

```

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
import time

# Load the data from the CSV file
file_path = '/content/test.csv'

try:
    data = pd.read_csv(file_path)
    print("Data Loaded Successfully:")
    print(data)
except FileNotFoundError:
    print(f"Error: The file {file_path} was not found.")
    exit()

# Prepare the data
X = data[['Vehicle_Count']] # Features
y = data['Light_Duration'] # Target variable

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Create a linear regression model
model = LinearRegression()

# Train the model
model.fit(X_train, y_train)

# Make predictions
predictions = model.predict(X_test)

# Evaluate the model
mse = mean_squared_error(y_test, predictions)
r2 = r2_score(y_test, predictions)

print("\nModel Evaluation:")
print(f"Mean Squared Error: {mse:.2f}")
print(f"R^2 Score: {r2:.2f}")

# Visualize the results
plt.scatter(X_test, y_test, color='blue', label='Actual Data')
plt.scatter(X_test, predictions, color='red', label='Predictions')
plt.plot(X_test, predictions, color='green', linewidth=2, label='Regression Line')
plt.title('Vehicle Count vs Light Duration')
plt.xlabel('Vehicle Count')
plt.ylabel('Light Duration')
plt.legend()
plt.show()

# Function to predict light duration based on vehicle count
def predict_light_duration(vehicle_count):
    if vehicle_count < 0:
        raise ValueError("Vehicle count cannot be negative.")
    return model.predict(np.array([[vehicle_count]]))[0]

# Function to simulate traffic light control
def traffic_light_control(vehicle_count):
    predicted_duration = predict_light_duration(vehicle_count)
    print(f"\nPredicted Light Duration for {vehicle_count} vehicles: {predicted_duration:.2f} seconds")

    # Simulate the traffic light timing
    print("Traffic Light will be GREEN for {:.2f} seconds.".format(predicted_duration))
    time.sleep(predicted_duration) # Simulate the duration of the green light
    print("Traffic Light will turn RED now.")

# Example usage
try:
    vehicle_count_input = 30 # Example vehicle count
    traffic_light_control(vehicle_count_input)
except ValueError as e:
    print(f"Error: {e}")

```



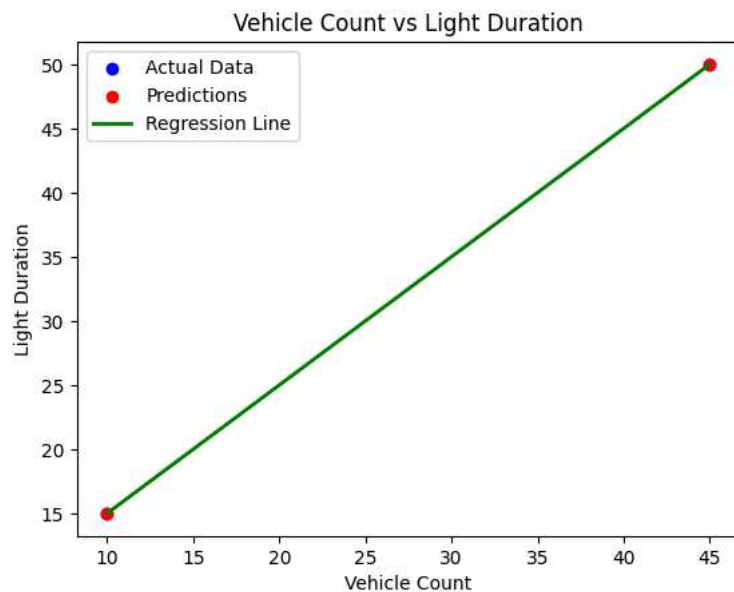
Data Loaded Successfully:

	Vehicle_Count	Light_Duration
0	5	10
1	10	15
2	15	20
3	20	25
4	25	30
5	30	35
6	35	40
7	40	45
8	45	50
9	50	55

Model Evaluation:

Mean Squared Error: 0.00

R<sup>2</sup> Score: 1.00



/usr/local/lib/python3.11/dist-packages/sklearn/utils/validation.py:2739: UserWarning: X does not have valid feature names, but LinearRegression will use the indices to access data  
warnings.warn(

Predicted Light Duration for 30 vehicles: 35.00 seconds

Traffic Light will be GREEN for 35.00 seconds.

Traffic Light will turn RED now.