Date Page

2) Linear Regression : predict house prices using the Boston Housing dataset.

```
Import pandas as pd
Import numpy as np
Import seaborn as sns
Import matplotlib.pyplot as plt
from sklearn.datasets import load_boston
from sklearn.model_selection import
train test split.
from sklearn.linear_model import Linear
Regression
from sklearn.metrics import mean_sq
_uared Error, r2_score.

boston = load_boston()
df = pd.DataFrame(boston.data, columns=
boston.feature_names)
df['MEDV'] = boston.target
print(df.head())
X = df.drop('MEDV', axis=1)
y = df['MEDV']
X_train, X_test, y_train, y_test =
train_test_split(X, y, test_size=0.2,
random_state=42)
model = LinearRegression()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)
print(f"Mean Squared Error: {mse:.2f}")
print(f"R^2 Score: {r2:.2f}")
```

Date Page

```
plt.figure(figsize=(8,6))
sns.scatterplot(x=y_test, y=y_pred)
plt.xlabel("Actual prices ($1000's)")
plt.ylabel("predicted prices ($1000's)")
plt.title("Actual vs predicted house prices")
plt.plot([0,50],[0,50],'...r')
plt.show
```

OUTPUT :

	MedInc	HouseAge	AveRooms	AveBedrms	Population	AveOccup
0	8.3252	41.0	6.984127	1.023810	322.0	2.555556
1	8.3014	21.0	6.238138	0.971880	2401.0	2.109842
2	7.2574	52.0	8.288136	1.073446	496.0	2.802260
3	5.6431	52.0	5.817352	1.073059	558.0	2.547945
4	3.8462	52.0	6.281853	1.081081	565.0	2.181467

	Longitude	MedHouseVal
0	-122.23	4.526
1	-122.22	3.585
2	-122.24	3.521
3	-122.25	3.413

Mean Squared Error : 0.56

R^2 score : 0.58