```
# %%

# ================================================================================
# SIMPLE LINEAR REGRESSION
# ================================================================================


# %%

# importing the scikit-learn.dataset library

from sklearn import datasets


#%%

# generating the dataset for 1000 samples with linear relation from make_regression() fucntion of skearn

x, y = datasets.make_regression(n_samples=1000,#number of samples
                                n_features=1,#number of features
                                n_informative=1,#number of useful features
                                noise=10)#bias and standard deviation of the guassian noise


#%%

# importing the train_test_split function of sklearn

from sklearn.model_selection import train_test_split


#%%

# creating training and test datasets with train_test_split func. of sklearn.model_selection

train_x,test_x,train_y,test_y=train_test_split(x,y, test_size=0.2)

print("train data size:",len(train_x)) # 800

print("test data size:",len(test_x)) # 200


#%%
```

```
# importing the matplotlib library

import matplotlib.pyplot as plt


#%%

# plotting the x and y values generated

plt.scatter(x,y)


#%%

# importing the LinearRegression() func. of sklearn.linear_model

from sklearn.linear_model import LinearRegression


#%%

# craeting and fitting the linear model on the training dataset

linear = LinearRegression()
linear.fit(train_x,train_y)
y_predicted=linear.predict(test_x)

#%%

%matplotlib inline

# ploting the predicted value with the actual one

plt.figure(figsize=(15,15))

p1=max(max(y_predicted),max(test_y))
p2=min(min(y_predicted),min(test_y))

plt.plot([p1,p2],[p1,p2], 'b-')

plt.xlabel("True Values")
plt.ylabel("Predicted Values")

plt.scatter(test_y,y_predicted,color='crimson')
```

```
plt.axis("equal")


#%%

# calculating the model accuracy

from sklearn.metrics import r2_score

print("Model Accuracy: ",str(r2_score(test_y,y_predicted))+"") # 0.9188302349633785
# %%

# ================================================================================
# THE END
# ================================================================================
```