📊 Python Data Visualization Notes

---

1. Libraries

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import matplotlib.image as mpimg
```

- **Matplotlib** → Basic plotting library for 2D plots
- **Seaborn** → Advanced statistical visualization (based on Matplotlib)
- **Plotly / Cufflinks** → Interactive plots (mainly for Jupyter / Colab)

---

2. Matplotlib Basics

2.1 Simple Plot

```
x = np.linspace(0,5,11)
y = x**2
plt.plot(x, y)
plt.show()
```

2.2 Adding Labels and Title

```
plt.plot(x, y)
plt.title("My Plot")
plt.xlabel("X-Axis")
plt.ylabel("Y-Axis")
plt.show()
```

---

3. Multiple Subplots

```
x = np.linspace(0,5,11)
y = x**2
z = x**3

plt.subplot(2,2,1)  # 2 rows, 2 columns, position 1
plt.plot(x,y)

plt.subplot(2,2,2)
plt.plot(x,z)

plt.subplot(2,2,3)
plt.plot(x,y)

plt.subplot(2,2,4)
plt.plot(x,z)

plt.show()
```

---

4. Object-Oriented Approach

```
fig = plt.figure()
axis1 = fig.add_axes([0.1,0.1,0.8,0.8])   # Main axes
axis2 = fig.add_axes([0.25,0.45,0.35,0.35]) # Small inset axes

axis1.plot(x,y)
axis2.plot(x,z)
plt.show()
```

---

5. Types of Plots in Matplotlib

```
plt.scatter(x,y)       # Scatter plot
plt.hist(x)            # Histogram
plt.boxplot(x)         # Box plot
```

5.1 Customizing

```
plt.plot(x,y, color="blue", linewidth=2, linestyle='-', marker='o')
```

5.2 Saving Plots

```
plt.savefig("my_plot.png")
```

5.3 Working with Images

```
img = mpimg.imread("my_plot.png")
plt.imshow(img)
plt.show()
```

---

6. Seaborn Basics

6.1 Loading Dataset

```
df = sns.load_dataset('tips')
```

6.2 Distribution Plots

```
sns.displot(df['total_bill'])
sns.histplot(df['total_bill'], bins=10)
sns.displot(df['total_bill'], bins=50, kde=True)
```

6.3 Comparing Two Quantities

```
sns.jointplot(x='total_bill', y='tip', data=df)          # Default
scatter
sns.jointplot(x='total_bill', y='tip', data=df, kind='kde')
sns.jointplot(x='total_bill', y='tip', data=df, kind='hist')
sns.jointplot(x='total_bill', y='tip', data=df, kind='hex')
```

6.4 Pair Plots

```
sns.pairplot(df)
```

```
sns.pairplot(df, hue='sex', palette='rainbow')  # Differentiate by category
```

6.5 Rug Plot

```
sns.rugplot(df['total_bill'])
```

---

7. Categorical Plots

7.1 Count Plot

```
sns.countplot(df['sex'])
sns.countplot(x=df['sex'], hue=df['smoker'])
```

7.2 Bar Plot

```
sns.barplot(x=df['sex'], y=df['total_bill'])
sns.barplot(x=df['sex'], y=df['total_bill'], estimator=np.sum)
```

7.3 Box / Violin / Strip / Swarm Plots

```
sns.boxenplot(x='tip', y='day', data=df, palette='rainbow')
sns.violinplot(x='day', y='total_bill', data=df)
sns.stripplot(x='day', y='total_bill', data=df)
sns.swarmplot(x='day', y='total_bill', data=df)

# Combining violin + swarm
sns.violinplot(x='tip', y='day', data=df)
sns.swarmplot(x='tip', y='day', data=df)
```

---

8. Heatmaps and Correlation

8.1 Correlation Heatmap

```
tipscorr = df[['total_bill', 'tip', 'size']]
sns.heatmap(tipscorr.corr(), annot=True, cmap='coolwarm')
```

```
plt.show()
```

8.2 Cluster Map

```
sns.clustermap(tipscorr.corr())
```

---

9. Pivot Tables for Heatmaps

```
flights = sns.load_dataset('flights')
pvtflight = flights.pivot_table(values='passengers', index='month',
columns='year')
sns.heatmap(pvtflight, cmap='YlGnBu')
plt.show()
```

---

10. Regression Plots

```
sns.lmplot(x='total_bill', y='tip', data=df)
```

---

11. Interactive Plots (Plotly / Cufflinks)

```
import cufflinks as cf
cf.go_offline()  # For Jupyter / Colab

tips['total_bill'].iplot()
tips.groupby('day')['tip'].mean().iplot(kind='bar')
```