```
# 👋✨ AIML Minor Project Introduction

print("""
🌟 Hey there, Buds! 🌟

This is my AIML Minor Project for the course 🧠🤖 *AI
Get ready to explore 📊🔍 real-world housing data, vi

Proudly coded by 👉 Charu Priya ✨

Let's dive in! 🚀
""")



# 📚 Import libraries
import pandas as pd
import matplotlib.pyplot as plt

# ✅ Load the dataset (Capital H!)
df = pd.read_csv('/content/Housing.csv')
print("✅ Data loaded successfully!")

# ✅ Define price ranges
bins = [0, 2500000, 5000000, 7500000, 10000000, df['pric
labels = ['0-25 lakhs', '26-50 lakhs', '51-75 lakhs', '7

# ✅ Create new column for price ranges
df['price_range'] = pd.cut(df['price'], bins=bins, label

# ✅ Count houses in each range
price_counts = df['price_range'].value_counts().sort_ind
print("\n📊 Houses in each price range:\n")
print(price_counts)

# ✅ Visualize with a line chart
plt.figure(figsize=(10, 6))
plt.plot(labels, price_counts, marker='o', linestyle='-'
plt.title('Number of Houses in Different Price Ranges')
plt.xlabel('Price Range')
plt.ylabel('Number of Houses')
plt.grid(True)
plt.show()
```

Housing.csv ✕ •••

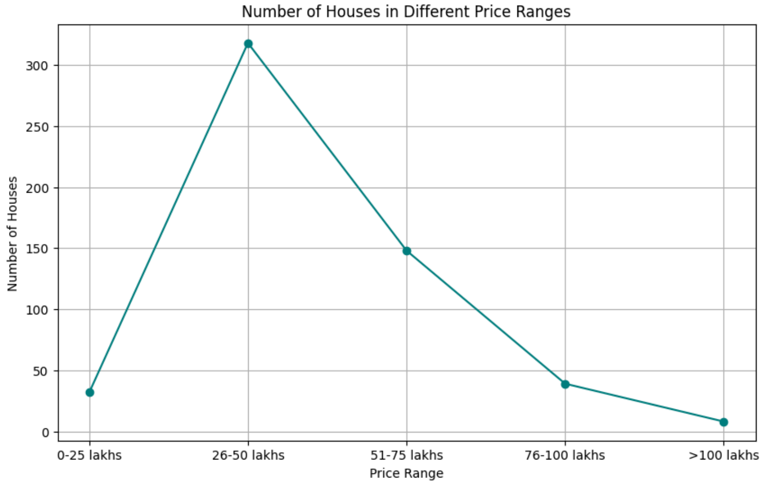
1 to 10 of 545 entries Filter

price	area	bedrooms	bathroo
13300000	7420	4	2
12250000	8960	4	4
12250000	9960	3	2
12215000	7500	4	2
11410000	7420	4	1
10850000	7500	3	3
10150000	8580	4	3
10150000	16200	5	3
9870000	8100	4	1
9800000	5750	3	2

Show 10 per page

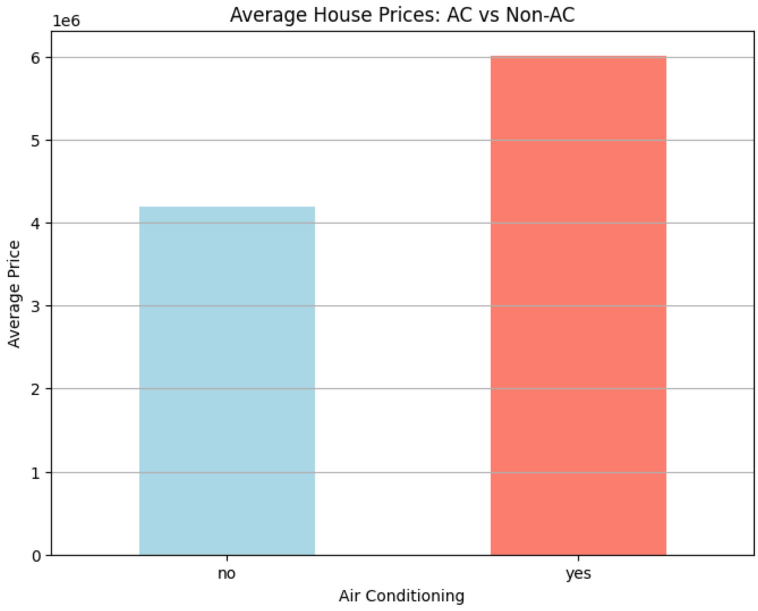
1 2 10 50 55

```
✅ Data loaded successfully!

📊 Houses in each price range:

price_range
0-25 lakhs        32
26-50 lakhs      318
51-75 lakhs      148
76-100 lakhs      39
>100 lakhs         8
Name: count, dtype: int64
```



```
# 📚 Import libraries
import pandas as pd
import matplotlib.pyplot as plt

# ✅ Load the dataset (make sure path is correct)
df = pd.read_csv('/content/Housing.csv')
print("✅ Data loaded successfully!")

# ✅ Check unique values in 'airconditioning'
print("\n🔍 Unique values in 'airconditioning':", df['a

# ✅ Group by AC / non-AC and calculate average prices
avg_prices = df.groupby('airconditioning')['price'].mear
print("\n📊 Average house prices:\n", avg_prices)

# ✅ Plot a bar chart for AC vs non-AC
plt.figure(figsize=(8, 6))
avg_prices.plot(kind='bar', color=['lightblue', 'salmon'
plt.title('Average House Prices: AC vs Non-AC')
plt.xlabel('Air Conditioning')
plt.ylabel('Average Price')
plt.ylabel('Average Price')
plt.xticks(rotation=0)
plt.grid(axis='y')
plt.show()
```

```
✅ Data loaded successfully!

🔍 Unique values in 'airconditioning': ['yes' 'no

📊 Average house prices:
 airconditioning
no     4.191940e+06
yes    6.013221e+06
Name: price, dtype: float64
```



```
# 📚 Import libraries
import pandas as pd
import matplotlib.pyplot as plt

# ✅ Load the dataset
df = pd.read_csv('/content/Housing.csv')
print("✅ Data loaded successfully!")

# ✅ Check column names
print("\n📂 Columns:", df.columns)
```

