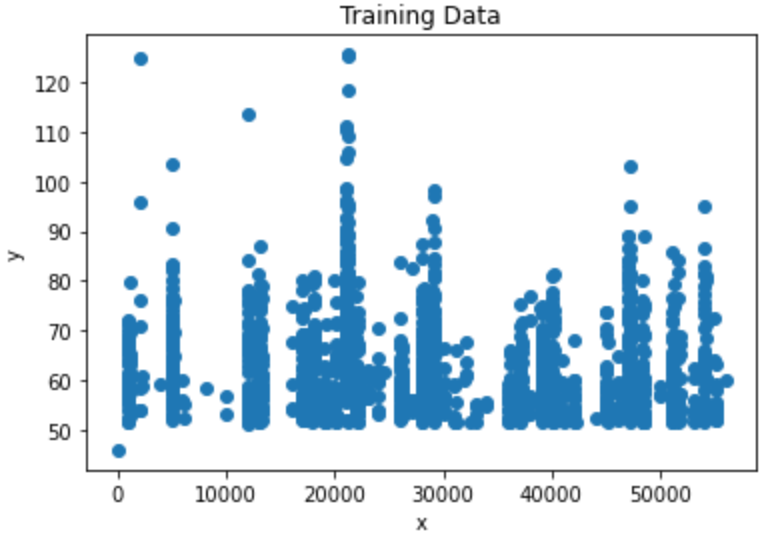
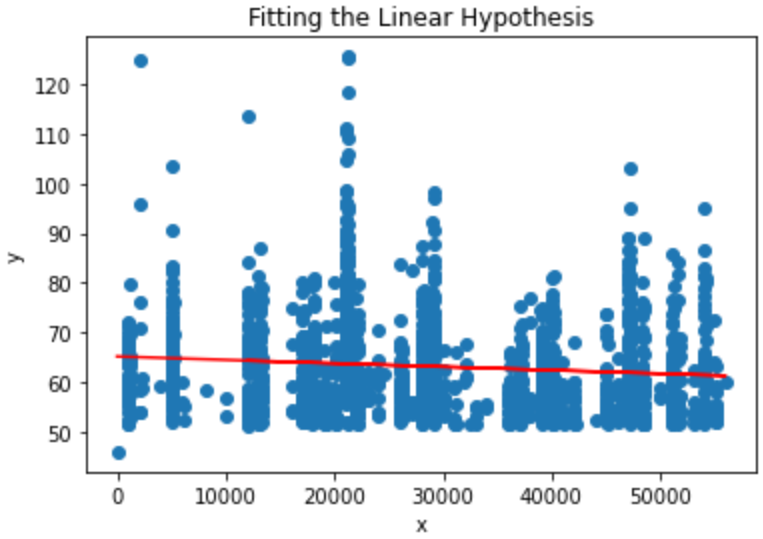
In [5]:

```
#Making the import
import numpy as np
from sklearn.linear_model import LinearRegression
import pandas as pd
import matplotlib.pyplot as plt
plt.rcParams['figure.figsize'] = (6.0, 4.0)
from sklearn.metrics import mean_squared_error # for calculating mean_squared error
#Preprocessing Input data
data = pd.read_csv('death.csv')
X = data.iloc[:, 0]
Y = data.iloc[:, 1]
plt.scatter(X, Y)
plt.xlabel('x')
plt.ylabel('y')
plt.title('Training Data')
plt.show()
X = np.array(X).reshape((-1, 1))

#Building the model
model = LinearRegression()
model.fit(X, Y)
print('theta0 = ', model.intercept_, 'theta1 = ', model.coef_)

#Making predictions for Training Data
# Making predictions for Training Data
Y_pred = model.predict(X)

#Computing mean_squared error
print('RMSE error for Regression=>',np.sqrt(mean_squared_error(Y,Y_pred)))
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()

#Making predictions
test_X = 47
Y_pred = model.predict(np.array(test_X).reshape((-1, 1)))
#predict() is equivalent to below
Y_pred2 = model.intercept_ + model.coef_ * test_X
print("pred Prediction for ", str(test_X), " fips =", Y_pred, Y_pred2)
```




```
theta0 =  65.20330594400816 theta1 =  [-6.89006386e-05]
RMSE error for Regression=> 10.24901061264842
```




```
pred Prediction for  47  fips = [65.20006761] [65.20006761]
```

In [3]:

```
data
```

Out[3]:

	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

In [ ]: