What is Matplotlib:

Matplotlib is a popular Python library for creating static, animated, and interactive visualizations. It is widely used for plotting graphs, charts, and other visual representations of data.

Key Features:

- 2D and 3D plotting: Supports line charts, bar charts, scatter plots, histograms, pie charts, and more.
- Highly customizable: Allows users to modify colors, labels, legends, and styles.
- Integration with NumPy and Pandas: Works seamlessly with these libraries for data manipulation and analysis.
- Object-oriented and state-based interfaces: Supports both a scripting (MATLAB-like) interface via pyplot and an object-oriented approach.
- Exporting options: Can save plots in various formats like PNG, PDF, SVG, and EPS.

In [1]:

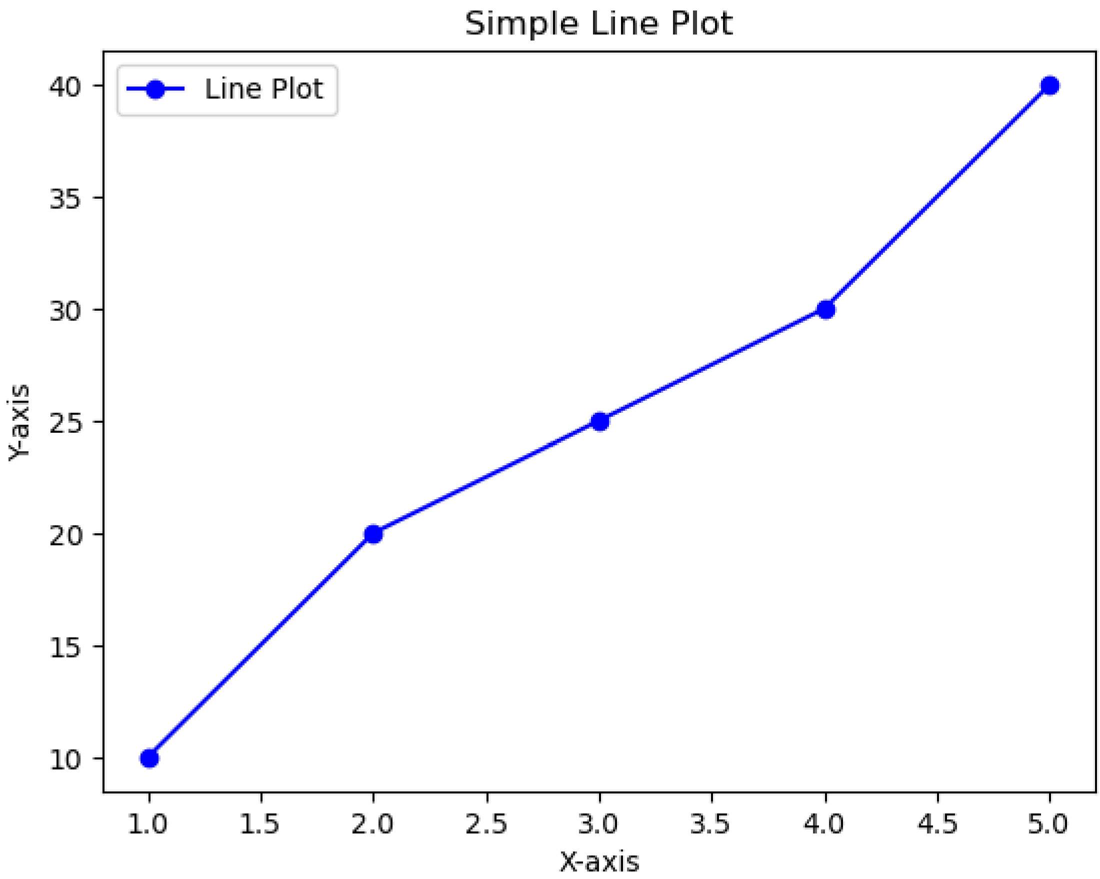
```
import matplotlib.pyplot as plt

# Sample data
x = [1, 2, 3, 4, 5]
y = [10, 20, 25, 30, 40]

# Create a simple line plot
plt.plot(x, y, marker='o', linestyle='-', color='b', label='Line Plot')

# Adding labels and title
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.title('Simple Line Plot')
plt.legend()

# Show the plot
plt.show()
```



What is Seaborn :

It looks like you might be referring to Seaborn, a Python data visualization library built on top of Matplotlib. Seaborn makes it easier to create aesthetically pleasing and informative statistical graphics.

Key Features of Seaborn:

- Beautiful default styles: Automatically applies styling that makes plots visually appealing.
- Statistical visualization: Easily creates histograms, violin plots, box plots, scatter plots, and more.
- Integration with Pandas: Works seamlessly with Pandas DataFrames.
- High-level API: Simplifies complex visualizations that would require more effort in Matplotlib.
- Built-in datasets: Provides example datasets for practice and experimentation.

