🟨 SQL – Structured Query Language

1️⃣ What happens in the Frontend and Backend

- All the actions users perform on the **frontend** (like filling a form, making a payment, logging in, etc.) are stored in the **backend**, which is the **database**.
- Think of the database as a big **folder or filing cabinet** where all data is kept safely.
- Example: When you sign up on a website, your name, email, and password are saved in a database, not in the app itself.

2️⃣ What is SQL

- **SQL stands for Structured Query Language.**
- It is a **language used to communicate with databases** — to store, retrieve, update, and delete data.
- In short, SQL helps us **talk to the database** and perform operations on it.

3️⃣ RDBMS and SQL

- **RDBMS** stands for **Relational Database Management System**. Examples include MySQL, PostgreSQL, Oracle, SQL Server, etc.
- These are **products (or systems)** that manage databases.
- To **interact with these products**, we use **SQL** as the standard language.

Example: Think of RDBMS as a smartphone, and SQL as the language (like English) you use to give it commands.

4️⃣ SQL Versions

- **SQL-92** (released in 1992) was the **first standardized version** of SQL.
- Most modern database systems like **MySQL, PostgreSQL, Oracle, and SQL Server** still follow SQL-92 standards, with some additional features or syntax differences.
- This ensures that SQL remains a **universal language** for working with data, no matter which database you use.

5️⃣ How SQL Works

- SQL acts as a **communication medium** between the **application** and the **database**.
-
- These programming languages connect to the database through **connectors or drivers**.

🧩 SQL is not the only way to access the database

- While **SQL** is the **standard language** to interact with databases (for storing, retrieving, and managing data), it's **not the only way** applications or programs access the database.
- Other **programming languages** like **Python**, **Java**, **C#**, **PHP**, etc., can also **connect and communicate** with databases — but they still use **SQL commands** *indirectly* inside their code.

💻 How it works

- These programming languages don't talk to the database directly.
- Instead, they use **connectors, drivers, or APIs** (Application Programming Interfaces) that act as **a bridge** between the language and the database.

In summary:

Concept	Description

SQL	The universal language for interacting with databases
Programming Language (like Java/Python)	Uses connectors or APIs to send SQL commands
JDBC	A specific protocol for Java to talk to databases
Result	Allows apps and websites to store, fetch, or update data dynamically

---

6️⃣ JDBC – Java Database Connectivity

- **JDBC** stands for **Java Database Connectivity**.
- It is a **protocol** (a set of rules) that allows Java applications to **connect and communicate with a database** using SQL queries.

Example: When a Java program wants to fetch user details from a MySQL database, it uses JDBC.

Note: JDBC is not only for JAVA, even Python can also use JDBC

7️⃣ UI (User Interface) vs Direct SQL Access

- The **UI** (frontend) is not the only way to send data to a database.
- You can also:

- **Insert** data directly using SQL commands (e.g., `INSERT INTO table_name ...`)
- **Update** data using SQL queries
- **Load files** (like CSVs or Excel files) directly into the database

- This is commonly done by **data engineers, analysts, or database administrators** for faster data management.

8️⃣ Real-World Example: Amazon or Google Pay

- When you make a **transaction** (like paying through Google Pay or shopping on Amazon), the details are sent from the app to the **database** through an **API** (Application Programming Interface).

- The API uses protocols like **HTTP** or **HTTPS** to safely transfer the data from the frontend (app) to the backend (database).

Example: Amount, user ID, time, and transaction ID all get stored in the database instantly.

HTTP / HTTPS Protocols

Protocol	Full Form	Purpose
HTTP	HyperText Transfer Protocol	Transfers data between client (frontend) and server (backend)
HTTPS	HyperText Transfer Protocol Secure	Same as HTTP, but **secure and encrypted** for safety

---

9️⃣ File System vs Database

File System	Database
Stores data in files or folders (like text files, CSVs)	Stores data in structured tables
No standard way to query data	SQL allows powerful data querying
Data security and consistency are low	High security, consistency, and backup options
Hard to manage when data grows	Efficiently handles large-scale data
Example: Excel files, text documents	Example: MySQL, PostgreSQL, Oracle