# What is SQL?

SQL (**Structured Query Language**) is a standard programming language used to manage and manipulate **relational databases**. It allows users to create, retrieve, update, and delete data efficiently. SQL follows a structured syntax and works with **Relational Database Management Systems (RDBMS)** like MySQL, PostgreSQL, SQL Server, and Oracle.

### Key Features of SQL:

- Easy to learn and use
- Supports large databases
- Provides powerful query capabilities
- Ensures data integrity and security
- Supports **transactions** and **stored procedures**

---

# Importance of SQL in Databases

SQL is essential for managing data stored in relational databases. Its importance includes:

1. **Efficient Data Management** – Enables storing, retrieving, and modifying large datasets efficiently.
2. **Data Integrity & Security** – Ensures data consistency using constraints (`PRIMARY KEY`, `FOREIGN KEY`).
3. **Multi-User Access** – Allows concurrent data access by multiple users.
4. **Data Analytics** – Used for generating reports and analyzing business data.
5. **Integration with Applications** – Works with programming languages like Java, Python, PHP, etc.

---

# SQL vs. NoSQL Databases

SQL databases and NoSQL databases serve different purposes in data management.

| Feature | SQL (Relational DB) | NoSQL (Non-Relational DB) |
|---|---|---|
| **Structure** | Structured (Tables with Rows & Columns) | Unstructured or Semi-Structured (JSON, Key-Value, Document-based) |
| **Schema** | Predefined Schema (Fixed structure) | Dynamic Schema (Flexible structure) |
| **Scalability** | Vertical Scaling (Adding resources to a single server) | Horizontal Scaling (Distributing data across multiple servers) |
| **Query Language** | SQL (Standardized) | Varies (MongoDB Query Language, Cassandra CQL, etc.) |
| **Examples** | MySQL, PostgreSQL, Oracle, SQL Server | MongoDB, Cassandra, Redis, Firebase |

### When to Use SQL?

- Structured, consistent data
- Complex queries and relationships

### When to Use NoSQL?

- Large, rapidly changing datasets
- High scalability needs (e.g., real-time applications)

---

# Types of SQL Commands

SQL commands are divided into **five categories**:

### 1. DDL (Data Definition Language) – Defines the structure of the database

DDL commands modify database objects like **tables, schemas, and indexes**.

| Command | Description |
|---|---|
| `CREATE` | Creates new databases, tables, views, or indexes. |
| `ALTER` | Modifies an existing database object (add/remove columns). |
| `DROP` | Deletes a database, table, or other objects permanently. |
| `TRUNCATE` | Removes all records from a table but keeps its structure. |

**Example**

```
CREATE TABLE Employees (
    EmpID INT PRIMARY KEY,
    Name VARCHAR(50),
    Salary DECIMAL(10,2)
);
```

---

## 2. DML (Data Manipulation Language) – Modifies the data in tables

DML commands deal with **inserting, updating, and deleting records**.

| Command | Description |
|---|---|
| `INSERT` | Adds new records into a table. |
| `UPDATE` | Modifies existing records. |
| `DELETE` | Removes specific records from a table. |

**Example**

```
INSERT INTO Employees (EmpID, Name, Salary) VALUES (101, 'Alice',
50000.00);

UPDATE Employees SET Salary = 55000.00 WHERE EmpID = 101;

DELETE FROM Employees WHERE EmpID = 101;
```

---

## 3. DQL (Data Query Language) – Retrieves data from the database

DQL consists of the **SELECT** statement, used to query data from tables.

| Command | Description |
|---|---|
| `SELECT` | Retrieves data from a table. |

**Example**

```
SELECT * FROM Employees; -- Fetch all records

SELECT Name, Salary FROM Employees WHERE Salary > 40000; -- Fetch specific
columns with condition
```

---

## 4. TCL (Transaction Control Language) – Manages database transactions

TCL commands **ensure data consistency** in case of errors or failures.

| Command | Description |
|---|---|
| `COMMIT` | Saves all changes made in the current transaction. |
| `ROLLBACK` | Undoes changes made in the current transaction. |
| `SAVEPOINT` | Creates a checkpoint to partially roll back a transaction. |

**Example**

```
BEGIN TRANSACTION;
UPDATE Employees SET Salary = 60000 WHERE EmpID = 102;

SAVEPOINT save1;
```

```
UPDATE Employees SET Salary = 65000 WHERE EmpID = 103;

ROLLBACK TO save1; -- Undo only the second update

COMMIT; -- Finalize the transaction
```

---

## 5. DCL (Data Control Language) – Manages user permissions

DCL commands **control access** to database objects.

| Command | Description |
|---|---|
| `GRANT` | Provides specific privileges to users. |
| `REVOKE` | Removes granted privileges from users. |

**Example**

```
GRANT SELECT, INSERT ON Employees TO user1; -- Grant privileges

REVOKE INSERT ON Employees FROM user1; -- Revoke privileges
```

---

SQL is essential for working with relational databases. It consists of different types of commands:

- **DDL** for defining structure
- **DML** for modifying data
- **DQL** for querying data
- **TCL** for managing transactions
- **DCL** for controlling permissions