PRESENTATION

By Rishabh Dubey

AGENDA

1. Quick Revision Quiz (Pandas)
2. Introduction to Data Visualization
3. Core Graphs
4. Essential Matploltlib Functions
6. Class Hands-on Exercise
7. Q&A & Wrap-up

Quiz Time

QUICK REVISION QUIZ (MCQS & TRUE/FALSE)

Q1: What is Pandas primarily used for?

A) Web development
B) Data manipulation and analysis
C) Image processing
D) Machine learning models

QUICK REVISION QUIZ (MCQS & TRUE/FALSE)

Q2: Which data structure in Pandas is used to represent a table with labeled axes?

A) Series
B) DataFrame
C) Array
D) Dictionary

QUICK REVISION QUIZ (MCQS & TRUE/FALSE)

Q3: How do you check the first 5 rows of a DataFrame named df?

A) df.head(5)
B) df.tail(5)
C) df.show(5)
D) df.first(5)

QUICK REVISION QUIZ (MCQS & TRUE/FALSE)

Q4: What function is used to remove missing values from a DataFrame?

A) dropna()

B) fillna()

C) replace()

D) remove_na()

QUICK REVISION QUIZ (MCQS & TRUE/FALSE)

Q5: What does the interpolate() function do in Pandas?

A) Removes missing values

B) Fills missing values using interpolation

C) Drops duplicate values

D) Sorts the DataFrame

Introduction to Data Visualization

INTRODUCTION TO MATPLOTLIB

📌 What is Matplotlib?

- A Python library for visualization
- Works with NumPy & Pandas
- Creates static, animated, and interactive plots

IMPORTANT GRAPHS IN MATPLOTLIB

Graph Type	Function	Use Case
Line Plot	plt.plot()	Trends over time
Bar Chart	plt.bar()	Comparing categories
Histogram	plt.hist()	Distribution of data
Scatter Plot	plt.scatter()	Relationship between variables
Box Plot	plt.boxplot()	Detecting outliers
Pie Chart	plt.pie()	Showing proportions

LINE PLOT – TRENDS OVER TIME

📌 What is a Line Plot?

- A continuous line connecting data points
- Used to show trends, patterns, and changes over time

When to Use?

✅ When analyzing time-series data

✅ When tracking continuous data changes

📌 When NOT to Use?

❌ When comparing categories (Use bar chart instead)

❌ When dealing with discrete variables

```
import matplotlib.pyplot as plt
x = [1, 2, 3, 4, 5]
y = [10, 20, 15, 25, 30]
plt.plot(x, y, marker='o', linestyle='--', color='r')
plt.xlabel("Time")
plt.ylabel("Value")
plt.title("Line Plot Example")
plt.show()
```

BAR CHART – COMPARING CATEGORIES

📌 What is a Bar Chart?

- Represents categorical data with rectangular bars
- Bar height represents value/count

📌 **When to Use?**

✅ When comparing categories or groups

✅ When showing discrete data distributions

📌 **When NOT to Use?**

❌When visualizing continuous trends (Use line plot instead)

❌ When displaying too many categories (Use grouped bar charts or different visualization)

```
categories = ["A", "B", "C", "D"]
values = [10, 15, 7, 12]
plt.bar(categories, values, color='blue')
plt.xlabel("Categories")
plt.ylabel("Values")
plt.title("Bar Chart Example")
plt.show()
```

HISTOGRAM – DATA DISTRIBUTION

📌 What is a Histogram?

- Shows frequency distribution of a dataset
- Bins group data into ranges

When to Use?

✅ When analyzing distribution & shape of data

✅ When checking skewness or normality

📌 **When NOT to Use?**

❌ When comparing individual values (Use bar chart)

❌ When dealing with categorical data

```
import numpy as np
data = np.random.randn(1000)
plt.hist(data, bins=20, color='green', alpha=0.7)
plt.xlabel("Value")
plt.ylabel("Frequency")
plt.title("Histogram Example")
plt.show()
```

SCATTER PLOT – RELATIONSHIPS BETWEEN VARIABLES

📌 What is a Scatter Plot?

- Represents individual data points
- Helps to identify correlations & patterns

When to Use?

✅When analyzing relationships between two variables

✅ When checking for clusters or trends

📌 **When NOT to Use?**

❌ When one variable is categorical (Use bar chart instead)

❌ When there’s too much overlapping data (Use hexbin plot)

```
x = [10, 20, 30, 40, 50]
y = [5, 15, 25, 35, 45]
plt.scatter(x, y, color='red')
plt.xlabel("X-axis")
plt.ylabel("Y-axis")
plt.title("Scatter Plot Example")
plt.show()
```

BOX PLOT – DETECTING OUTLIERS

📌 What is a Box Plot?

- Displays data distribution, median, and outliers
- Uses quartiles to summarize spread

📌 **When to Use?**

✅ When detecting outliers & anomalies

✅ When comparing multiple distributions

📌 **When NOT to Use?**

❌ When visualizing exact values (Use scatter plot)

❌ When comparing few data points

```
data = [7, 8, 5, 12, 15, 9, 10, 18, 25, 30]
plt.boxplot(data)
plt.title("Box Plot Example")
plt.show()
```

PIE CHART – SHOWING PROPORTIONS

📌 What is a Pie Chart?

