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

def load_data():
    # Synthetic sample data for demo
    data = {
        'Mileage': [15000, 30000, 45000, 60000, 75000, 90000, 105000, 120000,
135000, 150000],
        'Age': [1, 2, 3, 4, 5, 6, 7, 8, 9, 10],
        'EngineSize': [1.2, 1.4, 1.6, 1.8, 2.0, 1.2, 1.4, 1.6, 1.8, 2.0],
        'Price': [25000, 22000, 20000, 18000, 16000, 14000, 12000, 10000, 8000,
6000]
    }
    return pd.DataFrame(data)

def train_model(df):
    X = df[['Mileage', 'Age', 'EngineSize']]
    y = df['Price']
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3,
random_state=0)

    model = LinearRegression()
    model.fit(X_train, y_train)

    y_pred = model.predict(X_test)
    print(f"Model Coefficients: {model.coef_}")
    print(f"Intercept: {model.intercept_}")
    print(f"R2 Score: {r2_score(y_test, y_pred):.2f}")
    print(f"Mean Squared Error: {mean_squared_error(y_test, y_pred):.2f}")

    return y_test, y_pred

def visualize_results(y_test, y_pred):
    plt.figure()
    plt.plot(y_test.values, label='Actual Price', marker='o')
    plt.plot(y_pred, label='Predicted Price', marker='x')
    plt.title("Car Price Prediction: Actual vs Predicted")
    plt.xlabel("Test Sample")
    plt.ylabel("Car Price")
    plt.legend()
    plt.grid(True)
    plt.show()

def main():
    print("=== Car Price Prediction using Linear Regression ===")
    df = load_data()
    y_test, y_pred = train_model(df)
    visualize_results(y_test, y_pred)

if __name__ == "__main__":
    main()
```