

```

# Import necessary libraries
from google.colab import files
import pandas as pd
import matplotlib.pyplot as plt

# Step 1: Upload the dataset
uploaded = files.upload()

# Step 2: Load the dataset
filename = list(uploaded.keys())[0] # Get the uploaded file name
data = pd.read_csv(filename) # Load the dataset (use pd.read_excel if
it's an Excel file)

# Task: Line Plot (Rent vs. Size)
plt.figure(figsize=(10, 6))
plt.plot(data['Size'], data['Rent'], linestyle='-', label='Solid
Line')
plt.plot(data['Size'], data['Rent'], linestyle='--', label='Dashed
Line')
plt.plot(data['Size'], data['Rent'], linestyle=':', label='Dotted
Line')

# Add gridlines
plt.grid(True, linestyle='--', alpha=0.6)

# Add titles and labels
plt.title("Rent vs. Size", fontsize=14, fontweight='bold')
plt.xlabel("Size (sq ft)", fontsize=12)
plt.ylabel("Rent (INR)", fontsize=12)

# Add legend
plt.legend(loc='upper right', fontsize=10)

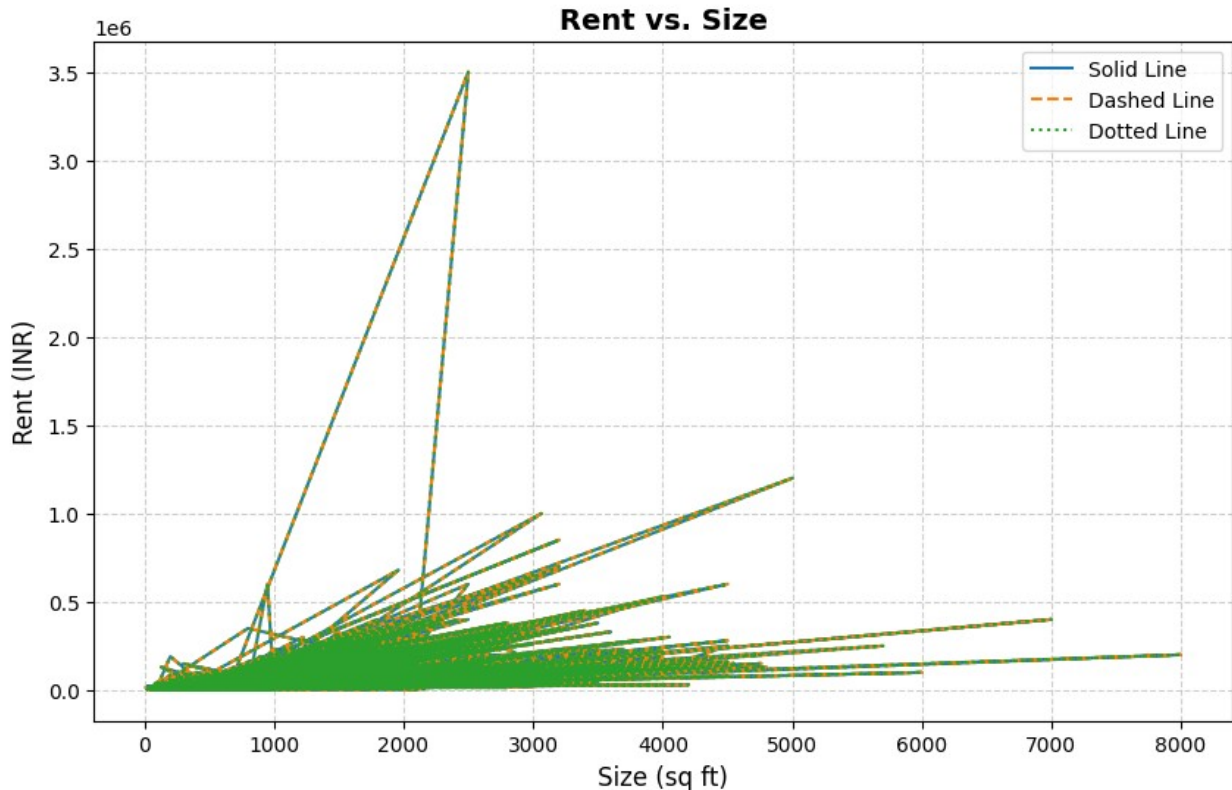
# Save the plot as an image file
output_path_line = '/content/Rent_vs_Size.png'
plt.savefig(output_path_line)
print(f"Line Plot saved as {output_path_line}")

# Show the plot
plt.show()

