**Visualization ON Matplotlib & Pandas**

Visualization is a way of representing. Data visualization is representing of data in a graphical or pictorial format for better understanding of data. Visualization gives a good idea of data and the trends in it.

In python, data visualization has multiple libraries like matplotlib, seaborn, plotly, Pandas etc.

Let us dive deeper into matplotlib and Pandas.

* **Visualization Library Documentation: Matplotlib & Pandas:**

**Introduction:**

**Matplotlib:** Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. It provides control over every aspect of a figure, from individual plots to entire figure layout.

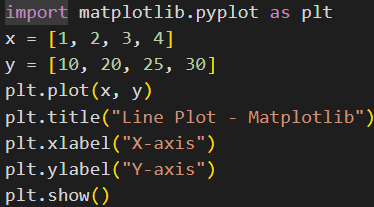
**Pandas:** Pandas is primarily a data manipulation library, but it comes with built-in plotting capabilities powered by Matplotlib. It allows quick plotting of Series and Data Frame objects.

**Graph Types:**

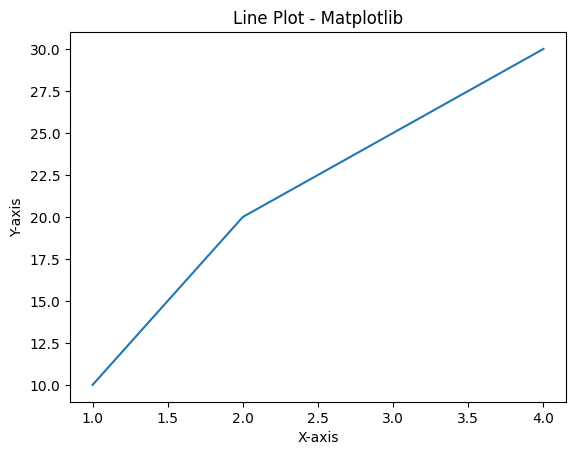
* **Line Plot:** A **line chart** is a type of graph used to display information that changes over time. It consists of a series of data points called "markers" connected by straight line segments. Line charts are commonly used to visualize trends, patterns, and comparisons in continuous data, such as stock prices, temperatures, or sales over months.

**Matplotlib**

Code snippet:

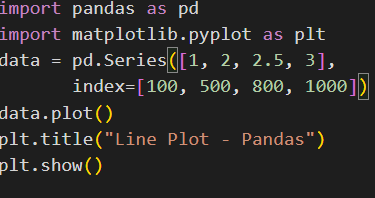


Output:

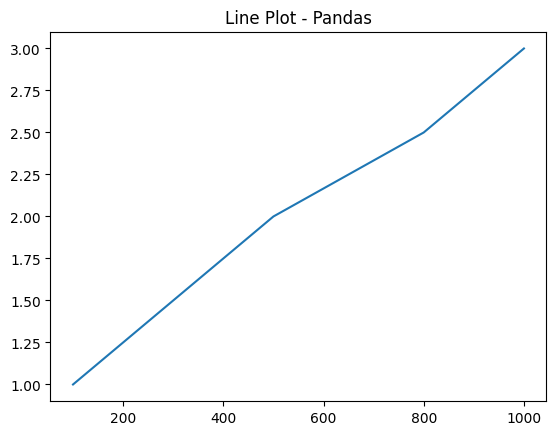


**Pandas**

Code snippet:



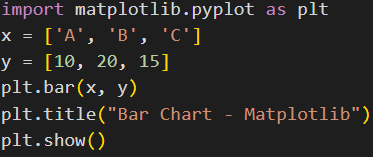
Output:



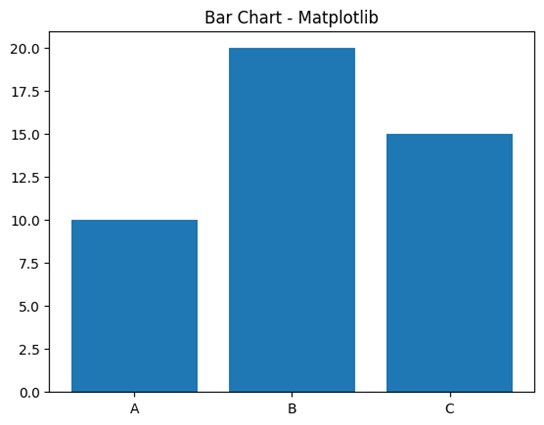
* **Bar Chart:** A **bar chart** is a type of graph that represents data using rectangular bars, where the length or height of each bar is proportional to the value it represents. Bar charts are commonly used to compare different categories or groups of data and can be displayed either vertically or horizontally.