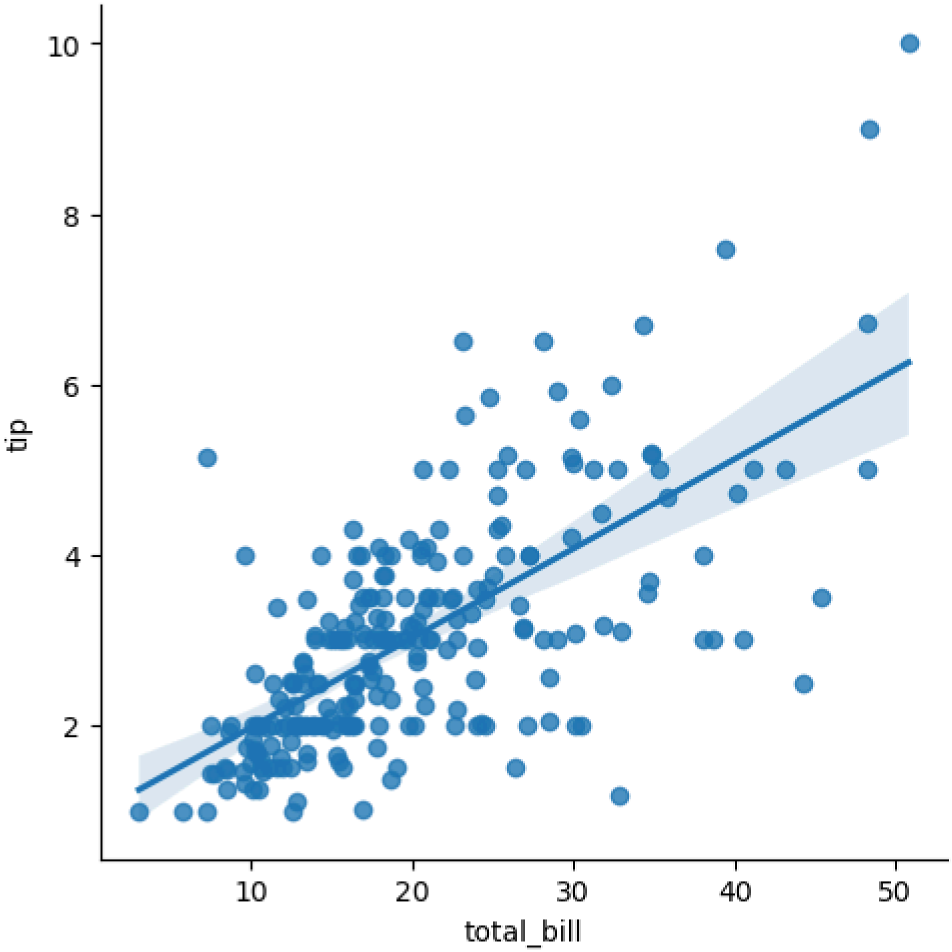
In [2]:

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

# Load a sample dataset
tips = sns.load_dataset("tips")

# Create a scatter plot with regression line
sns.lmplot(x="total_bill", y="tip", data=tips)
```

```
# Show the plot
plt.show()
```



various plot with interpretations :

1. Scatter Plot

Used to show the relationship between two numerical variables.

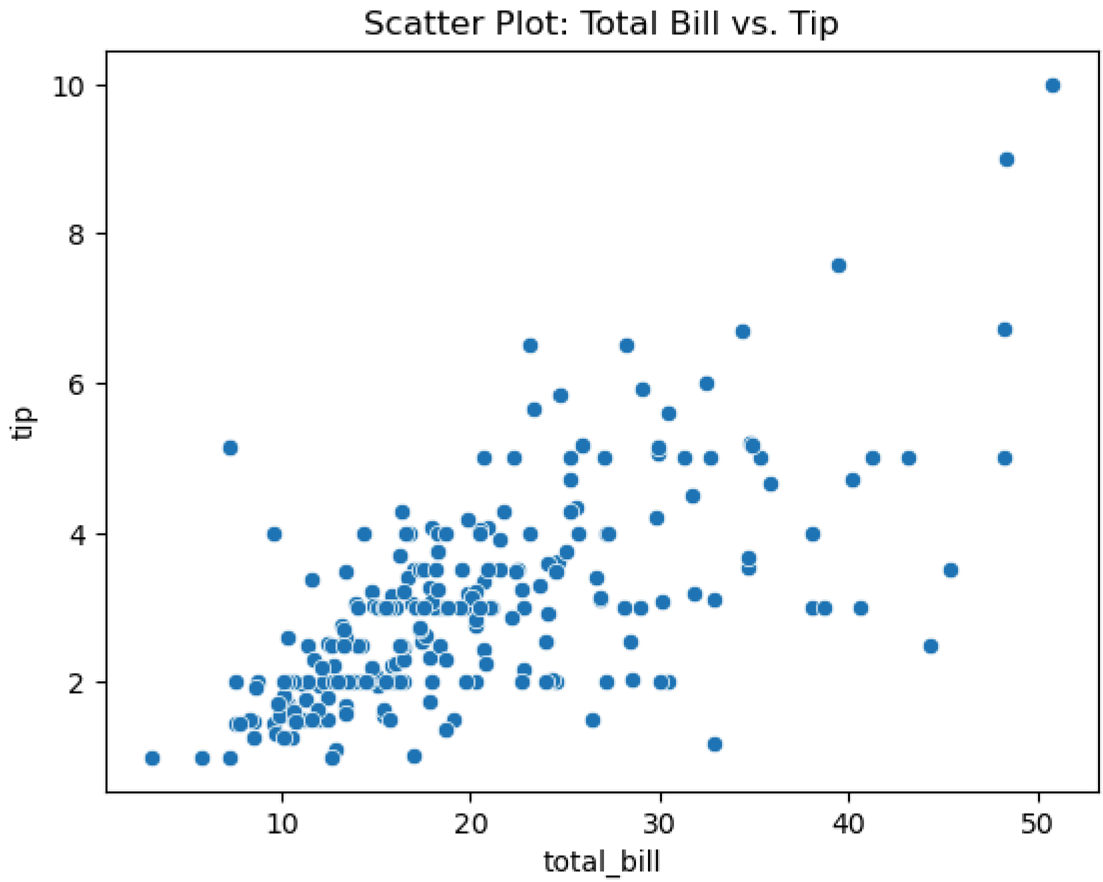
In [3]:

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

# Load dataset
tips = sns.load_dataset("tips")

# Scatter plot
sns.scatterplot(x="total_bill", y="tip", data=tips)

# Show the plot
plt.title("Scatter Plot: Total Bill vs. Tip")
plt.show()
```



