📘 Module 12: SQL Introduction

🔷 Introduction to SQL

SQL (Structured Query Language) is used to manage and manipulate **relational databases**.

It helps users store, retrieve, and analyze data using simple commands.

❓ Why Not Excel? Why SQL?

Feature	Excel	SQL
Data Volume	Limited rows (~1M max)	Handles millions of records
Multi-user	Difficult	Designed for multiple users
Security	Basic password protection	Robust access control
Automation	Manual	Highly scriptable and programmable

Use Excel for small, personal datasets.

Use SQL for large, structured, scalable databases.

🗂️ What is a Relational Database?

- A **Relational Database** stores data in tables (rows & columns).
- Tables are related using **keys** (Primary Key, Foreign Key).
- Example:
  - `Students (StudentID, Name, Age)`
  - `Marks (StudentID, Subject, Score)`
  - `StudentID` links both tables.
  -

🎯 What You'll Learn Ahead

✅ SQL & DB Basics

- Writing SQL queries
- Creating/modifying database tables

🛠️ Fundamental SQL Commands

Command	Purpose
`CREATE`	Create tables/databases
`SELECT`	Retrieve data
`INSERT`	Add new data
`ALTER`	Modify table structure
`UPDATE`	Modify existing records

🔎 Data Filtering & Sorting

Keyword	Use
`WHERE`	Filter rows
`ORDER BY`	Sort results
`AND`, `OR`	Combine multiple conditions
`NOT`, `IN`, `BETWEEN`, `LIKE`	Advanced filters

🧮 Aggregates & Grouping

Function	Use
`SUM()`	Total values
`COUNT()`	Number of rows
`MIN()`	Minimum value
`MAX()`	Maximum value
`GROUP BY`	Group data
`HAVING`	Filter groups

🔗 JOINS – Combine Data from Tables

Join Type	Description
`INNER JOIN`	Common data in both tables
`LEFT JOIN`	All left + matching right data
`RIGHT JOIN`	All right + matching left data
`FULL OUTER JOIN`	All data, match or not
`CROSS JOIN`	Cartesian product
`UNION` / `EXCEPT`	Combine / subtract datasets

🧠 Advanced Topics

- **Subqueries** – Nested `SELECT` statements
- **Views** – Virtual tables
- **Indexes** – Faster data access

🔤 String Functions

Function	Use
`UPPER()` / `LOWER()`	Case conversion
`TRIM()` , `LTRIM()` , `RTRIM()`	Remove spaces
`SUBSTRING()`	Extract text
`REPLACE()`	Replace characters
`CONCAT()`	Join strings
`STRING_AGG()`	Combine string rows

🔢 Mathematical Functions

Function	Use
`CEIL()`	Round up
`FLOOR()`	Round down
`ROUND()`	Round to decimal
`RANDOM()`	Generate random
`POWER()`	Exponents

📅 Date & Time Functions

Function	Use
`CURRENT_DATE` , `CURRENT_TIME`	Get now
`DATEDIFF()`	Date difference
`MONTH()` , `YEAR()`	Extract parts of date

🔄 Data Type Conversion

To Type	Function Examples
String	`CAST(... AS VARCHAR)`
Date	`CAST(... AS DATE)`
Time	`CAST(... AS TIME)`

🔍 Pattern Matching

Technique	Use
`LIKE`	Match patterns using `%` or `_`
`REGEXP`	Match complex regex patterns

📌 Summary

- SQL is essential for managing structured data in relational databases.
- It helps retrieve, filter, organize, and analyze large datasets efficiently.
- You've seen core commands (`SELECT`, `INSERT`, `UPDATE`, etc.), joins, filters, aggregates, functions, and more.
- As we move forward, you'll learn to apply these concepts hands-on with real datasets.

💡 Pro Tip:

Just like Excel formulas, SQL is best learned by **doing**. Practice daily, build queries, and try out challenges to improve.