Daily Sales Summary Report

Set Up the SQLite Database

```
CREATE TABLE orders (
  order_date TEXT,
  total_orders INT,
  total_sales REAL,
  Avg_Order_value REAL
);
```

Populate it with some example data

```
INSERT INTO orders (order_date, amount) VALUES
('2025-06-01', 200.0),
('2025-06-02', 115.0),
('2025-06-02', 50.0);
```

Sales Data Table:

Order_date	Total_Orders	Total Sales ($)	Avg_Order_Value ($)
2025-06-01	2	400.0	200.0
2025-06-02	2	230.0	115.0
2025-06-03	1	50.0	50.0

Insights

- - 2025-06-01 had the highest sales and average order value, indicating strong performance.
- - 2025-06-02 saw a drop in revenue but had the same number of orders as the previous day.
- - 2025-06-03 had the lowest activity, possibly indicating low customer engagement or fewer promotions.

Python Code :

```
import sqlite3
import pandas as pd
import matplotlib.pyplot as plt
```

```python
# 1. Load SQLite database
conn = sqlite3.connect("sales_data.db")

# 2. Run basic SQL query = """
SELECT
    product,
    SUM(quantity) AS total_qty,
    SUM(quantity * amount / quantity) AS revenue
FROM sales
GROUP BY product
"""
df = pd.read_sql_query(query, conn)

# 3. Print results
print("=== Sales Summary ===")
print(df)



# 4. Plot simple bar chart
df.plot(kind='bar', x='product', y='revenue', color='skyblue', legend=False)
plt.title("Revenue by Product")
plt.xlabel("Product")
plt.ylabel("Revenue ($)")
plt.tight_layout()

# 5. Save chart to file (optional)
plt.savefig("sales_chart.png")
plt.show()

# 6. Close connection
conn.close()Sales Bar Chart
```

Total Sales by Date