Interpretation:

- Each point represents a restaurant bill and the corresponding tip.
- A positive correlation is visible: as the total bill increases, the tip also increases.
- Some points are outliers where the tip is either very high or very low for a given bill.

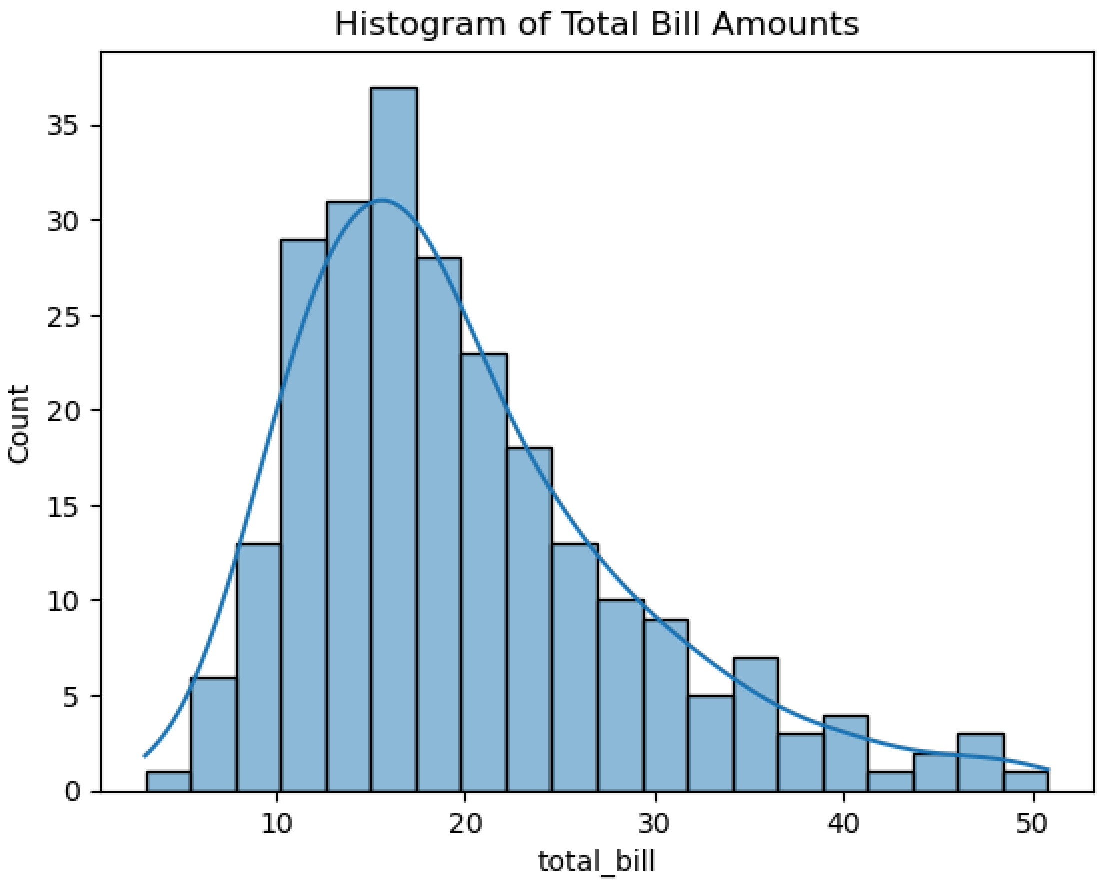
2. Histogram

Used to visualize the distribution of a single numerical variable.

In [5]:

```python
sns.histplot(tips["total_bill"], bins=20, kde=True)
plt.title("Histogram of Total Bill Amounts")
plt.show()
```

```
C:\Users\saksh\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarnin
g: use_inf_as_na option is deprecated and will be removed in a future version. Co
nvert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
```



Interpretation:

- The majority of the total bills are between $10and20$.
- The distribution is right-skewed, meaning some customers spent significantly more.
- The KDE (Kernel Density Estimation) curve shows the probability density.