```

<IPython.core.display.HTML object>

Saving House_Rent_Dataset - House_Rent_Dataset(1).xlsx -
House_Rent_Dataset - House_Rent (2).csv to House_Rent_Dataset -
House_Rent_Dataset(1).xlsx - House_Rent_Dataset - House_Rent (2).csv
Line Plot saved as /content/Rent_vs_Size.png



```
# Import necessary libraries
from google.colab import files
import pandas as pd
import matplotlib.pyplot as plt

# Step 1: Upload the dataset
uploaded = files.upload()

# Step 2: Load the dataset
filename = list(uploaded.keys())[0] # Get the uploaded file name
data = pd.read_csv(filename) # Load the dataset (use pd.read_excel if
it's an Excel file)

# Scatter Plot
plt.figure(figsize=(10, 6))
plt.scatter(data['Size'], data['Rent'], color='blue', alpha=0.7)
plt.title("Scatter Plot: Rent vs. Size", fontsize=14,
fontweight='bold')
plt.xlabel("Size (sq ft)", fontsize=12)
plt.ylabel("Rent (INR)", fontsize=12)
scatter_output_path = '/content/Scatter_Plot.png'
plt.savefig(scatter_output_path)
print(f"Scatter Plot saved as {scatter_output_path}")
plt.show()
```

```

# Histogram
plt.figure(figsize=(10, 6))
plt.hist(data['Rent'], bins=10, color='orange', alpha=0.7,
edgecolor='black')
plt.title("Histogram: Rent Distribution", fontsize=14,
fontweight='bold')
plt.xlabel("Rent (INR)", fontsize=12)
plt.ylabel("Frequency", fontsize=12)
histogram_output_path = '/content/Histogram.png'
plt.savefig(histogram_output_path)
print(f"Histogram saved as {histogram_output_path}")
plt.show()

# Stacked Bar Chart
grouped = data.groupby('Furnishing Status')[['Size', 'Rent']].sum()
grouped.plot(kind='bar', stacked=True, figsize=(10, 6),
color=['#FF5733', '#3357FF'])
plt.title("Stacked Bar Chart: Size and Rent by Furnishing Status",
fontsize=14, fontweight='bold')
plt.xlabel("Furnishing Status", fontsize=12)
plt.ylabel("Total", fontsize=12)
stacked_bar_output_path = '/content/Stacked_Bar_Chart.png'
plt.savefig(stacked_bar_output_path)
print(f"Stacked Bar Chart saved as {stacked_bar_output_path}")
plt.show()

# Pie Chart
furnishing_counts = data['Furnishing Status'].value_counts()
plt.figure(figsize=(8, 8))
furnishing_counts.plot(kind='pie', autopct='%1.1f%%', startangle=140,
colors=['#FF5733', '#33FF57', '#3357FF'])
plt.title("Pie Chart: Furnishing Status Distribution", fontsize=14,
fontweight='bold')
plt.ylabel("") # Hide y-axis label
pie_chart_output_path = '/content/Pie_Chart.png'
plt.savefig(pie_chart_output_path)
print(f"Pie Chart saved as {pie_chart_output_path}")
plt.show()

# Stacked Plot
plt.figure(figsize=(10, 6))
plt.stackplot(data['Size'], data['Rent'], alpha=0.7, labels=['Rent'],
colors=['#FF5733'])
plt.title("Stacked Plot: Rent vs. Size", fontsize=14,
fontweight='bold')
plt.xlabel("Size (sq ft)", fontsize=12)
plt.ylabel("Rent (INR)", fontsize=12)
plt.legend(loc='upper left', fontsize=10)
stacked_plot_output_path = '/content/Stacked_Plot.png'
plt.savefig(stacked_plot_output_path)

