SQL Syntax and Structure:

SQL Syntax and Structure

SQL (Structured Query Language) is used to interact with databases. It allows us to retrieve, add, modify, and delete data stored in tables. Below are some basic SQL statements and their purposes.

Basic SQL Statements

1.SELECT Statement
- Used to retrieve data from a table.
-Structure:

```
SELECT column1, column2 FROM table_name WHERE condition;
```

-Example:
Get the names of all employees:

```
SELECT name FROM employees;
```

2.INSERT Statement
- Adds new data into a table.
-Structure:

```
INSERT INTO table_name (column1, column2) VALUES (value1, value2);
```

-Example:
Add a new employee:

```
INSERT INTO employees (name, age) VALUES ('Alice', 25);
```

3.UPDATE Statement
- Modifies existing data in a table.
-Structure:

```
UPDATE table_name SET column1 = value1 WHERE condition;
```

-Example:
Update Alice’s age to 26:

UPDATE employees SET age = 26 WHERE name = 'Alice';

4.DELETE Statement
- Removes data from a table.
-Structure:

DELETE FROM table_name WHERE condition;

-Example:
Delete Alice’s record:

DELETE FROM employees WHERE name = 'Alice';

SQL Comments

Comments are notes within your SQL code that are ignored by the database. They help explain the code.

1.Single-line Comments:
Use `--` for single-line comments.
```sql
-- This retrieves all employee names
SELECT name FROM employees;
```

2.Multi-line Comments:
Use `/* */` for multi-line comments.

```
/* This query retrieves all employees
   older than 30 years */
SELECT * FROM employees WHERE age > 30;
```

SQL Data Types

Data types define the kind of data that can be stored in each column of a table. Here are common data types:

1.Numeric Types:
  - `INT` (whole numbers)
  - `DECIMAL(p, s)` or `FLOAT` (numbers with decimals)

2.Text Types:
  - `VARCHAR(n)` (variable-length text)
  - `CHAR(n)` (fixed-length text)
  - `TEXT` (long text)

3.Date and Time Types:
  - `DATE` (YYYY-MM-DD)
  - `TIME` (HH:MM:SS)
  - `DATETIME` (combines date and time)

4.Boolean Type:
  - `BOOLEAN` or `BOOL` (true/false values)

SQL Naming Conventions

1.Table and Column Names:
  - Use meaningful names (e.g., `employees`, `order_details`).
  - Avoid spaces; use underscores (_) instead (`first_name`, not `First Name`).

2.Case Sensitivity:
  - SQL keywords are not case-sensitive (`SELECT` is the same as `select`), but table and column names may be, depending on the database.

3.Consistency:
  - Stick to a single style, like lowercase for all names (`customer_id` instead of `CustomerId`).

4.Avoid Reserved Words:
  - Don’t use SQL keywords (like `SELECT` or `TABLE`) as table or column names.

Summary

- SQL statements like `SELECT`, `INSERT`, `UPDATE`, and `DELETE` are the core of interacting with databases.
- Comments (`--`, `/* */`) help explain your SQL code.
- Data types (e.g., `INT`, `VARCHAR`, `DATE`) define the kind of data stored in tables.
- Use meaningful and consistent naming conventions to keep your database clean and understandable.