EXPERIMENT NO 3 DATASET 1

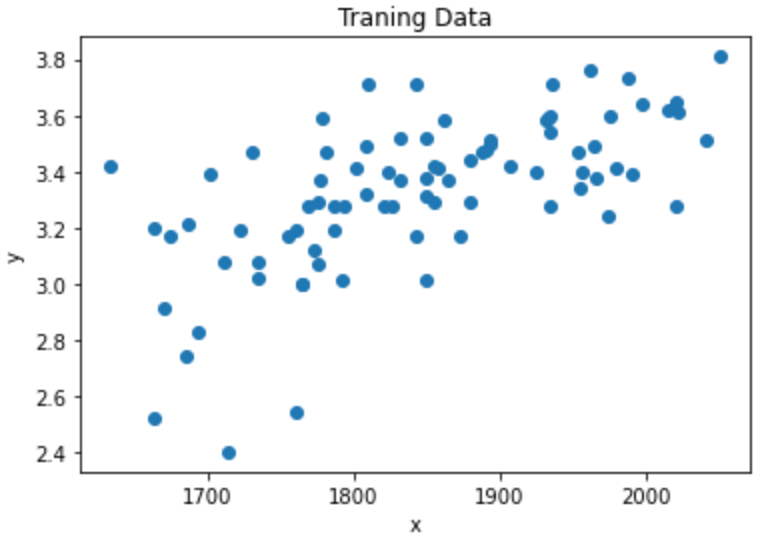
SIMPLE LINEAR REGRATION

IMPORT REQUIRED LIBRARIES

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

PLOT THE DATA ACCORDING TO FEAURES AND INITIALIZE THE VALUES OF THETA AND LEARNING RATE

In [5]:
```
plt.rcParams["figure.figsize"] = (6.0, 4.0)

data = pd.read_csv('1.01. Simple linear regression.csv')
data

#output y extracting and storing in data frame
#plt.scatter(SAT,GPA)
X = data.iloc[:, 0]#input x
Y = data.iloc[:, 1]
plt.scatter(X,Y)
plt.xlabel('x')
plt.ylabel('y')
plt.title('Traning Data')
plt.show()

theta0 = 1.039654822716
theta1 = 0.001738856046

alpha = 0.0000001

m = float(len(X))# no of samples in data set get it
epsilon = 0.000001
CostOld = 99999
diff = 1
iteration = 1
```



PREDICT THE HYPOTHESIS FOR GIVEN DATA

In [6]:

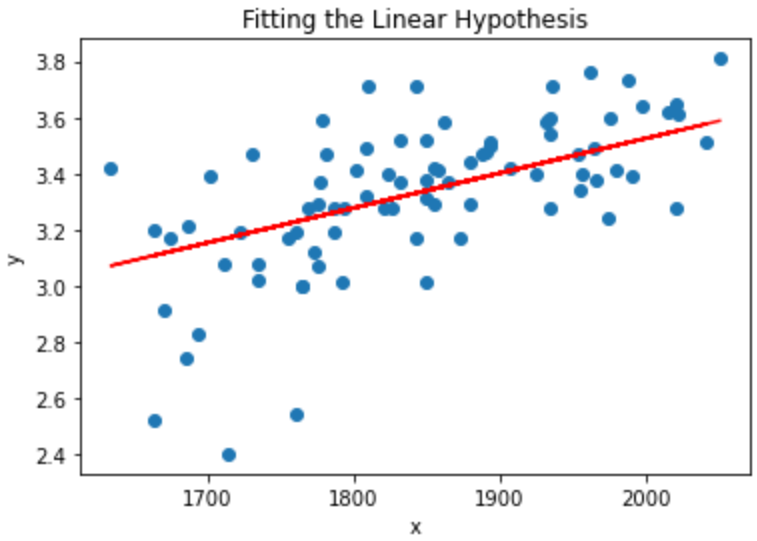
```
while abs(diff) > epsilon:#diff is toooo small then stop itteration
    #automatic convergence test ACT some threshold should be  there .. that is epsilon
    Y_pred = theta1 * X + theta0#h theta(x)=o0+o1x) and prediction for all vector X sample
    Cost = (1 / m) * sum((Y_pred - Y) ** 2)#X&Y capital 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 [7]:

```
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 =1666  #float(input ("enter no of year "))#1.2 #initializing the value of (x theta
Y_pred = theta1 * test_X + theta0
print('STA of ', str(test_X), ' has predected GPA = ', round(Y_pred,2))
```

```
Iteration =  19
theta0 =  1.0396545504457453 ,theta1 =  0.0012429025174922757
```




```
STA of  1666  has predected GPA =  3.11
```

COMPARE THE DATA WITH ACTUAL VALUE

In [35]:
```
data
```

Out[35]:

	SAT	GPA
0	1714	2.40
1	1664	2.52
2	1760	2.54
3	1685	2.74
4	1693	2.83
...	...	...
79	1936	3.71
80	1810	3.71
81	1987	3.73
82	1962	3.76
83	2050	3.81

84 rows × 2 columns

In [ ]:
```
```