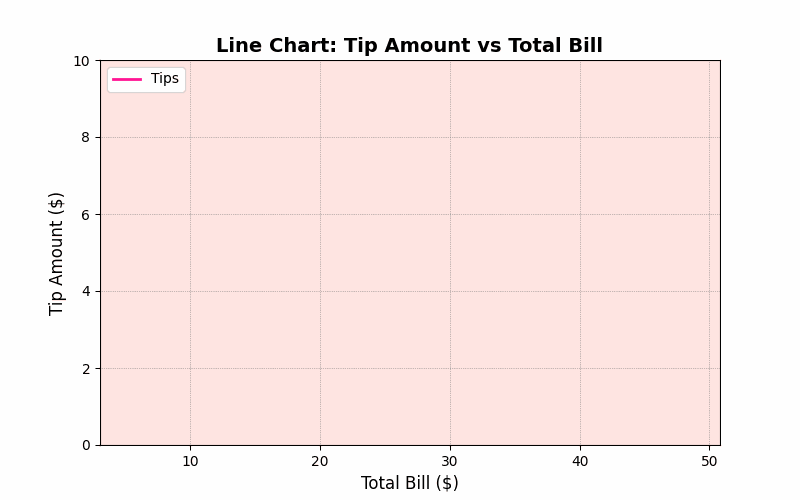
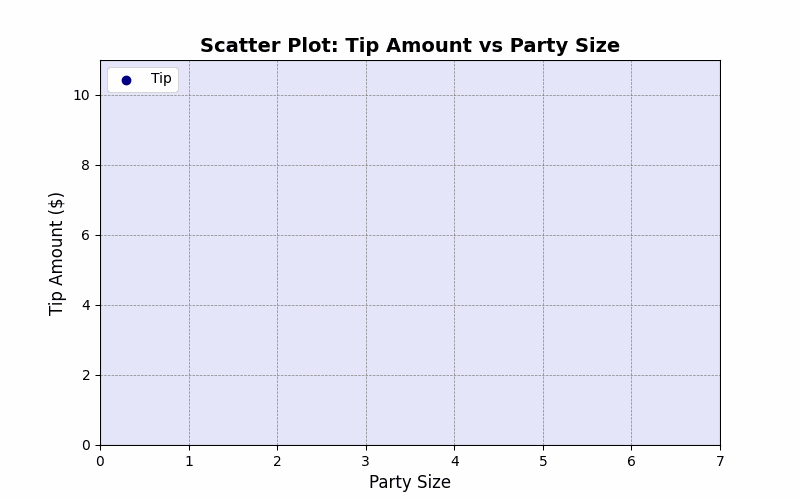
**Task-1:** **Data Visualization**

1. **Description**

Create animated plots using Matplotlib.

1. **Output  
     
     
     
     
   **
2. **Algorithm Used in Task**

In this task, we created animated plots, specifically Line Charts and Scatter Plots, using the Matplotlib library in Python. The key focus was to animate the data in a way that allows dynamic visualization, updating the plot progressively as new data points are added.

**Explanation of Libraries and Their Usage:**

1. **Pandas (import pandas as pd)**
   * Used for data manipulation and analysis.
   * Reads the CSV file containing the data and organizes it into a DataFrame for further operations.
2. **Matplotlib (import matplotlib.pyplot as plt)**
   * Primary library for creating static, animated, and interactive plots.
3. **Matplotlib Animation (from matplotlib.animation import FuncAnimation, PillowWriter)**
   * Enables the creation of animated visualizations.
   * FuncAnimation is used to update the plot frame-by-frame.
   * PillowWriter is used to save animations as GIFs.

**Approach:**

1. **Data Preprocessing:**
   * The data is loaded using Pandas and sorted based on relevant columns.
2. **Data Visualization:**
   * **Line Chart:** Shows how tip amounts vary with total bill values.
   * **Scatter Plot:** Illustrates the relationship between tip amounts and party size.
3. **Animation:**
   * Gradually plots points or lines by iterating through frames.
   * Creates engaging visualizations saved as animated GIFs.

**Algorithm:**

**Line Chart: Tip Amount vs Total Bill**

1. Load the data from tips.csv using Pandas.
2. Sort the data by total\_bill to ensure smooth animation.
3. Extract total\_bill and tip columns for x and y values.
4. Create a plot with:
   * Title, labels, axis limits, and gridlines for clarity.
   * A line object initialized as empty.
5. Define an update() function to progressively add data points to the line.
6. Use FuncAnimation to animate the plot by calling update() frame-by-frame.
7. Save the animation as line\_chart.gif.

**Scatter Plot: Tip Amount vs Party Size**

1. Sort the data by size to organize the plot logically.
2. Extract size and tip columns for x and y values.
3. Create a scatter plot with:
   * Title, labels, axis limits, and gridlines.
   * A scatter object initialized with empty data.
4. Define an update() function to incrementally add points to the scatter plot.
5. Use FuncAnimation to animate the scatter plot by calling update() frame-by-frame.
6. Save the animation as scatter\_plot.gif.