Task 6: Get Basic Sales Summary from a Tiny SQLite Database using Python
Objective: Use SQL inside Python to pull simple sales info (like total quantity sold, total revenue), and display it using basic print statements and a simple bar chart.
Tools Required:

- Python (with sqlite3, pandas, matplotlib)
- SQLite (built into Python — no setup!)
- Jupyter Notebook or a .py file

Deliverables: Python script (or notebook) that:

- Connects to sales_data.db
- Runs 1–2 SQL queries
- Displays output using print and basic matplotlib bar chart

Hints / Mini Guide:

- Load SQLite database: import sqlite3 conn = sqlite3.connect("sales_data.db")
- Run basic SQL: query = "SELECT product, SUM(quantity) AS total_qty, SUM(quantity * price) AS revenue FROM sales GROUP BY product"
- Load into pandas: import pandas as pd df = pd.read_sql_query(query, conn)
- Print results: print(df)
- Plot simple bar chart: df.plot(kind='bar', x='product', y='revenue')
- Save chart if needed: plt.savefig("sales_chart.png")

Dataset suitable for this Task: Create a small SQLite database file (sales_data.db) with just one sales table

By completing this task, you will:

- Learn how to write basic SQL queries
- Practice importing SQL data into Python
- Perform simple data summaries
- Create your first sales chart

Interview Questions Related To Above Task:

- How did you connect Python to a database?
- What SQL query did you run?
- What does GROUP BY do?
- How did you calculate revenue?
- How did you visualize the result?
- What does pandas do in your code?
- What's the benefit of using SQL inside Python?
- Could you run the same SQL query directly in DB Browser for SQLite?

DATA ANALYTICS INTERNSHIP

Task Submission Guidelines

⏰ Time Window: You can complete the task anytime between 10:00 AM of Assigned task to 10:00 AM of next day. Submission link closes at 10:00 AM of next day

🔍 Self-Research Allowed: You are free to explore, Google, or refer to tutorials to understand concepts and complete the task effectively.

🛠️ Debug Yourself: Try to resolve a l errors by yourself. This helps you learn problem-solving and ensures you don't face the same issues in future tasks.

💸 No Paid Tools: If the task involves any paid software/tools, do not purchase anything. Just learn the process or find free alternatives.

📂 GitHub Submission: Create a new GitHub repository for each task. Add everything you used for the task — code, datasets, screenshots (if any), and a short README.md explaining what you did.

📤 Submit Here: After completing the task, paste your GitHub repo link and submit it using the link below:
[Submission Link]