

# MATPLOTLIB –

---

## 1 Graph Customization

### ◆ Theory (Brief)

Graph ko **readable** aur **attractive** banane ke liye:

- color → line ka rang
  - linestyle → line ka style
  - marker → points ka shape
  - grid → reading easy
- 

### ◆ Example Code

```
import matplotlib.pyplot as plt

x = [1, 2, 3, 4, 5]
y = [10, 20, 30, 40, 50]

plt.plot(x, y, color='red', linestyle='--', marker='o')
plt.grid(True)
plt.title("Customized Line Graph")
plt.show()
```

### ◆ Explanation

- color='red' → line ka color
- linestyle='--' → dotted line
- marker='o' → points dikhte hain
- plt.grid(True) → background grid

## 2 Labels & Legend

Graph ko **self-explanatory** banane ke liye:

- X-axis label
  - Y-axis label
  - Legend (line ka meaning)
- 

### ◆ Example Code

```
plt.plot(x, y, label="Sales")
plt.xlabel("Days")
plt.ylabel("Amount")
plt.legend()
plt.show()
```

---

### ◆ Explanation

label → line ka name  
legend() → label display karta hai  
xlabel() / ylabel() → axis names

## 3 Multiple Lines (Comparison)

Ek graph me 2 ya 3 datasets compare karte hain.

---

### ◆ Example Code

```
y1 = [10, 20, 30, 40, 50]
y2 = [8, 18, 28, 38, 48]
```

```
plt.plot(x, y1, label="2024")
plt.plot(x, y2, label="2025")
plt.legend()
plt.show()
```

---

### ◆ Explanation

Same x ke saath different y  
Multiple lines ek hi graph me

---

## 4 Subplots

Ek window me multiple graphs dikhane ke liye.

---

◆ Example Code

```
plt.subplot(1, 2, 1)  
plt.plot(x, y)  
plt.title("Line Graph")
```

```
plt.subplot(1, 2, 2)  
plt.bar(x, y)  
plt.title("Bar Graph")
```

```
plt.show()
```

---

◆ Explanation

subplot(1,2,1) → 1 row, 2 columns, 1st graph  
subplot(1,2,2) → 2nd graph

---

## 5 Pandas + Matplotlib (Real Data)

Real data mostly CSV / Excel me hota hai.

---

◆ Example Code

```
import pandas as pd  
import matplotlib.pyplot as plt
```

```
data = {  
    "Day": [1, 2, 3, 4, 5],  
    "Sales": [100, 150, 130, 180, 200]  
}
```

```
df = pd.DataFrame(data)
```

```
plt.plot(df["Day"], df["Sales"])  
plt.title("Sales Data")  
plt.show()
```

---

◆ Explanation

Pandas data handle karta hai  
Matplotlib graph banata hai

---

## 6 Save Graph as Image

Graph ko image file me save karte hain.

---

### ◆ Example Code

```
plt.plot(x, y)  
plt.savefig("sales_graph.png")  
plt.show()
```

---

### ◆ Explanation

**savefig()** → graph ko file me save karta hai

.png, .jpg supported

---

## 7 Large Data Handling (Concept Only)

Large data me:

- Trend important hota hai
- Har point dikhana zaroori nahi
- Same code kaam karta hai

## 4 PRACTICE QUESTIONS (IMPORTANT)

### 1 Level 1 – Basic

1. Create a line graph using numbers from 1 to 10.

- 
2. Create a bar chart showing marks of 5 students.
  3. Create a scatter plot using any 5 values.
  4. Create a histogram using random numbers.
  5. Create a pie chart showing percentage distribution of any 3 categories.
- 

## **Level 2 – Intermediate**

6. Create a line graph and add a title, X-axis label, and Y-axis label.
  7. Create a graph with two lines showing comparison of two subjects' marks.
  8. Create a bar chart and add labels and title to it.
  9. Display a line graph and a bar graph in the same window using subplots.
  10. Customize a line graph by changing color, line style, and marker
- 

## **Level 3 – Advanced / Practice Project**

11. Create a Pandas DataFrame using a dictionary.
12. Plot a graph using the DataFrame values.
13. Read a CSV file and plot a graph using one column as X-axis and another as Y-axis.
14. Save a plotted graph as an image file.
15. Create a graph using a large dataset and explain the trend.