

# **Introduction to SQL**

## **DDL and DML: The Fundamental Commands**

# Lesson Objectives

- Understand the distinction between **DDL** and **DML**
- Learn the fundamental SQL commands
- See minimal and clear examples for each command

# Types of SQL Commands

## DDL - Data Definition Language

Used to define and modify the structure of tables

- CREATE TABLE
- ALTER TABLE
- DROP TABLE

# **DML - Data Manipulation Language**

Used to manipulate data within tables

- INSERT INTO
- SELECT
- UPDATE
- DELETE

# DDL - CREATE TABLE

Used to create new tables

# Components of CREATE TABLE

- Table name
- List of columns
- Data type for each column
- Optional constraints (e.g. PRIMARY KEY, NOT NULL)


# Generic Syntax

```
CREATE TABLE table_name (  
    column_name type [constraints],  
    ...  
);
```

# Example

```
CREATE TABLE Books (  
    id SERIAL PRIMARY KEY,  
    title VARCHAR(100)  
);
```

# SQLite - CREATE TABLE Example

 **Difference:** SERIAL is not available in SQLite. Use INTEGER PRIMARY KEY – SQLite automatically assigns an auto-incremented value for this column.

```
CREATE TABLE Books (  
    id INTEGER PRIMARY KEY,  
    title VARCHAR(100)  
);
```

# DDL - ALTER TABLE

Used to modify the structure of a table

# Components of ALTER TABLE

- Table name
- Type of modification (ADD COLUMN, DROP COLUMN, RENAME, etc.)
- Definition of the modification

# Generic Syntax

```
ALTER TABLE table_name  
ADD COLUMN column_name type;
```

# Example

```
ALTER TABLE Books  
ADD COLUMN author VARCHAR(100);
```

# DDL - DROP TABLE

Used to delete a table and its data

# Components of DROP TABLE

- Name of the table to delete
- Optionally IF EXISTS

# Generic Syntax

```
DROP TABLE [IF EXISTS] table_name;
```

# Example

```
DROP TABLE Books;
```

# **DML - INSERT INTO**

Used to insert data into a table

# Components of INSERT INTO

- Table name
- List of columns
- List of values to insert (in the same order)

# Generic Syntax

```
INSERT INTO table_name (col1, col2)  
VALUES (val1, val2);
```

# Example

```
INSERT INTO Books (title, author)
VALUES ('1984', 'George Orwell');
```

# DML - SELECT

Used to query data present in a table

# Components of SELECT

- List of columns to display
- Table name
- Optional clauses: WHERE, ORDER BY, GROUP BY, HAVING

# Generic Syntax

```
SELECT col1, col2  
FROM table_name  
[WHERE condition];
```

# Example

```
SELECT title, author  
FROM Books;
```

# **DML - UPDATE**

Used to modify existing data

# Components of UPDATE

- Table name
- List of changes to apply
- WHERE clause to limit the changes

# Generic Syntax

```
UPDATE table_name  
SET col1 = new_value  
WHERE condition;
```

# Example

```
UPDATE Books  
SET author = 'G. Orwell'  
WHERE title = '1984';
```

# DML - DELETE

Used to remove rows from a table

# Components of DELETE

- Table name
- WHERE clause to specify which rows to delete

# Generic Syntax

```
DELETE FROM table_name  
WHERE condition;
```

# Example

```
DELETE FROM Books  
WHERE title = '1984';
```