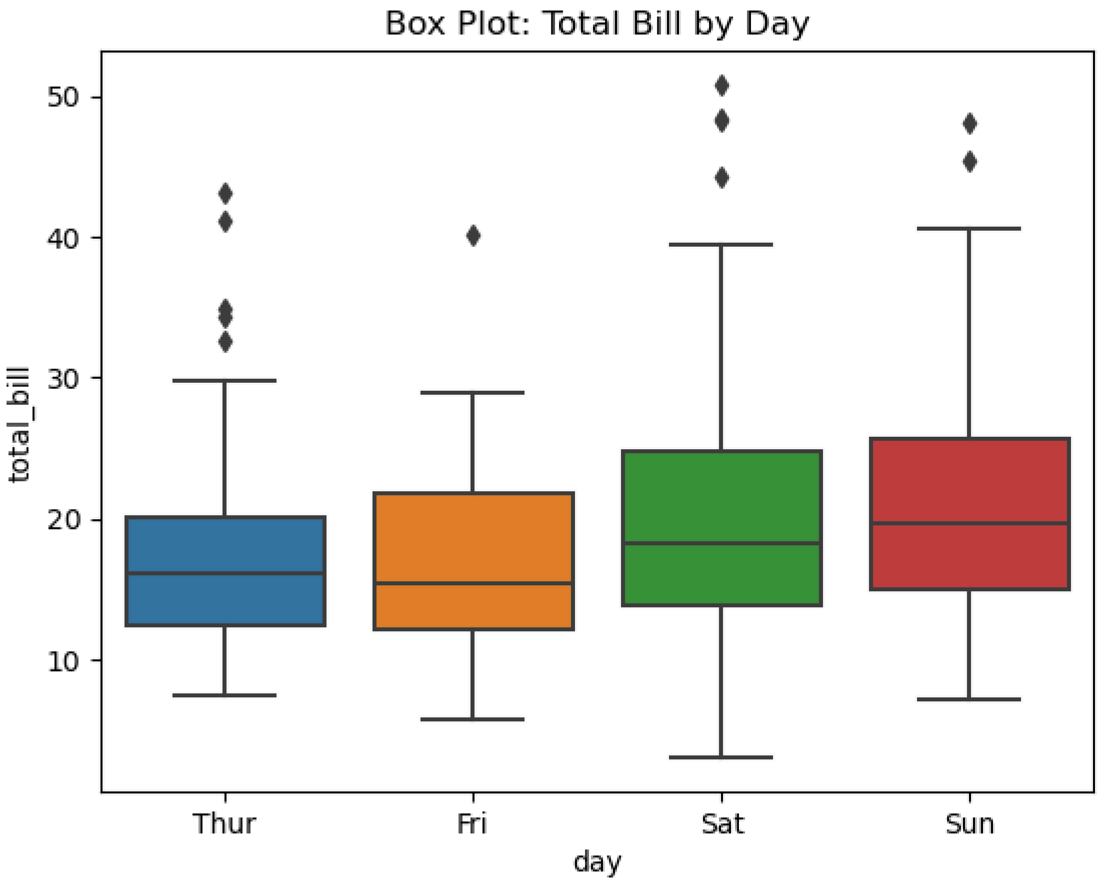
3. Box Plot

Used to show the distribution, outliers, and spread of numerical dat

In [8]:

```
sns.boxplot(x="day", y="total_bill", data=tips)
plt.title("Box Plot: Total Bill by Day")
plt.show()
```

```
C:\Users\saksh\anaconda3\Lib\site-packages\seaborn\categorical.py:641: FutureWarn
ing: The default of observed=False is deprecated and will be changed to True in a
future version of pandas. Pass observed=False to retain current behavior or obser
ved=True to adopt the future default and silence this warning.
  grouped_vals = vals.groupby(grouper)
```



Interpretation:

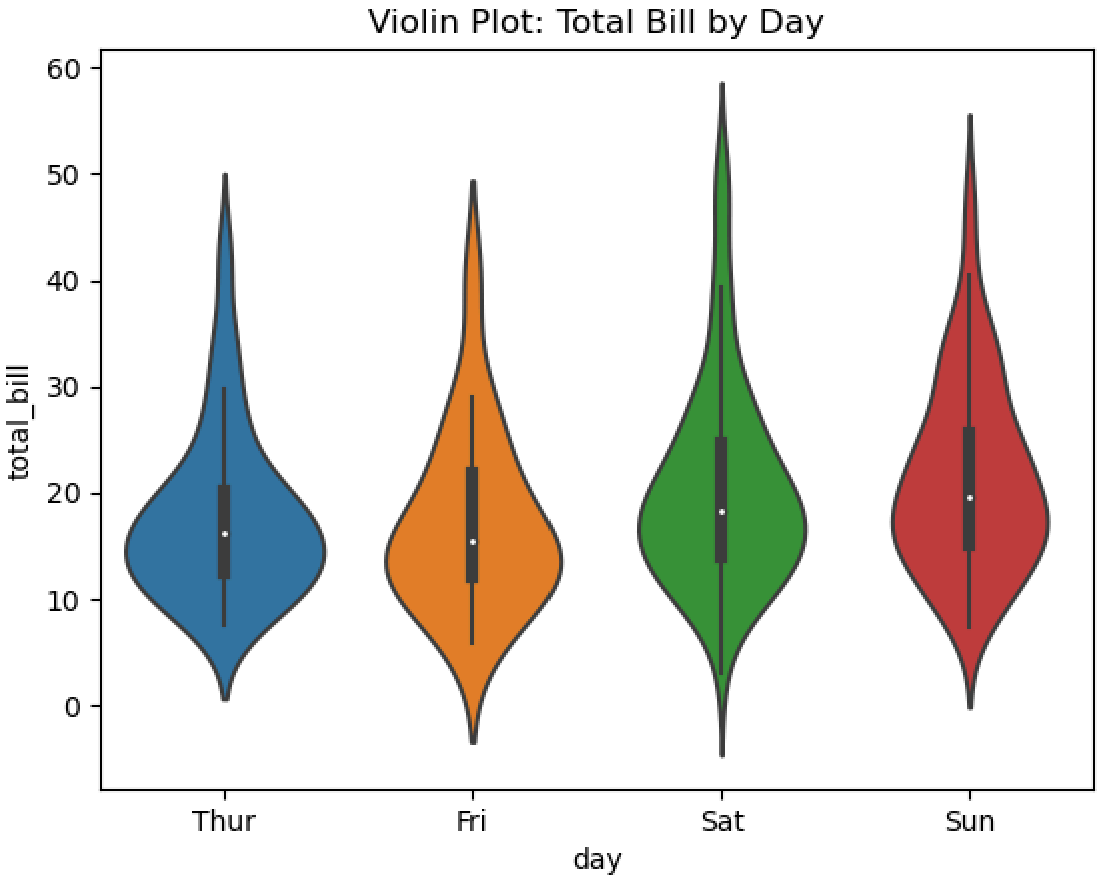
- Each box represents the interquartile range (IQR) (middle 50% of data).
- The line inside the box is the median (middle value).
- Points outside the whiskers are outliers, indicating unusually high bills.
- The median bill amount differs across days.