**Matplotlib**

Code snippet:

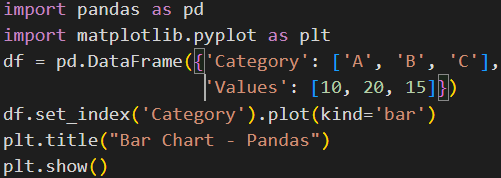


Output:

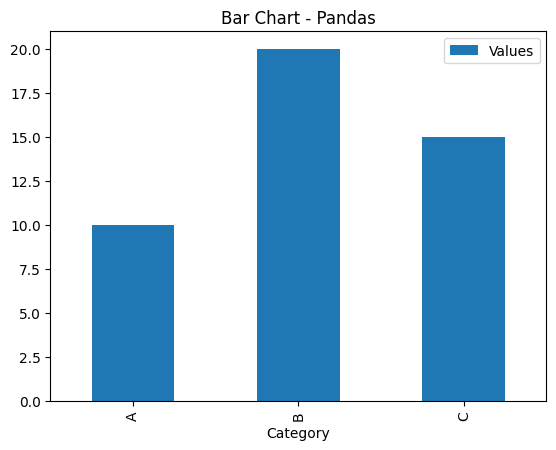


**Pandas**

Code snippet:



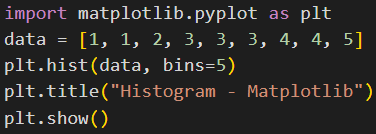
Output:



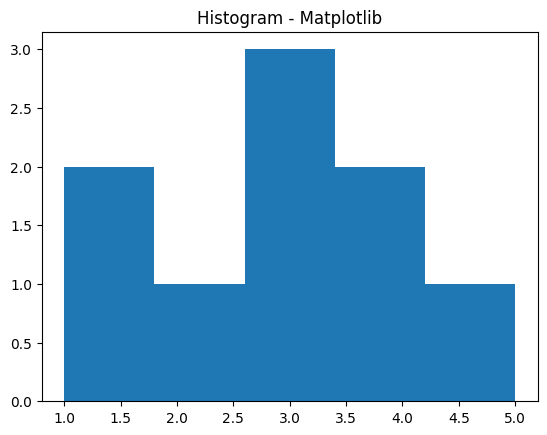
* **Histogram:** A histogram is a type of graph used to represent the distribution of numerical data. It consists of adjacent bars (without gaps) where each bar represents the frequency (or count) of data values within a specific range or interval, called a bin. Histograms are commonly used to show how data is spread out and to identify patterns like skewness, peaks, or gaps.

**Matplotlib**

Code snippet:

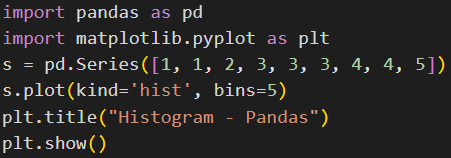


Output:

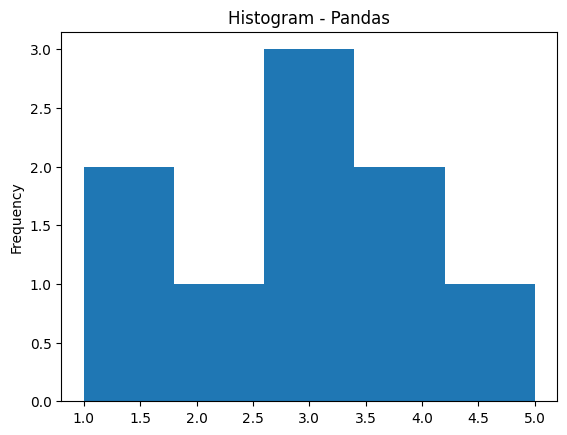


**Pandas**

Code snippet:



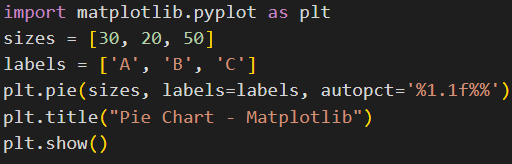
Output:



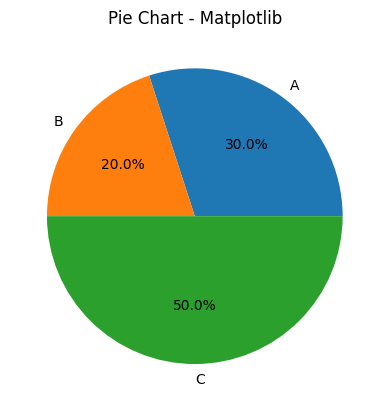
* **Pie Chart:** A **pie chart** is a circular graph divided into slices to illustrate numerical proportions. Each slice of the pie represents a category's contribution to the whole, with the size of each slice proportional to its percentage of the total. Pie charts are commonly used to show parts of a whole, such as market share or budget distribution.

