Task Level :Beginner

Libraries: Seaborn and Pygal

1. Library Overview

Seaborn

- **Overview:**
  Seaborn is a Python data visualization library built on top of Matplotlib. It provides a high-level interface for creating attractive and informative statistical graphics with minimal code.
- **Key Features:**
  - Integrates seamlessly with **pandas DataFrames**.
  - Built-in support for **statistical visualizations** (e.g., regression lines, distribution plots).
  - **Automatic color themes** and improved aesthetics over Matplotlib.
  - Handles **categorical, relational, and distributional** data effectively.
- **Typical Use Cases:**
  - Data exploration and trend analysis.
  - Visualizing relationships between variables.
  - Creating publication-ready plots for analytics reports.

. Seaborn Examples

Seaborn is ideal for statistical and analytical visualizations. Below are two sample graphs using custom demo data:

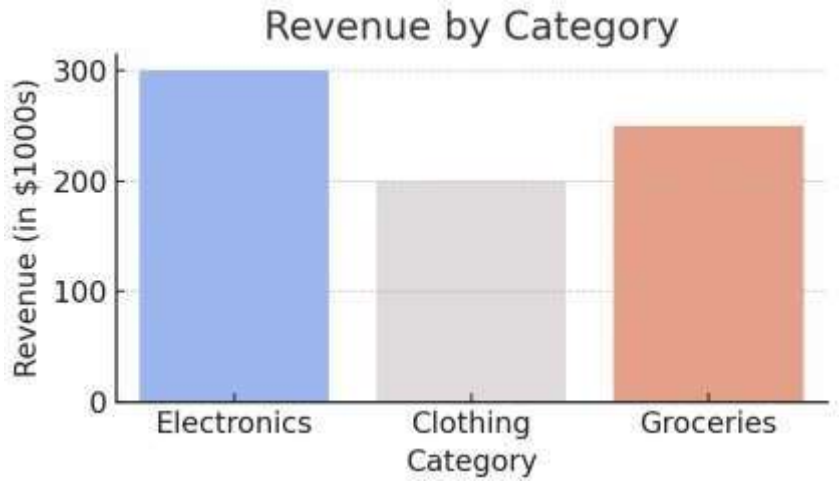
Example 1: Line Plot – Monthly Sales Trend

Code Example:

```python
import seaborn as sns
import matplotlib.pyplot as plt

months = ['Jan', 'Feb', 'Mar', 'Apr', 'May']
sales = [100, 150, 130, 180, 200]
sns.lineplot(x=months, y=sales, marker='o')
plt.title('Monthly Sales Trend')
plt.show()
```

Example 2: Bar Plot – Revenue by Category



Code Example:
```python
categories = ['Electronics', 'Clothing', 'Groceries']
revenue = [300, 200, 250]
sns.barplot(x=categories, y=revenue, palette='coolwarm')
plt.title('Revenue by Category')
plt.show()
```

Pygal

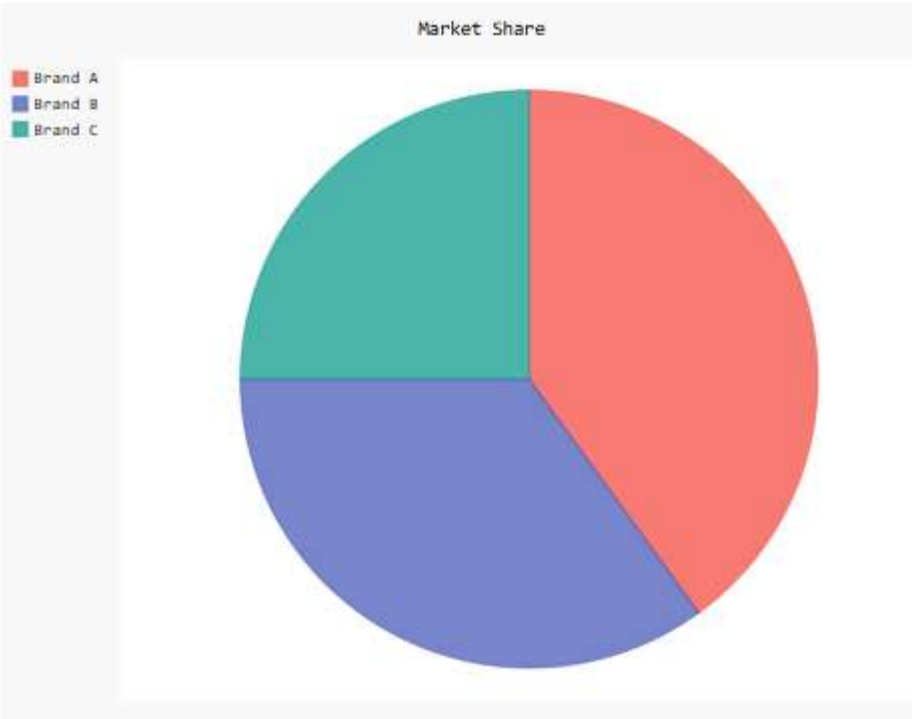
- **Overview:**
  Pygal is a Python library for creating **interactive SVG (Scalable Vector Graphics)** charts. Unlike Seaborn, which outputs static images, Pygal's charts can be zoomed, hovered, and embedded in web applications.
- **Key Features:**
  - Generates **interactive SVGs** viewable in browsers.
  - Easy customization with tooltips, legends, and styles.
  - Works well for **dashboards and web-based analytics**.
  - Lightweight and requires minimal dependencies.
- **Typical Use Cases:**
  - Interactive reports and dashboards.
  - Web-based visual analytics.
  - Lightweight chart embedding in applications.