```

```

print(f"Stacked Plot saved as {stacked_plot_output_path}")
plt.show()

# Bar Chart
avg_rent_by_furnishing = data.groupby('Furnishing Status')
['Rent'].mean()
plt.figure(figsize=(10, 6))
colors = ['#FF5733', '#33FF57', '#3357FF'] # Different colors for bars
avg_rent_by_furnishing.plot(kind='bar', color=colors)
plt.title("Bar Chart: Average Rent by Furnishing Status", fontsize=14,
fontweight='bold')
plt.xlabel("Furnishing Status", fontsize=12)
plt.ylabel("Average Rent (INR)", fontsize=12)
bar_chart_output_path = '/content/Bar_Chart.png'
plt.savefig(bar_chart_output_path)
print(f"Bar Chart saved as {bar_chart_output_path}")
plt.show()

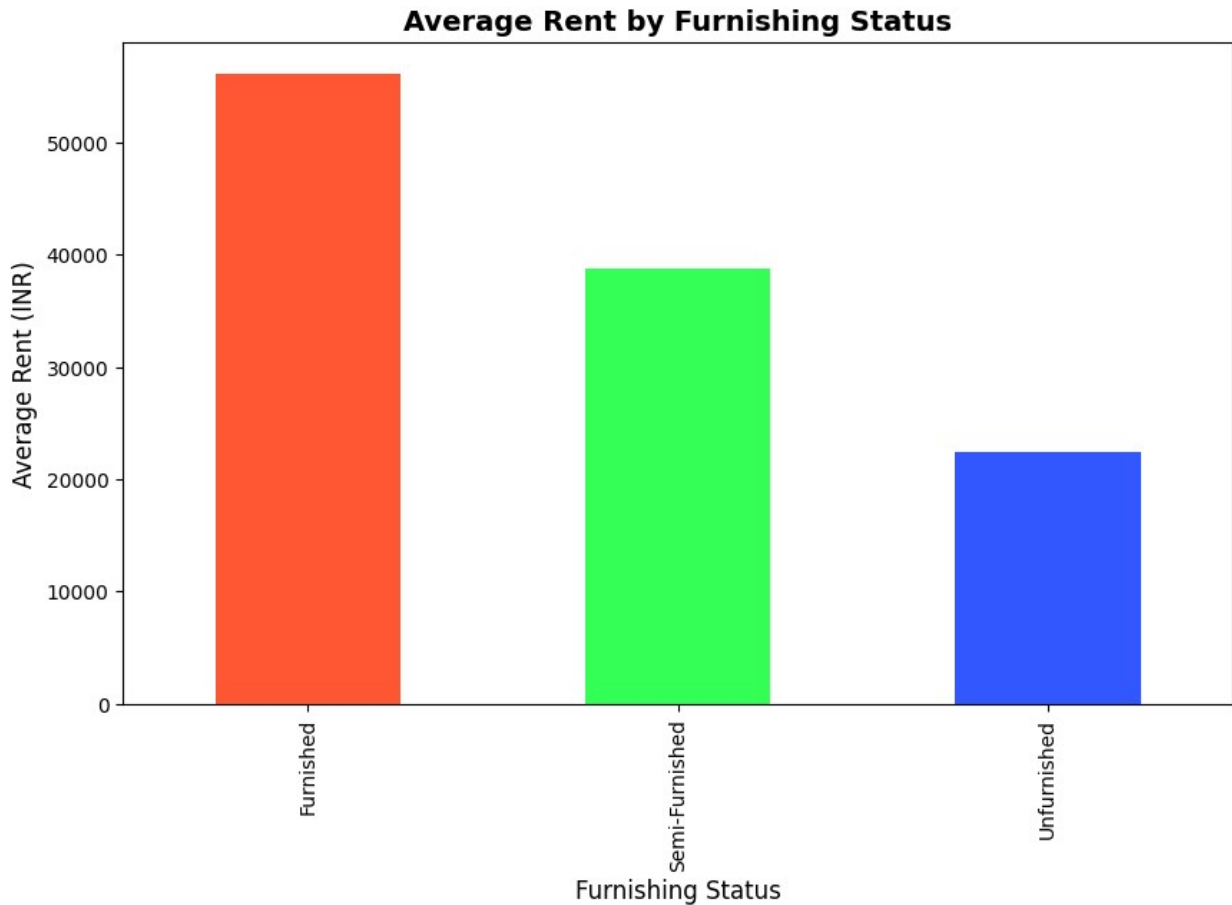
```

