# Linear Regression to Predict House Prices with Visualization

# Step 1: Import Libraries

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

# Step 2: Load Data

data = pd.DataFrame({

'sqft': [2000, 1500, 2500, 1800, 2200],

'bedrooms': [3, 2, 4, 3, 4],

'bathrooms': [2, 1, 3, 2, 3],

'price': [500000, 350000, 600000, 400000, 550000]

})

print("Data Preview:\n", data)

# Step 3: Split Features and Target

X = data[['sqft', 'bedrooms', 'bathrooms']]

y = data['price']

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Step 4: Train Linear Regression Model

model = LinearRegression()

model.fit(X\_train, y\_train)

# Step 5: Make Predictions

y\_pred = model.predict(X\_test)

# Step 6: Evaluate the Model

mse = mean\_squared\_error(y\_test, y\_pred)

r2 = r2\_score(y\_test, y\_pred)

print("\nMean Squared Error:", mse)

print("R-squared Score:", r2)

# Step 7: Predict Price of a New House

new\_house = np.array([[2500, 4, 3]]) # Example: 2500 sqft, 4 bedrooms, 3 bathrooms

predicted\_price = model.predict(new\_house)

print("\nPredicted Price for New House:", predicted\_price[0])

# Step 8: Visualization - Actual vs Predicted

plt.scatter(y\_test, y\_pred, color='blue')

plt.plot([min(y\_test), max(y\_test)], [min(y\_test), max(y\_test)], color='red', linewidth=2)

plt.xlabel("Actual Price")

plt.ylabel("Predicted Price")

plt.title("Actual vs Predicted House Prices")

plt.show()