4. Violin Plot

Similar to a box plot but also shows the density distribution.

In [17]:
```
sns.violinplot(x="day", y="total_bill", data=tips)
plt.title("Violin Plot: Total Bill by Day")
plt.show()
```

```
C:\Users\saksh\anaconda3\Lib\site-packages\seaborn\categorical.py:641: FutureWarn
ing: The default of observed=False is deprecated and will be changed to True in a
future version of pandas. Pass observed=False to retain current behavior or obser
ved=True to adopt the future default and silence this warning.
  grouped_vals = vals.groupby(grouper)
```



Interpretation:

- The thickness of the violin shows where most data points are concentrated.
- The plot combines a box plot and a KDE (density plot).
- Saturdays seem to have the widest range of bill amounts.

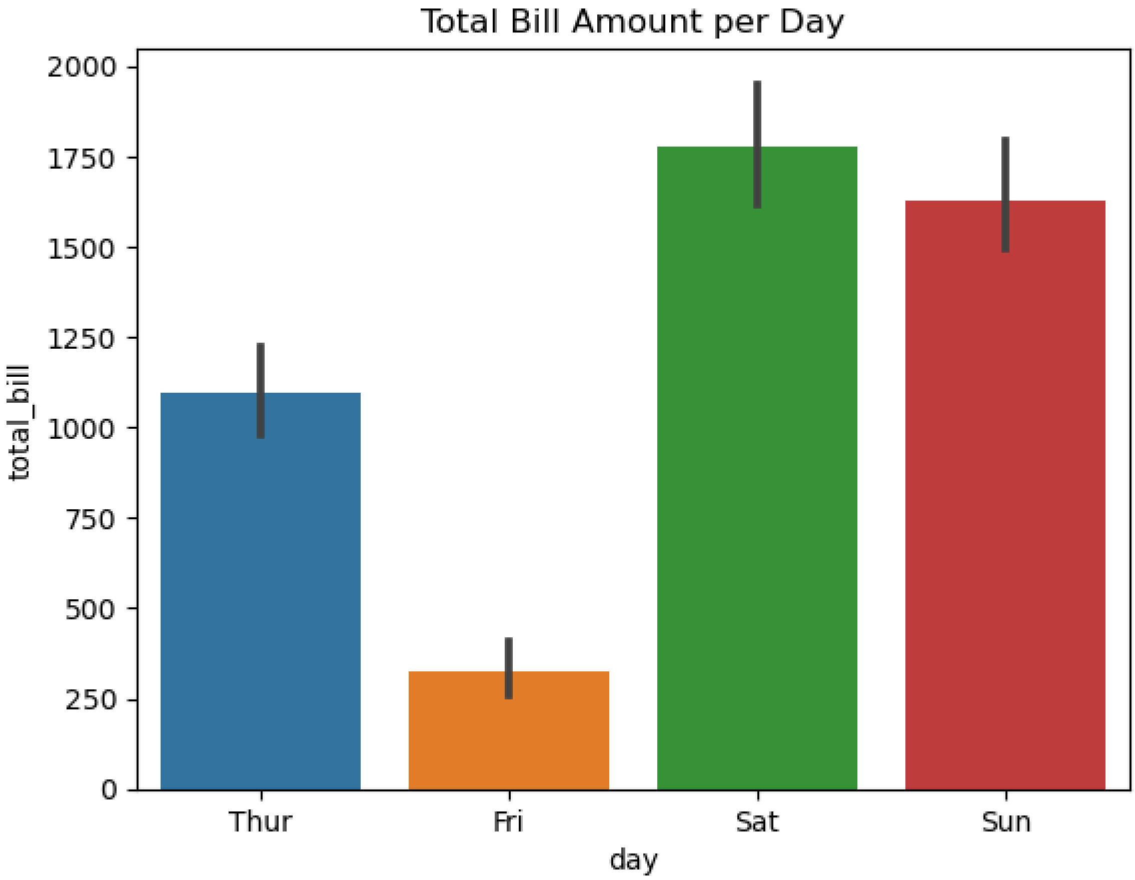
5. Bar Plot

Used for comparing categorical data.

