.

Task 3: SQL Basics – Filtering, Sorting & Aggregations

Objective:

To understand and apply basic SQL operations such as WHERE (filtering), ORDER BY (sorting), and aggregate functions (COUNT, SUM, AVG, MIN, MAX) on a dataset.

Dataset Used

Chinook Database (Beginner-friendly SQL dataset)

Common tables used:

Customer

Invoice

InvoiceLine

Track

Artist

1️⃣ **Filtering Data (WHERE clause)**

Example 1: Find customers from India

Copy code

Sql

```
SELECT *

FROM Customer

WHERE Country = 'India';
```

Analysis:

This query filters rows and returns only customers whose country is India.

Example 2: Invoices with total greater than 10

Copy code

Sql

```
SELECT *
```

```sql
FROM Invoice
WHERE Total > 10;
```

Analysis:

Used to filter invoices where the total bill amount exceeds 10.

2️⃣ Sorting Data (ORDER BY)

Example 3: Sort customers by First Name (A–Z)

Copy code

Sql

```sql
SELECT FirstName, LastName
FROM Customer
ORDER BY FirstName ASC;
```

Analysis:

Sorts customer names in ascending alphabetical order.

Example 4: Sort invoices by total amount (highest first)

Copy code

Sql

```sql
SELECT InvoiceId, Total
FROM Invoice
ORDER BY Total DESC;
```

Analysis:

Displays invoices starting from the highest total value.

3️⃣ Aggregation Functions

Example 5: Count total number of customers

Copy code

Sql

```sql
SELECT COUNT(*) AS Total_Customers
```

```
FROM Customer;
```

Analysis:

Counts the total number of records in the Customer table.

Example 6: Find total revenue

Copy code

Sql

```
SELECT SUM(Total) AS Total_Revenue
FROM Invoice;
```

Analysis:

Calculates the sum of all invoice totals (overall sales).

Example 7: Find average invoice value

Copy code

Sql

```
SELECT AVG(Total) AS Average_Invoice
FROM Invoice;
```

Analysis:

Returns the average invoice amount.

Example 8: Minimum and maximum invoice value

Copy code

Sql

```
SELECT
   MIN(Total) AS Min_Invoice,
   MAX(Total) AS Max_Invoice
FROM Invoice;
```

Analysis:

Shows the smallest and largest invoice totals.

4️⃣ Grouping Data (GROUP BY)

Example 9: Total sales by country

Copy code

Sql

```
SELECT BillingCountry, SUM(Total) AS Total_Sales

FROM Invoice

GROUP BY BillingCountry;
```

Analysis:

Groups invoices by country and calculates total sales per country.

Example 10: Number of invoices per customer

Copy code

Sql

```
SELECT CustomerId, COUNT(InvoiceId) AS Invoice_Count

FROM Invoice

GROUP BY CustomerId;
```

Analysis:

Counts how many invoices each customer has.

5️⃣ Filtering Groups (HAVING)

Example 11: Countries with sales greater than 100

Copy code

Sql

```
SELECT BillingCountry, SUM(Total) AS Total_Sales

FROM Invoice

GROUP BY BillingCountry

HAVING SUM(Total) > 100;
```

Analysis:

Filters grouped results using HAVING (used with aggregate functions).

✅ Conclusion (For Submission)

In this task, SQL queries were used to:

Filter data using WHERE

Sort records using ORDER BY

Perform calculations using aggregate functions

Analyze grouped data using GROUP BY and HAVING