<IPython.core.display.HTML object>

```

Saving House_Rent_Dataset - House_Rent_Dataset(1).xlsx -
House_Rent_Dataset - House_Rent (1).zip to House_Rent_Dataset -
House_Rent_Dataset(1).xlsx - House_Rent_Dataset - House_Rent (1)
(2).zip
Bar Chart saved as /content/Average_Rent_by_Furnishing_Status.png

```



```
# Import necessary libraries
from google.colab import files
import pandas as pd
import matplotlib.pyplot as plt

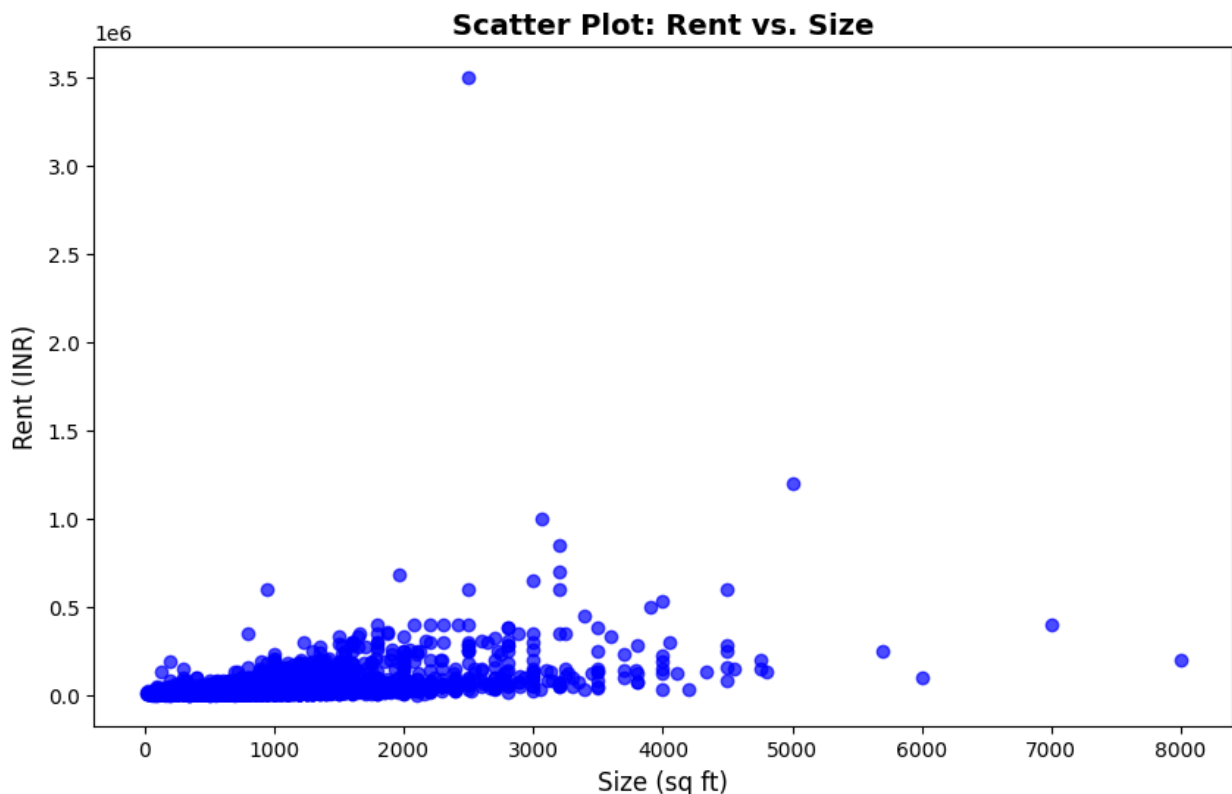
# Step 1: Upload the dataset
uploaded = files.upload()

# Step 2: Load the dataset
filename = list(uploaded.keys())[0] # Get the uploaded file name
data = pd.read_csv(filename) # Load the dataset (use pd.read_excel if
it's an Excel file)
# Scatter Plot
plt.figure(figsize=(10, 6))
plt.scatter(data['Size'], data['Rent'], color='blue', alpha=0.7)
plt.title("Scatter Plot: Rent vs. Size", fontsize=14,
fontweight='bold')
plt.xlabel("Size (sq ft)", fontsize=12)
plt.ylabel("Rent (INR)", fontsize=12)
scatter_output_path = '/content/Scatter_Plot.png'
plt.savefig(scatter_output_path)
```

```
print(f"Scatter Plot saved as {scatter_output_path}")
plt.show()
```