- Represents parts of a whole
- Each slice = percentage of total

📌**When to Use?**

✅ When showing proportions & percentages

✅ When comparing a few categories

📌 **When NOT to Use?**

❌ When categories exceed 5-6 (Use bar chart instead)

❌ When values are similar in size (Difficult to differentiate)

```
labels = ["A", "B", "C", "D"]
sizes = [30, 20, 25, 25]
plt.pie(sizes, labels=labels, autopct="%1.1f%%",
colors=["red", "blue", "green", "yellow"])
plt.title("Pie Chart Example")
plt.show()
```

BAR GRAPH VS HISTOGRAM

📌 Key Differences

Feature	Bar Graph	Histogram
Purpose	Compares categories	Shows data distribution
Data Type	Categorical data (e.g., colors, brands, cities)	Continuous data (e.g., age, height, temperature)
Bars Touch?	❌ No (Bars have gaps)	✅ Yes (Bars are connected)
X-Axis Representation	Discrete categories	Ranges (bins)
Use Case	Comparing different groups	Understanding spread & frequency

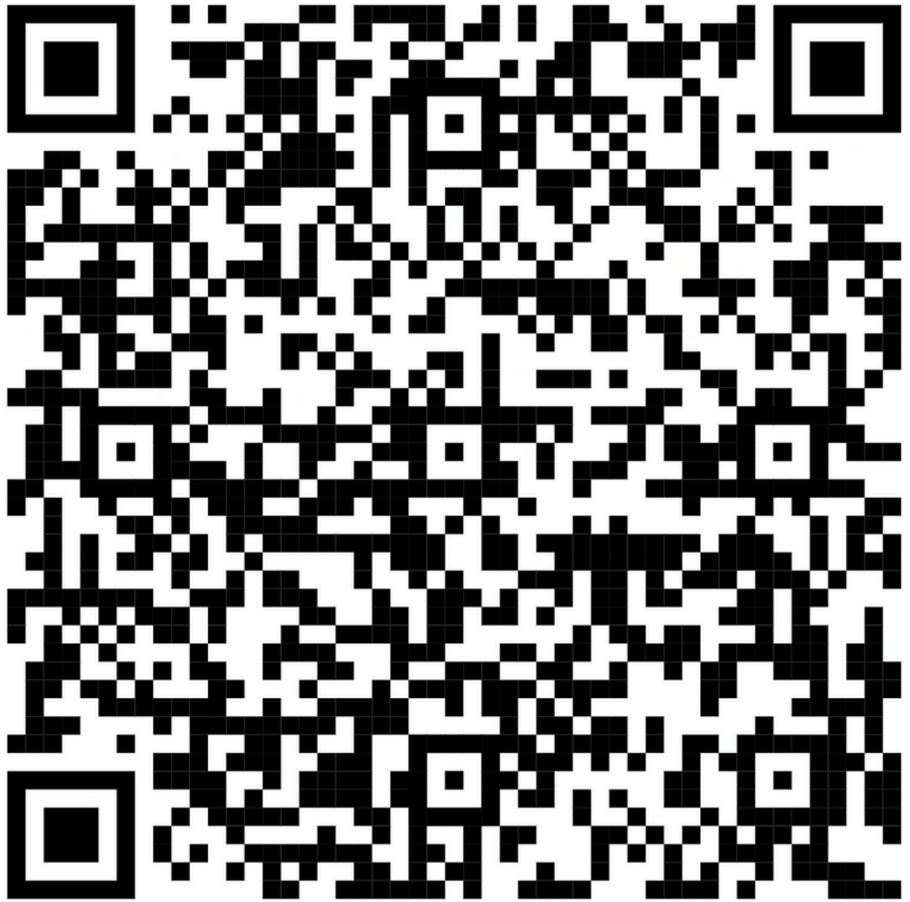
MATPLOTLIB IMPORTANT FUNCTIONS

Function	Purpose
plt.plot()	Creates a line plot
plt.bar()	Creates a bar chart
plt.hist()	Creates a histogram
plt.scatter()	Creates a scatter plot
plt.boxplot()	Creates a box plot

plt.pie()	Creates a pie chart
plt.xlabel()	Adds X-axis label
plt.ylabel()	Adds Y-axis label
plt.title()	Adds title
plt.legend()	Adds legend
plt.grid()	Adds gridlines
plt.show()	Displays the plot

LINK TO MATPLOTLIB GIST

https://gist.github.com/Rishabh7406/135b59cb8133e03d73cdd9ddb121090a



📌 Practice Task: Matplotlib Hands-On (15-20 mins)

Task: Practice Key Matplotlib Functions

📝 Complete the following tasks using Matplotlib:

1️⃣ Line Plot:

- Plot x = [1, 2, 3, 4, 5] and y = [5, 10, 5, 15, 10].
- Add a title, labels, and a grid.

2️⃣ Bar Chart:

- Create a bar chart for Categories = ["A", "B", "C", "D"] with values [12, 7, 15, 10].
- Customize colors and add axis labels.

3️⃣ Histogram:

- Generate 500 random numbers and plot a histogram.
- Use 20 bins and apply transparency (alpha=0.7).

📌 Practice Task: Matplotlib Hands-On (15-20 mins)

4️⃣ Scatter Plot:

- Generate two sets of random numbers (size 50) and plot a scatter plot.
- Customize markers and add a title.

5️⃣ Subplots(optional):

- Create a 1x2 subplot:
  - First Plot: Line Plot
  - Second Plot: Bar Chart
- Use plt.tight_layout() for spacing.

THANK YOU