**AI\_Phase 3**

**Dataset used = house\_price\_data.csv**

**CODE:**

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\_absolute\_error, mean\_squared\_error, r2\_score

# Step 1: Data Loading and Exploration

# Load the dataset

data = pd.read\_csv("house\_price\_data.csv")

# Explore the dataset

print(data.head())

print(data.info())

print(data.describe())

# Step 2: Data Preprocessing

# Handle missing values

data.fillna(0, inplace=True)

# Encode categorical variables if needed

# For example, you can use one-hot encoding

# Step 3: Data Splitting

X = data.drop("price", axis=1)

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: Model Selection and Training

model = LinearRegression()

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

# Step 5: Model Evaluation

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

mae = mean\_absolute\_error(y\_test, y\_pred)

rmse = np.sqrt(mean\_squared\_error(y\_test, y\_pred))

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

print("Mean Absolute Error:", mae)

print("Root Mean Square Error:", rmse)

print("R-squared:", r2)

# Step 6: Data Visualization

# Visualize the predictions vs. actual prices

plt.scatter(y\_test, y\_pred)

plt.xlabel("Actual Prices")

plt.ylabel("Predicted Prices")

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

plt.show()