<IPython.core.display.HTML object>

Saving House_Rent_Dataset - House_Rent_Dataset(1).xlsx -
House_Rent_Dataset - House_Rent (2).csv to House_Rent_Dataset -
House_Rent_Dataset(1).xlsx - House_Rent_Dataset - House_Rent (2)
(1).csv
Scatter Plot saved as /content/Scatter_Plot.png



```
# Import necessary libraries
from google.colab import files
import pandas as pd
import matplotlib.pyplot as plt

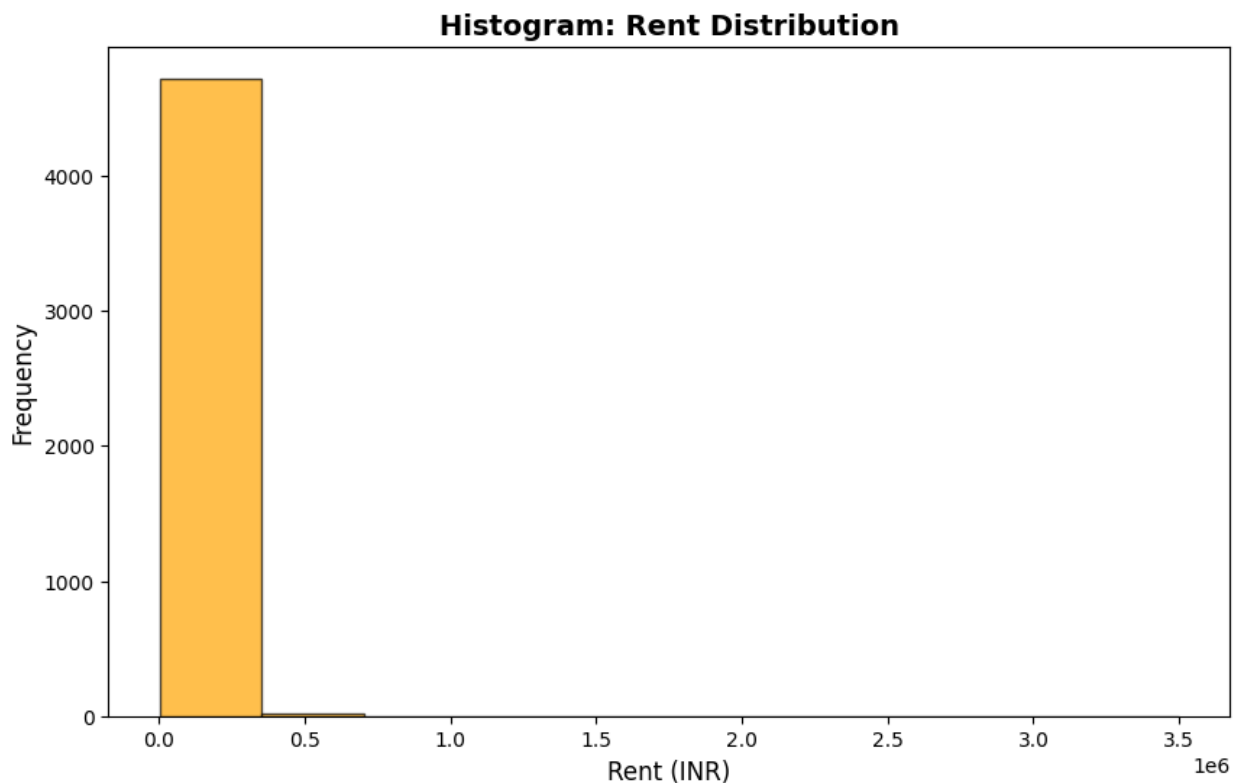
# Step 1: Upload the dataset
uploaded = files.upload()

# Step 2: Load the dataset
filename = list(uploaded.keys())[0] # Get the uploaded file name
data = pd.read_csv(filename) # Load the dataset (use pd.read_excel if
it's an Excel file)
# Histogram
plt.figure(figsize=(10, 6))
```

```
plt.hist(data['Rent'], bins=10, color='orange', alpha=0.7,
edgecolor='black')
plt.title("Histogram: Rent Distribution", fontsize=14,
fontweight='bold')
plt.xlabel("Rent (INR)", fontsize=12)
plt.ylabel("Frequency", fontsize=12)
histogram_output_path = '/content/Histogram.png'
plt.savefig(histogram_output_path)
print(f"Histogram saved as {histogram_output_path}")
plt.show()

<IPython.core.display.HTML object>

Saving House_Rent_Dataset - House_Rent_Dataset(1).xlsx -
House_Rent_Dataset - House_Rent (2).csv to House_Rent_Dataset -