  Example:

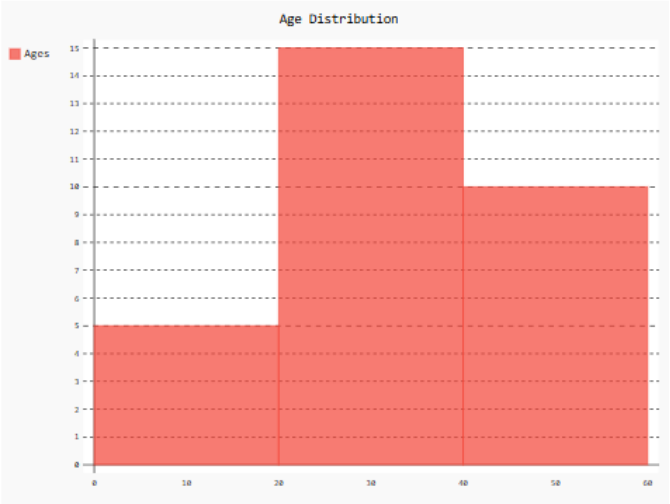
  import pygal; bar = pygal.Bar();
  bar.title = "Monthly Sales"; bar.add('Jan', 10); bar.add('Feb', 14);
  bar.add('Mar', 8); bar.render_in_browser()


```
import pygal
line = pygal.Line(); line.title = "Website Visitors";
line.x_labels = ['Mon','Tue','Wed','Thu','Fri'];
line.add('Visitors', [120, 150, 170, 160, 180]);
line.render_in_browser()
```

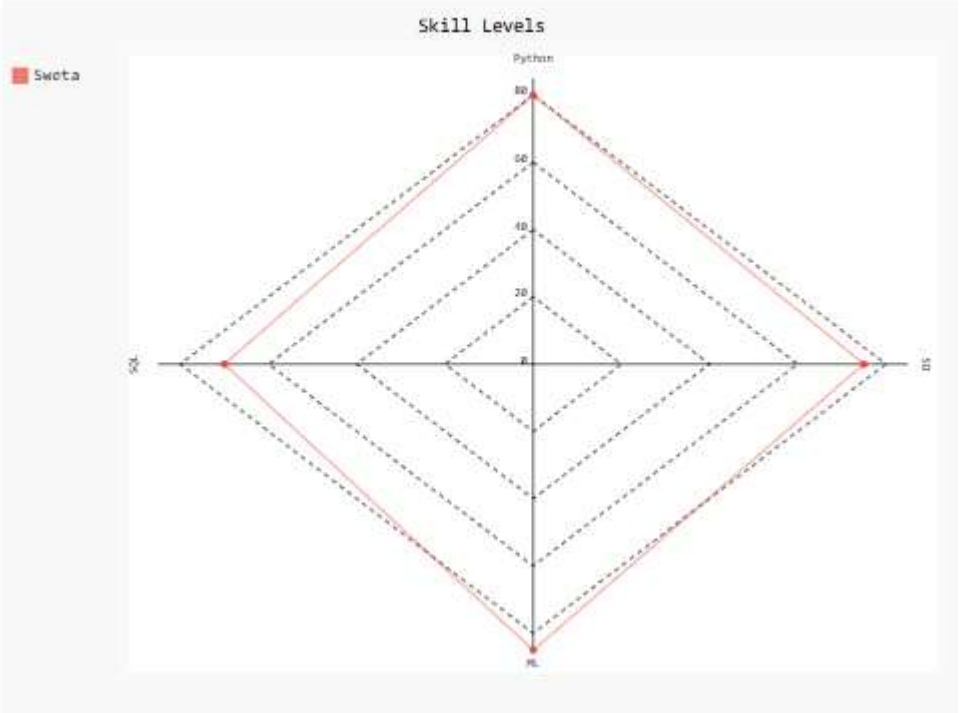
```
import pygal
pie = pygal.Pie(); pie.title = 'Market Share'; pie.add('Brand A', 40);
pie.add('Brand B', 35); pie.add('Brand C', 25);
pie.render_in_browser()
```




```
import pygal
hist = pygal.Histogram(); hist.title = 'Age Distribution'; hist.add('Ages',
[(5, 0, 20),
(15, 20, 40), (10, 40, 60)]);
hist.render_in_browser()
```



```
import pygal
radar = pygal.Radar(); radar.title = 'Skill Levels';
 radar.x_labels = ['Python','SQL','ML','DS']; radar.add('Sweta', [80, 70, 85, 75]);
     radar.render_in_browser()
```



