

Understanding the Building Blocks of a Plot :

Title: The title of the plot, which describes the main purpose or content of the visualization.

Axis Labels: The labels for the x and y axes, which describe the variables being plotted.

Tick Marks: The marks along the axes that indicate the scale or range of the data.

Gridlines: The lines that extend from the tick marks to assist in reading the plot.

Legend: The key that explains the colors, shapes, or sizes used to represent different groups or variables.

Data Points: The individual points, bars, lines, or other shapes that represent the data values.

Data Labels: The text or numbers that provide additional information about the data points.

Annotations: The additional text or shapes that highlight specific aspects of the plot.

Background: The overall appearance of the plot, including colors, fonts, and borders.

Data Visualization Cheat Sheet (Matplotlib, Seaborn, Plotly)

Matplotlib

1. Line Plot

```
plt.plot(x, y, marker='o', linestyle='-', color='b', label='Line')  
  
plt.legend()  
  
plt.show()
```

2. Bar Plot

```
plt.bar(x, y, color='g', label='Bar')  
  
plt.legend()  
  
plt.show()
```

3. Scatter Plot

```
plt.scatter(x, y, color='r', label='Scatter')  
  
plt.legend()  
  
plt.show()
```

4. Histogram

```
plt.hist(y, bins=5, color='purple', alpha=0.7)  
  
plt.show()
```

5. Pie Chart

```
plt.pie(y, labels=x, autopct='%1.1f%%')  
  
plt.show()
```

6. Box Plot

```
plt.boxplot([y, z], labels=['Y', 'Z'])  
  
plt.show()
```

7. Heatmap (using imshow)

```
plt.imshow(data, cmap='coolwarm', interpolation='nearest')  
  
plt.colorbar()  
  
plt.show()
```

Seaborn

1. Line Plot

```
sns.lineplot(x='X', y='Y', data=data, hue='Category', marker='o')  
plt.show()
```

2. Bar Plot

```
sns.barplot(x='Category', y='Y', data=data, ci=None, palette='viridis')  
plt.show()
```

3. Scatter Plot

```
sns.scatterplot(x='X', y='Y', data=data, hue='Category', style='Category')  
plt.show()
```

4. Histogram

```
sns.histplot(data['Y'], bins=10, kde=True, color='purple')  
plt.show()
```

5. Box Plot

```
sns.boxplot(x='Category', y='Y', data=data, palette='coolwarm')  
plt.show()
```

6. Heatmap

```
sns.heatmap(data[['X', 'Y', 'Z']].corr(), annot=True, cmap='coolwarm')  
plt.show()
```

7. Pair Plot

```
sns.pairplot(data, hue='Category', palette='husl')  
plt.show()
```

Plotly

1. Line Plot

```
fig = px.line(data, x='X', y='Y', color='Category', markers=True)
fig.show()
```

2. Bar Plot

```
fig = px.bar(data, x='Category', y='Y', color='Category')
fig.show()
```

3. Scatter Plot

```
fig = px.scatter(data, x='X', y='Y', color='Category')
fig.show()
```

4. Histogram

```
fig = px.histogram(data, x='Y', nbins=10, color='Category', marginal='box')
fig.show()
```

5. Box Plot

```
fig = px.box(data, x='Category', y='Y', color='Category')
fig.show()
```

6. Heatmap

```
corr = data[['X', 'Y', 'Z']].corr()

fig = go.Figure(data=go.Heatmap(z=corr.values, x=corr.index, y=corr.columns,
                                colorscale='Viridis'))

fig.show()
```

7. Pie Chart

```
fig = px.pie(data, names='Category', values='Y', hole=0.3)
fig.show()
```

8. 3D Scatter Plot

```
fig = px.scatter_3d(data, x='X', y='Y', z='Z', color='Category')
fig.show()
```