House_Rent_Dataset(1).xlsx - House_Rent_Dataset - House_Rent (2)
(2).csv
Histogram saved as /content/Histogram.png
```



```
# Import necessary libraries
from google.colab import files
import pandas as pd
import matplotlib.pyplot as plt

# Step 1: Upload the dataset
```

```

uploaded = files.upload()

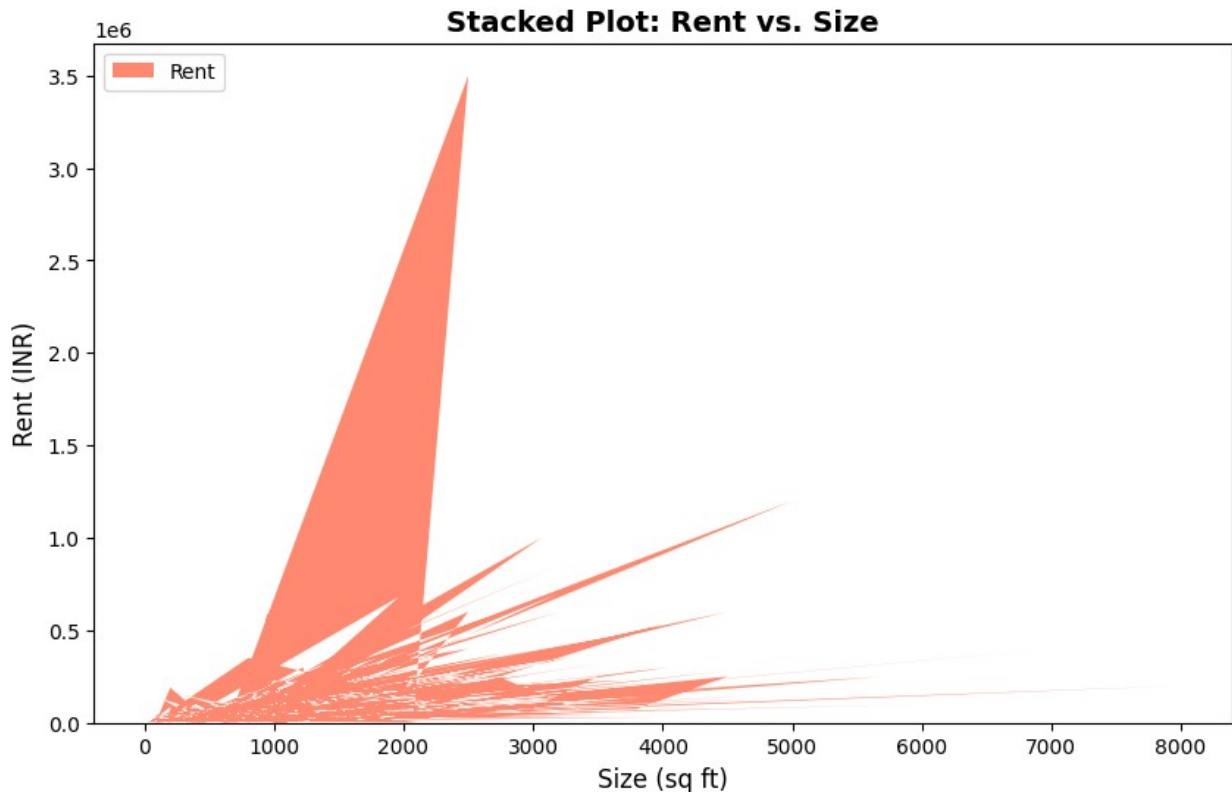
# Step 2: Load the dataset
filename = list(uploaded.keys())[0] # Get the uploaded file name
data = pd.read_csv(filename) # Load the dataset (use pd.read_excel if
it's an Excel file)

# Stacked Plot
plt.figure(figsize=(10, 6))
plt.stackplot(data['Size'], data['Rent'], alpha=0.7, labels=['Rent'],
colors=['#FF5733'])
plt.title("Stacked Plot: Rent vs. Size", fontsize=14,
fontweight='bold')
plt.xlabel("Size (sq ft)", fontsize=12)
plt.ylabel("Rent (INR)", fontsize=12)
plt.legend(loc='upper left', fontsize=10)
stacked_plot_output_path = '/content/Stacked_Plot.png'
plt.savefig(stacked_plot_output_path)
print(f"Stacked Plot saved as {stacked_plot_output_path}")
plt.show()

<IPython.core.display.HTML object>

Saving House_Rent_Dataset - House_Rent_Dataset(1).xlsx -
House_Rent_Dataset - House_Rent (2).csv to House_Rent_Dataset -
House_Rent_Dataset(1).xlsx - House_Rent_Dataset - House_Rent (2)
(5).csv
Stacked Plot saved as /content/Stacked_Plot.png

