D_theta0 = (1 / m) * sum(Y_pred - Y)
    theta1 = theta1 - alpha * D_theta1
    theta0 = theta0 - alpha * D_theta0
    #print(cost)
    diff = CostOld - Cost
    #   print(diff)

    CostOld = Cost
    iteration = iteration + 1
    # plt.scatter(X,Y)
    # plt.plot([min(X), max(X)],[min(Y_pred),max (Y_pred)], color='red')
    # plt.show()
```

PLOT THE HYPOTHESIS ON DATA

In [ ]:

```
print('Iteration = ', str(iteration))
print("theta0 = ", theta0, ",theta1 = ", theta1)


plt.scatter(X, Y)
plt.plot(X, Y_pred, color='red')
plt.xlabel('x')
plt.ylabel('y')
plt.title('Fitting the Linear Hypothesis')
plt.show()


test_X =2185 #initializing the value of (x theta1 *x)
Y_pred = theta1 * test_X + theta0
print('pred PREDICTED FOR fipe ', str(test_X), 'AS pred = ', round(Y_pred,2))
```

COMPARE THE DATA WITH ACTUAL VALUE

In [11]:

```
data
```

Out[11]:

	fips	pred
0	0	46.0
1	21193	125.6
2	21197	125.3
3	2185	124.9
4	21189	118.5
...	...	...
1503	22103	51.5
1504	42131	51.5
1505	45077	51.5
1506	48387	51.5
1507	12055	51.4

1508 rows × 2 columns