### 📝 Problem 1: Dual Y-Axis Plot - Stock Price & Trading Volume

📊 **Scenario:** You are analyzing a stock’s performance over **10 days**. You need to visualize the **closing price** and **trading volume** in a single plot using **dual Y-axes**.

📌 **Instructions:**  
- Use **closing price** (stock\_price) on the **left Y-axis**. - Use **trading volume** (trading\_volume) on the **right Y-axis**. - Format the plot with different **line styles, markers, and colors**. - Add proper **titles, labels, legends, and grid lines**.

days = np.arange(1, 11)   
stock\_price = [150, 152, 149, 155, 160, 162, 158, 165, 170, 175] # In USD  
trading\_volume = [1.2, 1.4, 1.1, 1.8, 2.0, 2.5, 2.2, 2.7, 3.0, 3.5] # In million shares

✅ **Concepts Covered:** twinx(), dual Y-axes, customization, legends

⏳ **Estimated Time: 30 minutes**

### 📝 Problem 2: Stacked Bar Chart - Monthly Expenses Comparison

📊 **Scenario:** You are tracking your **monthly expenses** across different categories (**Rent, Food, Entertainment, and Transport**) for the first **6 months** of the year. Create a **stacked bar chart** to compare the expenses for each category over time.

📌 **Instructions:**  
- Create a **stacked bar chart** to visualize monthly expenses. - Use **different colors** for each expense category. - Add a **legend, labels, and a title** to improve readability.

months = ["Jan", "Feb", "Mar", "Apr", "May", "Jun"]  
rent = [1200, 1200, 1200, 1200, 1250, 1250]  
food = [400, 420, 450, 480, 500, 520]  
entertainment = [150, 200, 220, 180, 250, 300]  
transport = [100, 120, 130, 110, 140, 160]

✅ **Concepts Covered:** bar(), stacking bars, adding legends, colors

⏳ **Estimated Time: 30 minutes**

### 📝 Problem 3: Subplots - COVID-19 Cases & Vaccination Progress

📊 **Scenario:** You have **weekly COVID-19 cases** and **vaccination rates** for **8 weeks**. Visualize both metrics using **two subplots in one figure**.

📌 **Instructions:**  
- The **first subplot** should display **weekly cases** using a **line plot with markers**. - The **second subplot** should show **vaccination progress (%)** using a **bar chart**. - Use **different colors, labels, and titles** for both subplots. - Apply tight\_layout() to prevent overlap.

weeks = np.arange(1, 9)   
covid\_cases = [500, 700, 1200, 2000, 3000, 4500, 6000, 8000]   
vaccination\_rate = [5, 10, 20, 35, 50, 65, 75, 85] # In percentage

✅ **Concepts Covered:** subplots(), multiple charts, formatting

⏳ **Estimated Time: 30 minutes**