```

```
# Import necessary libraries
from google.colab import files
import pandas as pd
import matplotlib.pyplot as plt

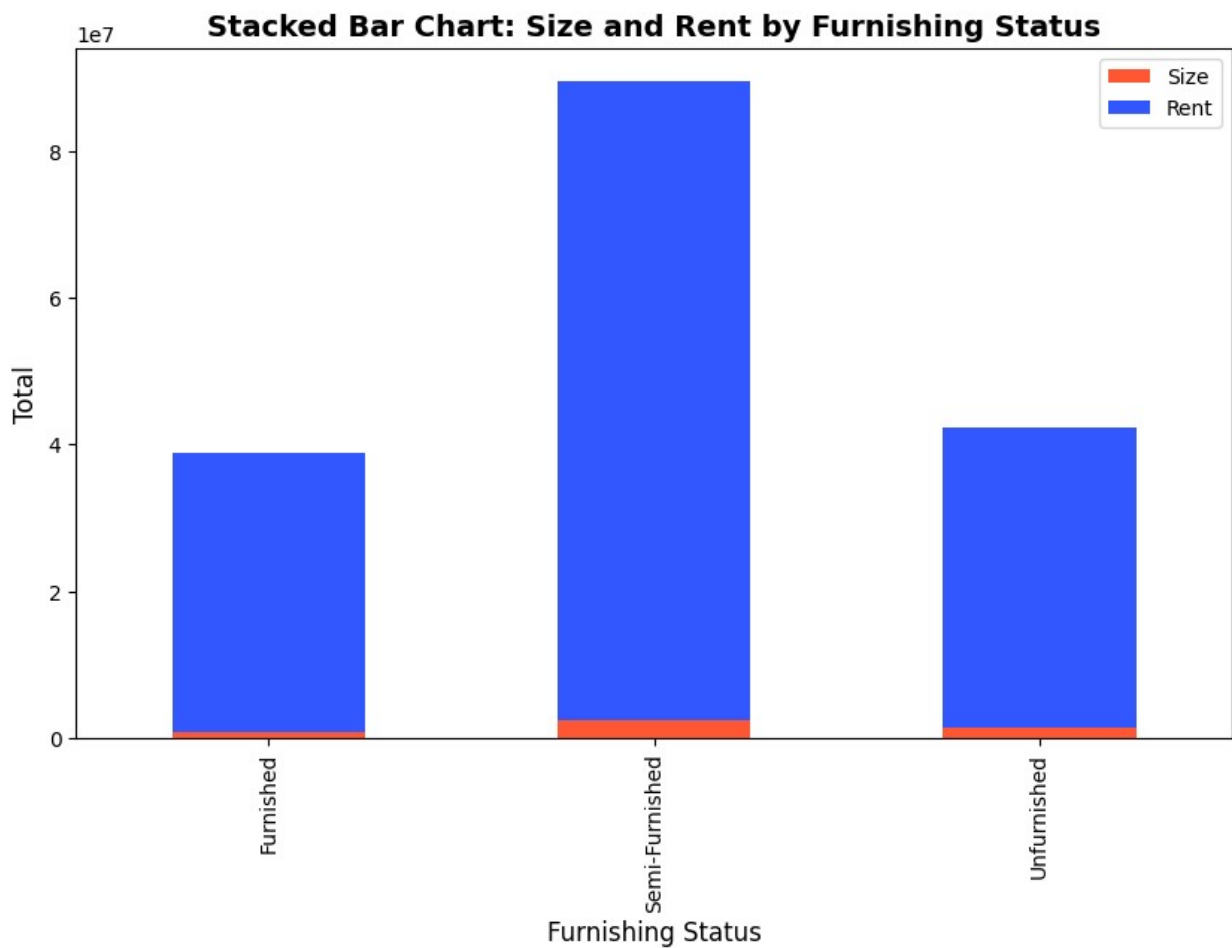
# Step 1: Upload the dataset
uploaded = files.upload()

# Step 2: Load the dataset
filename = list(uploaded.keys())[0] # Get the uploaded file name
data = pd.read_csv(filename) # Load the dataset (use pd.read_excel if
it's an Excel file)

# Stacked Bar Chart
grouped = data.groupby('Furnishing Status')[['Size', 'Rent']].sum()
grouped.plot(kind='bar', stacked=True, figsize=(10, 6),
color=['#FF5733', '#3357FF'])
plt.title("Stacked Bar Chart: Size and Rent by Furnishing Status",
fontsize=14, fontweight='bold')
plt.xlabel("Furnishing Status", fontsize=12)
plt.ylabel("Total", fontsize=12)
stacked_bar_output_path = '/content/Stacked_Bar_Chart.png'
plt.savefig(stacked_bar_output_path)
print(f"Stacked Bar Chart saved as {stacked_bar_output_path}")
plt.show()
```