**Matplotlib**

Code snippet:

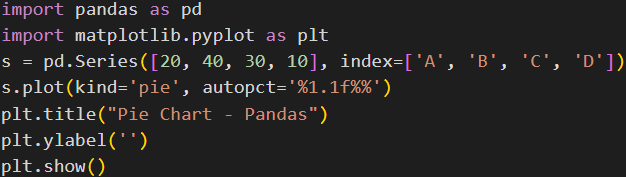


Output:

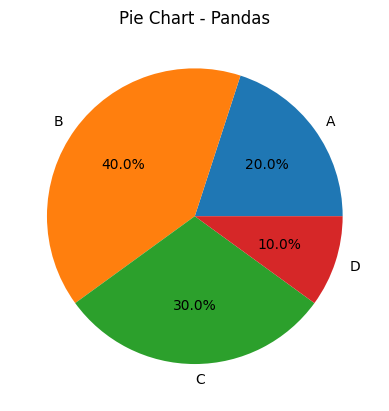


**Pandas**

Code snippet:



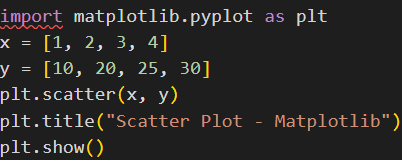
Output:



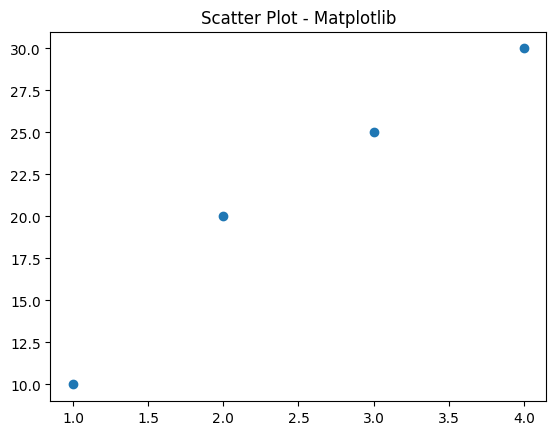
* **Scatter Plot:** A scatter plot is a type of graph that uses dots to represent the values of two different numerical variables. Each dot on the plot corresponds to one observation in the dataset, with its position determined by the values on the horizontal (x) and vertical (y) axes. Scatter plots are commonly used to identify relationships, patterns, or correlations between the two variables.

**Matplotlib**

Code snippet:

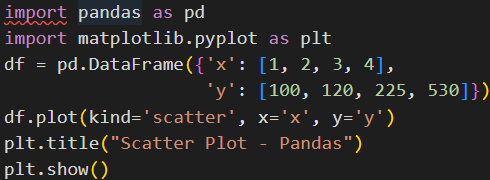


Output:

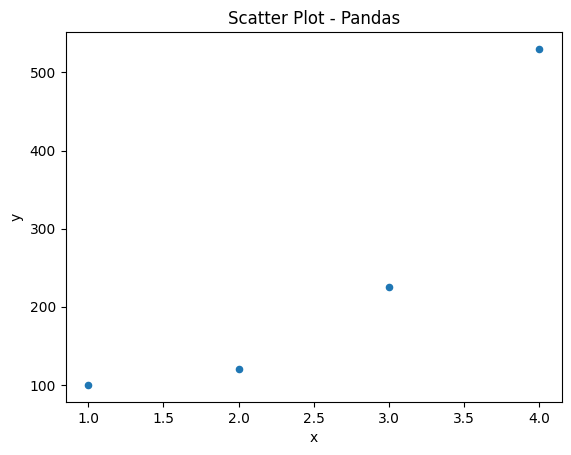


**Pandas**

Code snippet:



Output:



* **Comparison: Matplotlib vs Pandas**

|  |  |  |
| --- | --- | --- |
| **Feature** | **Matplotlib** | **Pandas** |
| EaseofUse | Detailed but verbose | Very simple for quick plots |
| Performance | High performance | Built on top of Matplotlib |
| Customization | Highly customizable | Limited customization |
| Interactivity | Limited (extendable with tools) | Very limited |
| Best Use Case | Custom/Complex Visualizations | Quick plotting of Data Frames |

* **Conclusion**
* Use Matplotlib when you need deep customization or creating multi-plot figures.
* Use Pandas for quick and easy plots directly from your data.