Types of Charts and Graphs

Bar Chart

Bar charts are one of the most common data visualizations. You can use them to quickly compare data across categories, highlight differences, show trends and outliers, and reveal historical highs and lows at a glance. Bar charts are especially effective when you have data that can be split into multiple categories.

Line Chart

The line chart, or line graph, connects several distinct data points, presenting them as one continuous evolution. Use line charts to view trends in data, usually over time (like stock price changes over five years or website page views for the month). The result is a simple, straightforward way to visualize changes in one value relative to another.

Pie Chart

Pie charts are powerful for adding detail to other visualizations. Alone, a [pie chart](https://www.tableau.com/data-insights/reference-library/visual-analytics/charts/pie-charts) doesn’t give the viewer a way to quickly and accurately compare information. Since the viewer has to create context on their own, key points from your data are missed. Instead of making a pie chart the focus of your dashboard, try using them to drill down on other visualizations.

Maps

Maps are a no-brainer for visualizing any kind of location information, whether it’s postal codes, state abbreviations, country names, or your own custom geocoding. If you have geographic information associated with your data, maps are a simple and compelling way to show how location correlates with trends in your data.

Density Maps

Density maps reveal patterns or relative concentrations that might otherwise be hidden due to an overlapping mark on a map—helping you identify locations with greater or fewer numbers of data points. Density maps are most effective when working with a data set containing many data points in a small geographic area.

Scatter Plot

Scatter plots are an effective way to investigate the relationship between different variables, showing if one variable is a good predictor of another, or if they tend to change independently. A scatter plot presents lots of distinct data points on a single chart. The chart can then be enhanced with analytics like cluster analysis or trend lines.

Gantt Chart

[Gantt charts](https://www.tableau.com/learn/articles/how-to/gantt-chart) display a project schedule or show changes in activity over time. A Gantt chart shows steps that need to be completed before others can begin, along with resource allocation.

Bubble Chart

Although bubbles aren’t technically their own type of visualization, using them as a technique adds detail to scatter plots or maps to show the relationship between three or more measures. Varying the size and color of circles creates visually compelling charts that present large volumes of data at once.

Treemap

Treemaps relate different segments of your data to the whole. As the name of the chart suggests, each rectangle in a treemap is subdivided into smaller rectangles, or sub-branches, based on its proportion to the whole. They make efficient use of space to show percent total for each category.

**Program 7: Perform Visualization techniques (types of maps - Bar, Colum, Line, Scatter, 3D Cubes etc)**

**Bar Chart:** Used to represent categorical data with rectangular bars. Each bar represents a category, and the length of the bar corresponds to the value of the category.

**Example:**

import matplotlib.pyplot as plt

categories = ['A', 'B', 'C', 'D']

values = [10, 20, 15, 25]

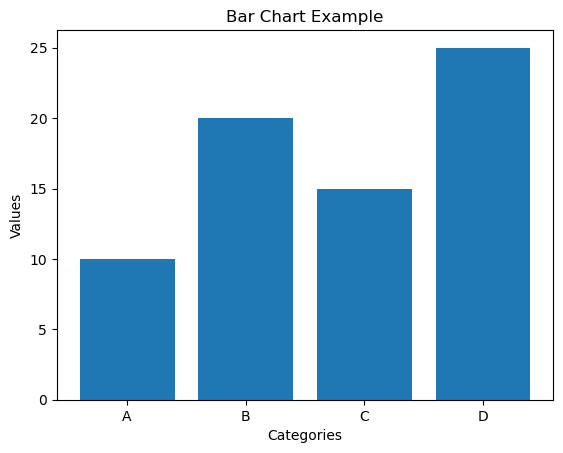
plt.bar(categories, values)

plt.xlabel('Categories')

plt.ylabel('Values')

plt.title('Bar Chart Example')

plt.show()



**Horizonatal bar graph**

import matplotlib.pyplot as plt

categories = ['Category A', 'Category B', 'Category C', 'Category D']

values = [25, 40, 30, 55]

