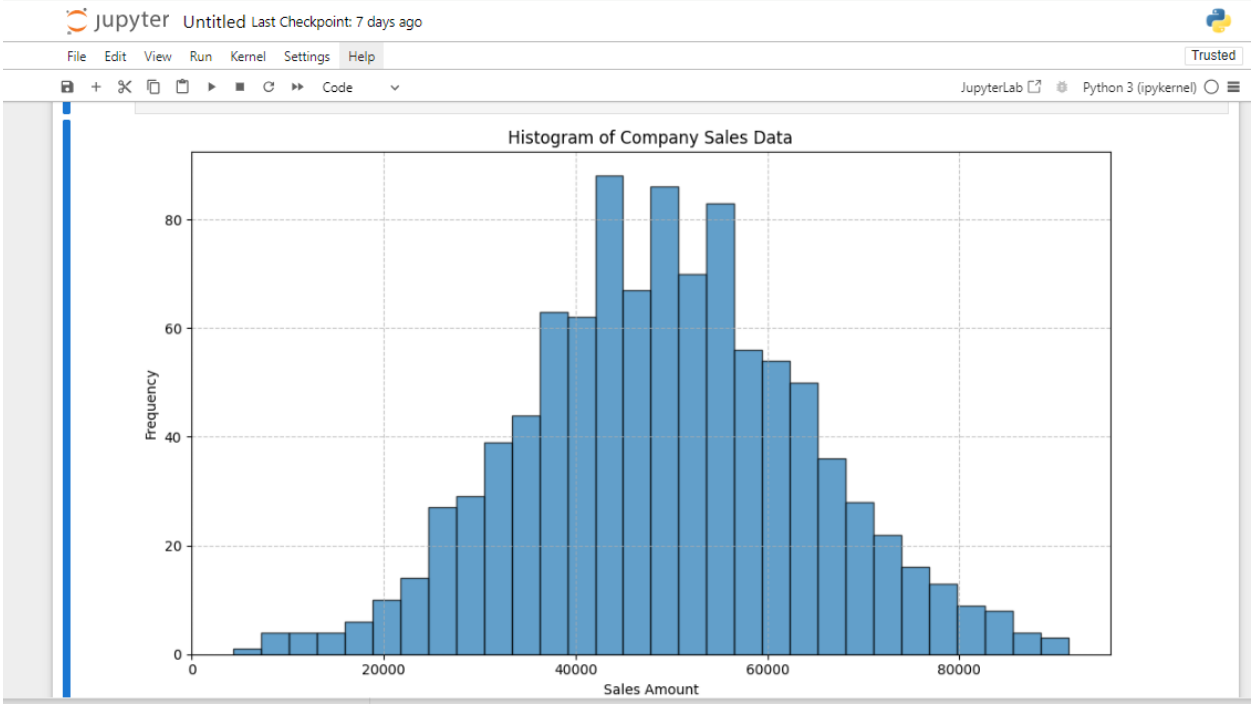
Introduction:

In Python, you can generate graphs such as line plots, bar charts, histograms, scatter plots, pie charts, and more. Below is an introduction to a basic Python code that uses Matplotlib to generate a graph. Environment jupyter notebook (anaconda).

- Green text python code.

Histogram:

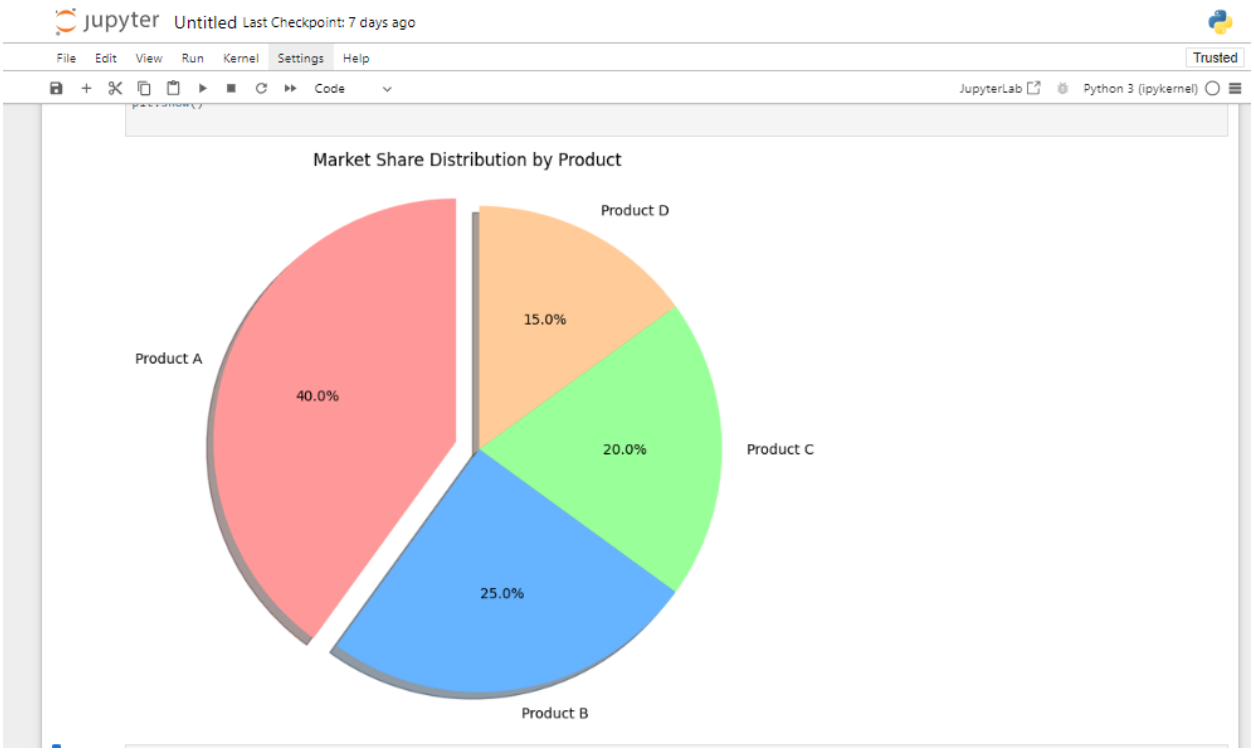
```
import matplotlib.pyplot as plt
import numpy as np
# Generate sample data
np.random.seed(0)  # For reproducibility
sales_data = np.random.normal(loc=50000, scale=15000, size=1000)  # Sample sales figures
# Create the histogram
plt.figure(figsize=(10, 6))
plt.hist(sales_data, bins=30, edgecolor='black', alpha=0.7)
# Customize the plot
plt.title('Histogram of Company Sales Data')
plt.xlabel('Sales Amount')
plt.ylabel('Frequency')
plt.grid(True, linestyle='--', alpha=0.7)
# Save the plot as an image file
plt.tight_layout()
plt.savefig('company_sales_histogram.png')
# Show the plot
plt.show()
```



Pie Graph:

```
import matplotlib.pyplot as plt

# Data for the pie chart

labels = ['Product A', 'Product B', 'Product C', 'Product D']

sizes = [40, 25, 20, 15]  # Market share percentages

colors = ['#ff9999', '#66b3ff', '#99ff99', '#ffcc99']  # Custom colors for each section

explode = (0.1, 0, 0, 0)  # "Explode" the first slice (Product A) to emphasize it

# Create the pie chart

plt.figure(figsize=(7, 7))

plt.pie(sizes, explode=explode, labels=labels, colors=colors, autopct='%1.1f%%', shadow=True, startangle=90)

# Equal aspect ratio ensures that the pie chart is drawn as a circle

plt.axis('equal')

# Title of the chart

plt.title('Market Share Distribution by Product')

# Save the pie chart as an image file

plt.savefig('market_share_pie_chart.png')
```

```
# Show the pie chart
plt.show()
```



Funnel:

```
import matplotlib.pyplot as plt
# Data for the funnel chart
stages = ['Initial Leads', 'Contacted', 'Qualified Leads', 'Proposals Sent', 'Closed Sales']
values = [10000, 6500, 3500, 2000, 800]
# Create a horizontal bar chart
plt.figure(figsize=(8, 6))
plt.barh(stages, values, color=['#ff9999','#66b3ff','#99ff99','#ffcc99','#ff6666'])
# Add labels to the bars
for i, v in enumerate(values):
    plt.text(v + 200, i, str(v), va='center', fontweight='bold')
# Customize the chart
plt.xlabel('Number of Customers')
plt.title('Sales Funnel')
```

```
# Invert the y-axis to have the largest bar on top (funnel shape)
plt.gca().invert_yaxis()
# Save the funnel chart as an image file
plt.tight_layout()
plt.savefig('sales_funnel_chart.png')
# Show the chart
plt.show()
```