In [10]:

```
sns.barplot(x="day", y="total_bill", data=tips, estimator=sum)
plt.title("Total Bill Amount per Day")
plt.show()
```

```
C:\Users\saksh\anaconda3\Lib\site-packages\seaborn\categorical.py:641: FutureWarn
ing: The default of observed=False is deprecated and will be changed to True in a
future version of pandas. Pass observed=False to retain current behavior or obser
ved=True to adopt the future default and silence this warning.
  grouped_vals = vals.groupby(grouper)
```



Interpretation:

- Saturday and Sunday have the highest total bill amounts.
- This suggests that more people dine out on weekends.

6. Heatmap

Used to visualize correlations or patterns in a dataset.

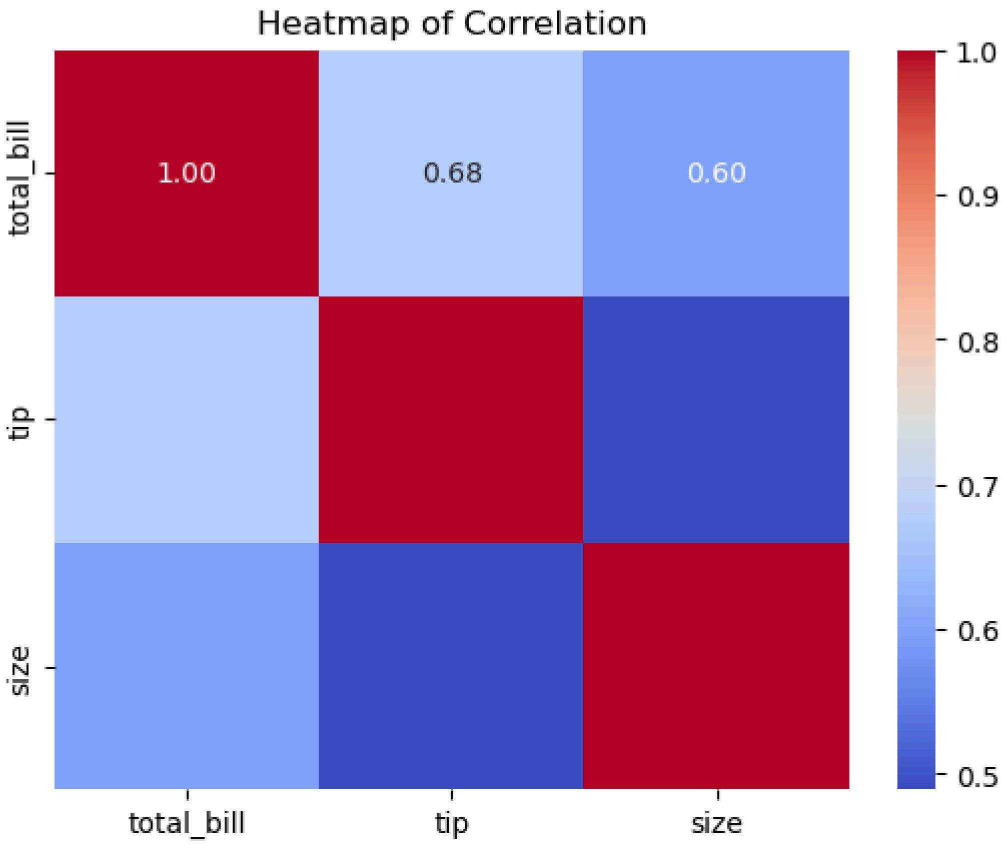
In [13]:
```python
import seaborn as sns
import matplotlib.pyplot as plt

# Load dataset
tips = sns.load_dataset("tips")

# Compute correlation matrix with numeric columns only
corr = tips.corr(numeric_only=True)

# Plot heatmap
sns.heatmap(corr, annot=True, cmap="coolwarm", fmt=".2f")

plt.title("Heatmap of Correlation")
plt.show()
```



Interpretation:

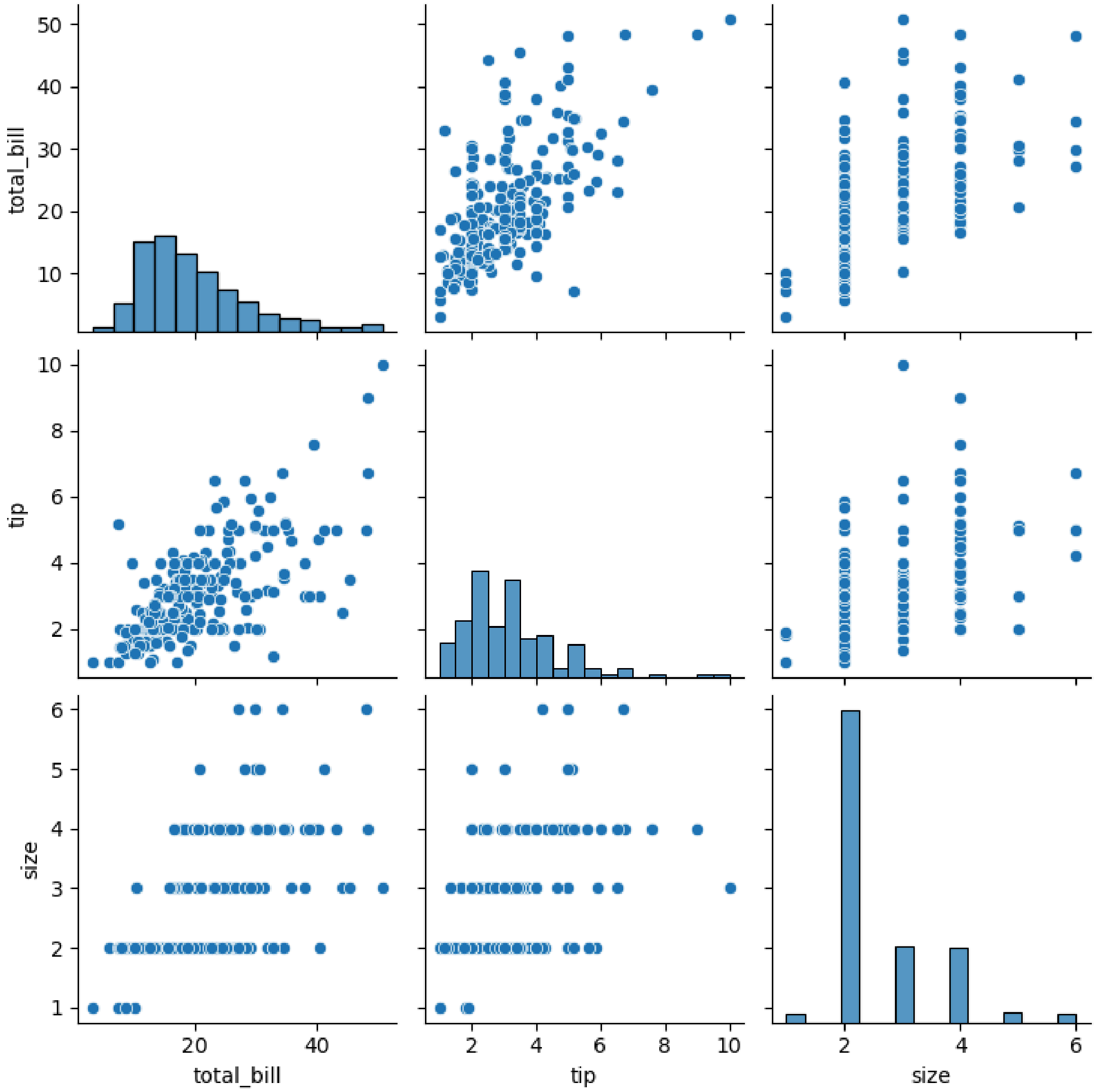
- The heatmap shows correlation values between variables.
- "Total bill" and "tip" have a high positive correlation (~0.68), meaning higher bills usually result in higher tips.
- "Size" (number of people at the table) has a weak correlation with tips.

7. Pair Plot

Used to visualize relationships between multiple numerical variables.

In [14]:
```
sns.pairplot(tips)
plt.show()
```

```
C:\Users\saksh\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarnin
g: use_inf_as_na option is deprecated and will be removed in a future version. Co
nvert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
C:\Users\saksh\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarnin
g: use_inf_as_na option is deprecated and will be removed in a future version. Co
nvert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
C:\Users\saksh\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarnin
g: use_inf_as_na option is deprecated and will be removed in a future version. Co
nvert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
```



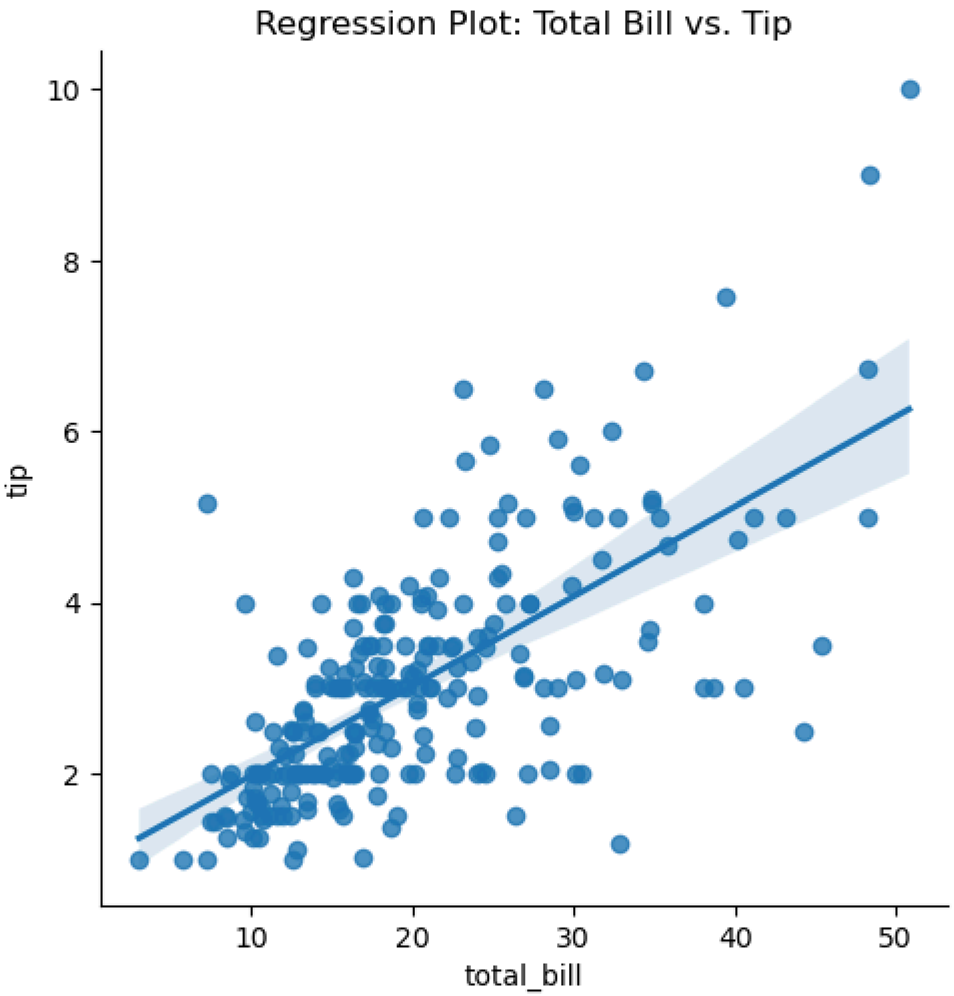
Interpretation:

