

```

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 house data
    data = {
        'Area': [1000, 1500, 1800, 2400, 3000, 3500, 4000, 4200, 5000, 5500],
        'Bedrooms': [2, 3, 3, 4, 4, 4, 5, 5, 6, 6],
        'Bathrooms': [1, 2, 2, 3, 3, 3, 4, 4, 5, 5],
        'Price': [150000, 200000, 220000, 300000, 350000, 380000, 450000,
470000, 550000, 580000]
    }
    return pd.DataFrame(data)

def train_model(df):
    X = df[['Area', 'Bedrooms', 'Bathrooms']]
    y = df['Price']
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3,
random_state=42)

    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"MSE: {mean_squared_error(y_test, y_pred):.2f}")

    return y_test, y_pred

def visualize(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("House Price Prediction - Actual vs Predicted")
    plt.xlabel("Test Sample Index")
    plt.ylabel("Price")
    plt.legend()
    plt.grid(True)
    plt.show()

def main():
    print("=== House Price Prediction Application ===")
    df = load_data()
    y_test, y_pred = train_model(df)
    visualize(y_test, y_pred)

if __name__ == "__main__":
    main()

```