**Interview Questions**

**1. How did you connect Python to a database?**  
I used Python’s built-in **sqlite3** module, which lets me easily connect to a local SQLite database file. I wrote:

conn = sqlite3.connect('sales\_data.db')

This single line opens a connection to my database file named **sales\_data.db**. After that, I can create tables, insert data, and run SQL queries right inside my Python script or notebook.

**2. What SQL query did you run?**  
I ran this SQL query:

SELECT product, SUM(quantity) AS TOTAL\_QTY, SUM(quantity \* price) AS REVENUE FROM sales\_data GROUP BY product;

This query goes through my sales table, finds all rows with the same product name, adds up the total quantity sold for each product, and also calculates the total revenue by multiplying quantity and price for each sale.

**3. What does GROUP BY do?**  
GROUP BY groups rows that have the same value in a specific column — in my case, the **product** column. Instead of treating all sales rows as one big chunk, GROUP BY splits them by product name. This way, I can get totals **per product**, like “how many units of Product A” and “how much money Product A made.”

**4. How did you calculate revenue?**  
To get revenue, I multiplied the quantity sold by the price for each row: quantity \* price. Then I wrapped that in SUM() to add up all those amounts for each product. So, the query tells me **how much money** each product brought in altogether.

**5. How did you visualize the result?**  
After I got the totals using SQL, I loaded the result into a **pandas DataFrame**. Then, I used **seaborn** and **matplotlib** to create a simple **bar chart**. The bar chart has **products** on the X-axis and **revenue** on the Y-axis. This makes it easy to see which product earned the most at a glance.

**6. What does pandas do in your code?**  
pandas makes it easy to **load**, **organize**, and **work with** the SQL data. After I run my SQL query, I use:

df = pd.read\_sql\_query(query, conn)

This loads the result into a **DataFrame**. With that, I can **print tables**, **filter data**, or make charts. Basically, pandas turns my raw SQL output into an easy-to-handle table inside Python.

**7. What’s the benefit of using SQL inside Python?**  
It’s very convenient because I can do everything in one place. I can **create databases**, **run SQL**, and **analyze the results** all in Python. This saves time — I don’t need to switch between a database tool and Python separately. It’s also good for automating repetitive tasks and for creating reusable scripts.

**8. Could you run the same SQL query directly in DB Browser for SQLite?**  
Yes — absolutely. The same SQL query works perfectly in **DB Browser for SQLite**. It’s just plain SQL, so you could paste it into DB Browser, click **Run**, and see the same grouped totals there too.