- A grid of scatter plots showing relationships between all numerical variables.
- Diagonal plots show histograms of individual variables.
- You can identify patterns, trends, and correlations.

9. Regression Plot

Shows a trend line between two numerical variables.

```
In [15]: sns.lmplot(x="total_bill", y="tip", data=tips)
         plt.title("Regression Plot: Total Bill vs. Tip")
         plt.show()
```



Interpretation:

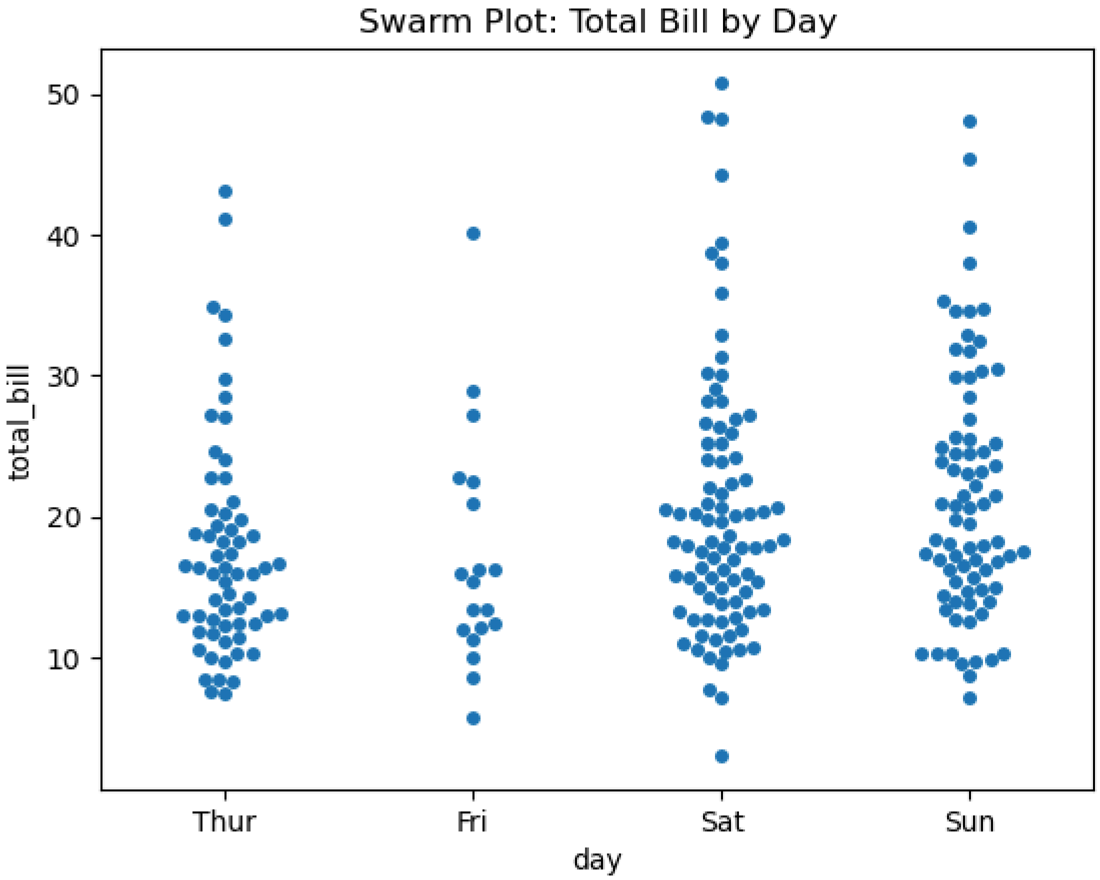
- The line represents the best-fit linear regression model.
- The positive slope indicates that higher bills lead to higher tips.

8. Swarm Plot

Used to show all individual data points while maintaining category separation.

In [16]:
```
sns.swarmplot(x="day", y="total_bill", data=tips)
plt.title("Swarm Plot: Total Bill by Day")
plt.show()
```

```
C:\Users\saksh\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarnin
g: use_inf_as_na option is deprecated and will be removed in a future version. Co
nvert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
C:\Users\saksh\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarnin
g: use_inf_as_na option is deprecated and will be removed in a future version. Co
nvert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
```



Interpretation:

- Each point represents a single transaction.
- Weekends have more high-value bills, suggesting higher spending.