Monthly Revenue Analysis from Online Sales Data

What I Did

I designed and executed a SQL project that analyzes monthly sales revenue from an online store. The primary goal was to extract meaningful insights from a sales dataset, including total monthly revenue and top-performing months.

How I Did It

1. Database and Table Creation

Database: Created a new database named 'sales_db'.

Table: Created a table 'online_sales' with columns:

- order_id (integer)

- order_date (date)

- amount (decimal)

- product_id (integer)

SQL:

```
CREATE DATABASE sales_db;

USE sales_db;

CREATE TABLE online_sales (

    order_id INT,

    order_date DATE,

    amount DECIMAL(10, 2),

    product_id INT
```

);

2. Data Insertion

Inserted 15 rows of mock sales data with various order dates, amounts, and product IDs.

SQL:

INSERT INTO online_sales (order_id, order_date, amount, product_id) VALUES

(1, '2023-01-10', 120.50, 101), ...

(15, '2023-08-01', 220.00, 114);

3. Basic Query

Displayed all data from the online_sales table.

SQL:

SELECT * FROM online_sales;

4. Monthly Revenue and Order Volume Analysis

Grouped sales by year and month. Calculated:

- total_revenue (sum of amount)

- order_volume (number of unique orders)

SQL:

SELECT

    YEAR(order_date) AS year,

    MONTH(order_date) AS month,

    SUM(amount) AS total_revenue,

    COUNT(DISTINCT order_id) AS order_volume

```
FROM online_sales
GROUP BY YEAR(order_date), MONTH(order_date)
ORDER BY year, month;
```

5. Top 3 Months by Revenue

Found the three highest-revenue months across the dataset.

SQL:

```
SELECT
    YEAR(order_date) AS year,
    MONTH(order_date) AS month,
    SUM(amount) AS total_revenue
FROM online_sales
GROUP BY YEAR(order_date), MONTH(order_date)
ORDER BY total_revenue DESC
LIMIT 3;
```

Outcome

Successfully analyzed monthly sales trends.

Identified peak sales months.

Created a foundation for deeper business insights and trend forecasting.