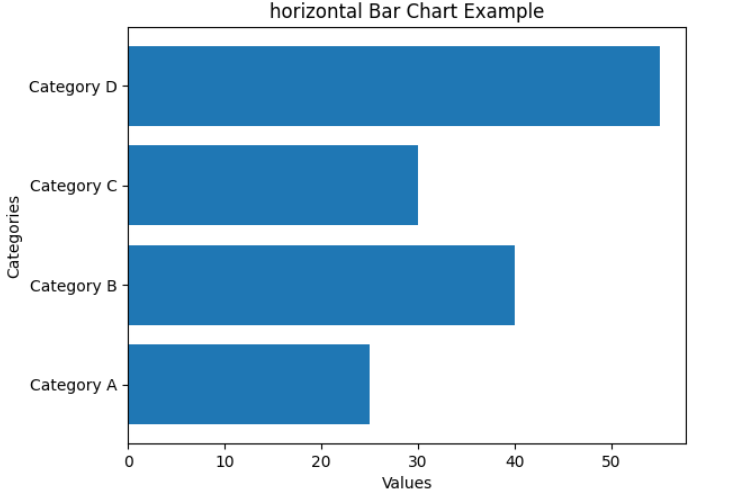
plt.barh(categories, values)

plt.xlabel('Values')

plt.ylabel('Categories')

plt.title('horizontal Bar Chart Example')

plt.show()



**Line Chart**: Used to represent data points connected by straight line segments. Typically used to show trends over time.

**Example:**

import matplotlib.pyplot as plt

x = [1, 2, 3, 4, 5]

y = [10, 15, 20, 25, 30]

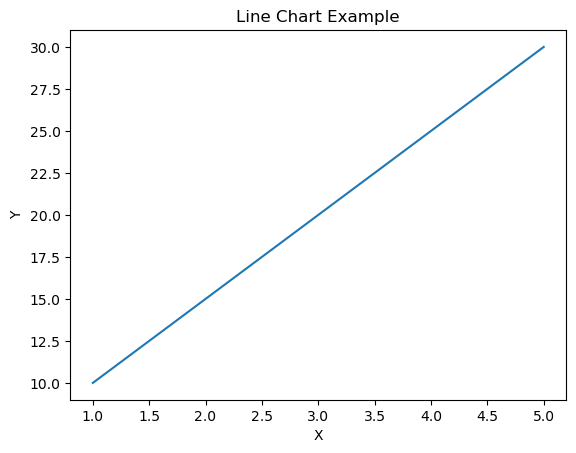
plt.plot(x, y)

plt.xlabel('X')

plt.ylabel('Y')

plt.title('Line Chart Example')

plt.show()



Bubble chart:

import matplotlib.pyplot as plt

import numpy as np

x = np.random.rand(10)

y = np.random.rand(10)

sizes = np.random.rand(10) \* 1000

colors = np.random.rand(10)

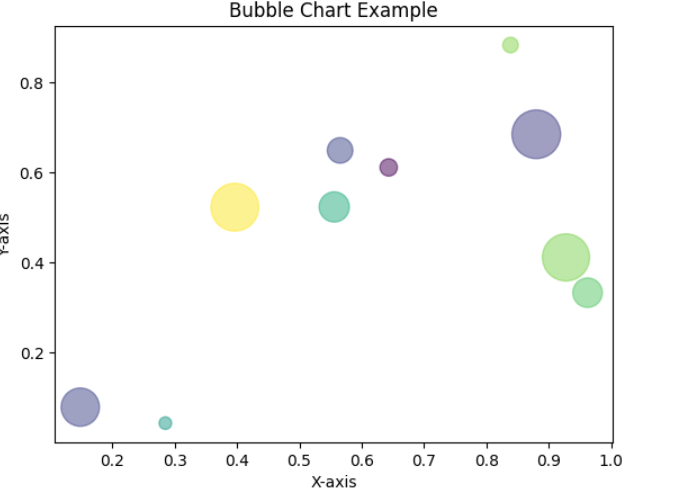
plt.scatter(x, y, s=sizes, c=colors, alpha=0.5)

plt.xlabel("X-axis")

plt.ylabel("Y-axis")

plt.title("Bubble Chart Example")

plt.show()



Scatter plot

import matplotlib.pyplot as plt

x = [5,7,8,7,2,17,2,9,4,11,12,9,6]

y = [99,86,87,88,111,86,103,87,94,78,77,85,86]

plt.scatter(x, y)

plt.show()

