

Task 1

Implement a linear regression model to predict the prices of houses based on their square footage and the number of bedrooms and bathrooms.

Code to create dataset:

```
import pandas as pd
import numpy as np

# Create dummy data for demonstration
dummy_data = {
    'square_footage': np.random.randint(1000, 3000, 100),
    'bedrooms': np.random.randint(2, 5, 100),
    'bathrooms': np.random.randint(1, 4, 100),
    'price': np.random.randint(150000, 500000, 100)
}
df_dummy = pd.DataFrame(dummy_data)

# Save the dummy data to 'your_dataset.csv'
df_dummy.to_csv('your_dataset.csv', index=False)

print("Dummy 'your_dataset.csv' created successfully.")
```

Python code

```
# House Price Prediction using Linear Regression

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

# Load Dataset (Change file name if needed)
df = pd.read_csv("your_dataset.csv")
```

```

# If column names are different, print and adjust
print("Columns in Dataset:", df.columns)

# Select Features and Target (Modify if needed)
X = df[['square_footage', 'bedrooms', 'bathrooms']]
y = df['price']

# Remove missing values
df = df.dropna()

# Train-Test Split
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42
)

# Create & Train Model
model = LinearRegression()
model.fit(X_train, y_train)

# Predictions
y_pred = model.predict(X_test)

# Evaluation
mae = mean_absolute_error(y_test, y_pred)
mse = mean_squared_error(y_test, y_pred)
rmse = np.sqrt(mse)
r2 = r2_score(y_test, y_pred)

print("\n===== MODEL RESULTS =====")
print("Intercept:", model.intercept_)
print("Coefficients:", model.coef_)
print("MAE:", mae)
print("MSE:", mse)
print("RMSE:", rmse)
print("R2 Score:", r2)

# Predict New House
new_house = np.array([[2000, 3, 2]])
predicted_price = model.predict(new_house)

print("\nPredicted Price for 2000 sqft, 3 bed, 2 bath:")
print(predicted_price[0])

```

```
# Visualization
plt.scatter(y_test, y_pred)
plt.xlabel("Actual Prices")
plt.ylabel("Predicted Prices")
plt.title("Actual vs Predicted Prices")
plt.show()
```

Output