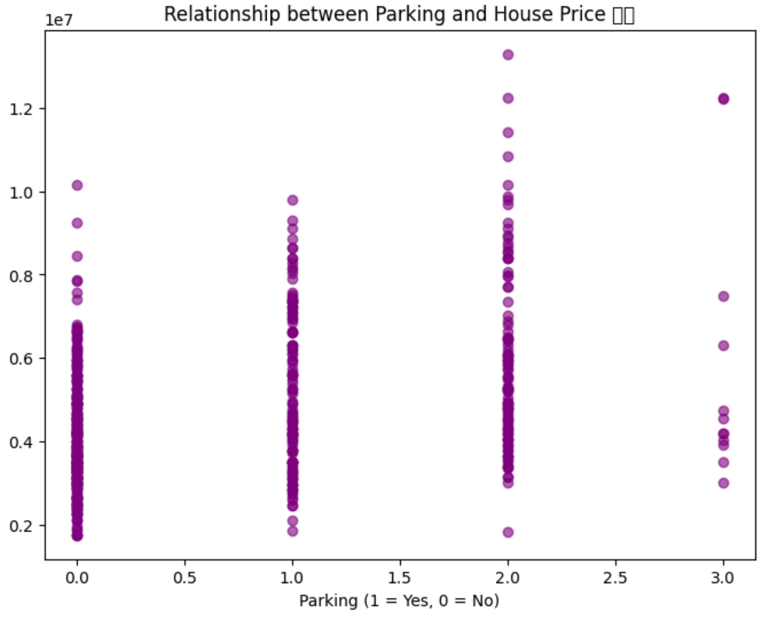
```
# ✅ Example: Check unique values for parking
print("\n🔍 Unique parking values:", df['parking'].uniq

# ✅ Convert parking to numeric if needed
# (Sometimes it's 'yes'/'no'. If it's already numeric, s
if df['parking'].dtype == 'object':
    df['parking'] = df['parking'].map({'yes': 1, 'no': 0

# ✅ Scatter plot: Parking vs House Price
plt.figure(figsize=(8, 6))
plt.scatter(df['parking'], df['price'], color='purple',
plt.title('Relationship between Parking and House Price
plt.xlabel('Parking (1 = Yes, 0 = No)')
plt.ylabel
```

```
✅ Data loaded successfully!

📂 Columns: Index(['price', 'area', 'bedrooms', 'b
       'guestroom', 'basement', 'hotwaterheating',
       'parking', 'prefarea', 'furnishingstatus'],
      dtype='object')

🔍 Unique parking values: [2 3 0 1]
```

```
matplotlib.pyplot.ylabel
def ylabel(ylabel: str, fontdict: dict[str,
Any] | None=None, labelpad: float | None=None,
*, loc: Literal['bottom', 'center', 'top'] |
None=None, **kwargs) -> Text
```

```
/usr/local/lib/python3.11/dist-packages/matplc
Set the label for the y-axis.

Parameters
----------
```

```
/usr/local/lib/python3.11/dist-packages/IPython/co
  func(*args, **kwargs)
/usr/local/lib/python3.11/dist-packages/IPython/co
  func(*args, **kwargs)
/usr/local/lib/python3.11/dist-packages/IPython/co
  fig.canvas.print_figure(bytes_io, **kw)
/usr/local/lib/python3.11/dist-packages/IPython/co
  fig.canvas.print_figure(bytes_io, **kw)
```



```
# 📚 Import libraries
import pandas as pd

# ✅ Load dataset
df = pd.read_csv('/content/Housing.csv')
print("✅ Data loaded successfully!")

# ✅ Check columns and unique prefarea values
print("\n📂 Columns:", df.columns)
print("\n🔍 Unique prefarea values:", df['prefarea'].un

# ✅ Convert prefarea to lower case for consistency (op
df['prefarea'] = df['prefarea'].str.lower()

# ✅ Filter: <5000 sqft & no prefarea
group1 = df[(df['area'] < 5000) & (df['prefarea'] == 'no

# ✅ Filter: >5000 sqft & prefarea
```