Graph Types and Examples

A. Seaborn

Graph Type	Description	Use Case	Example Code
Line Plot (sns.lineplot)	Shows trends over time or continuous data.	Time-series analysis, stock prices, trends.	python import seaborn as sns; import matplotlib.pyplot as plt; df = sns.load_dataset("flights"); sns.lineplot(x="year", y="passengers", data=df); plt.show()
Bar Plot (sns.barplot)	Displays average values of categorical data.	Comparing sales by category, region, etc.	python import seaborn as sns; sns.barplot(x="day", y="total_bill", data=sns.load_dataset("tips"))
Scatter Plot (sns.scatterplot)	Shows relationship between two continuous variables.	Correlation, regression patterns.	python df = sns.load_dataset("iris"); sns.scatterplot(x="sepal_length", y="petal_length", hue="species", data=df)
Heatmap (sns.heatmap)	Represents data values in a matrix form.	Correlation analysis, confusion matrices.	python corr = sns.load_dataset("iris").corr(); sns.heatmap(corr, annot=True, cmap="coolwarm")
Box Plot (sns.boxplot)	Visualizes data distribution and outliers.	Detecting anomalies, comparing distributions.	python sns.boxplot(x="day", y="total_bill", data=sns.load_dataset("tips"))

B. Pygal

Graph Type	Description	Use Case	Example Code
Bar Chart	Displays categorical comparisons with vertical bars.	Comparing sales, votes, or counts.	python import pygal; bar = pygal.Bar(); bar.title = "Monthly Sales"; bar.add('Jan', 10); bar.add('Feb', 14); bar.add('Mar', 8); bar.render_in_browser()

Graph Type	Description	Use Case	Example Code
Line Chart	Shows trends over time.	Time-series data visualization.	python line = pygal.Line(); line.title = "Website Visitors"; line.x_labels = ['Mon','Tue','Wed','Thu','Fri']; line.add('Visitors', [120, 150, 170, 160, 180]); line.render_in_browser()
Pie Chart	Displays proportions of a whole.	Market share, category distribution.	python pie = pygal.Pie(); pie.title = 'Market Share'; pie.add('Brand A', 40); pie.add('Brand B', 35); pie.add('Brand C', 25); pie.render_in_browser()
Histogram	Represents frequency distribution of numerical data.	Data analysis and probability.	python hist = pygal.Histogram(); hist.title = 'Age Distribution'; hist.add('Ages', [(5, 0, 20), (15, 20, 40), (10, 40, 60)]); hist.render_in_browser()
Radar Chart	Compares multiple variables.	Skill evaluation, performance metrics.	python radar = pygal.Radar(); radar.title = 'Skill Levels'; radar.x_labels = ['Python','SQL','ML','DS']; radar.add('Sweta', [80, 70, 85, 75]); radar.render_in_browser()

1. Comparison

Feature	Seaborn	Pygal
Output Type	Static images (PNG, JPG)	Interactive SVG (browser viewable)
Ease of Use	Simple for statistical plots	Simple for basic interactive plots
Customization	Extensive via Matplotlib backend	Good, but limited style options
Performance	High (renders large datasets easily)	Lightweight for small-to-medium datasets
Interactivity	None (static)	High (hover, zoom, tooltips)
Best For	Data analysis, publications	Dashboards, web embedding