<IPython.core.display.HTML object>

Saving House_Rent_Dataset - House_Rent_Dataset(1).xlsx -
House_Rent_Dataset - House_Rent (2).csv to House_Rent_Dataset -
House_Rent_Dataset(1).xlsx - House_Rent_Dataset - House_Rent (2)
(3).csv
Stacked Bar Chart saved as /content/Stacked_Bar_Chart.png



```
# Import necessary libraries
from google.colab import files
import pandas as pd
import matplotlib.pyplot as plt

# Step 1: Upload the dataset
uploaded = files.upload()

# Step 2: Load the dataset
filename = list(uploaded.keys())[0] # Get the uploaded file name
data = pd.read_csv(filename) # Load the dataset (use pd.read_excel if
it's an Excel file)
```

```

# Pie Chart
furnishing_counts = data['Furnishing Status'].value_counts()
plt.figure(figsize=(8, 8))
furnishing_counts.plot(kind='pie', autopct='%1.1f%%', startangle=140,
colors=['#FF5733', '#33FF57', '#3357FF'])
plt.title("Pie Chart: Furnishing Status Distribution", fontsize=14,
fontweight='bold')
plt.ylabel("") # Hide y-axis label
pie_chart_output_path = '/content/Pie_Chart.png'
plt.savefig(pie_chart_output_path)
print(f"Pie Chart saved as {pie_chart_output_path}")
plt.show()

```

<IPython.core.display.HTML object>

```

Saving House_Rent_Dataset - House_Rent_Dataset(1).xlsx -
House_Rent_Dataset - House_Rent (2).csv to House_Rent_Dataset -
House_Rent_Dataset(1).xlsx - House_Rent_Dataset - House_Rent (2)
(4).csv
Pie Chart saved as /content/Pie_Chart.png

```

Pie Chart: Furnishing Status Distribution

