**Lesson 1: Introduction to Databases and SQL**

**Introduction:**

In today's data-driven world, understanding how to store, access, and manipulate data is a key skill for anyone in IT or software development. This module introduces you to the fundamentals of SQL (Structured Query Language) and how relational databases work.

**Objectives:**

By the end of this module, students should be able to:

* Define SQL and understand its role in data management.
* Differentiate between DBMS and RDBMS.
* Identify the key components of a database: tables, rows, columns, primary and foreign keys.
* Write and execute basic SQL queries.

**Lesson:**

**What is SQL?**

Definition: SQL (Structured Query Language) is a standard language for accessing and manipulating databases.

Purpose: It allows users to create, retrieve, update, and delete data in a database.

Common SQL Commands:

* SELECT – retrieve data
* INSERT – add new data
* UPDATE – modify existing data
* DELETE – remove data
* CREATE, DROP, ALTER – manage database structure

**Tables, Rows, Columns, Keys**

Table: A collection of related data in rows and columns.

Row (Record): A single entry in a table.

Column (Field): A specific attribute or field within a table.

Primary Key: A unique identifier for each record (e.g., student\_id).

Foreign Key: A column that creates a link between two tables (e.g., course\_id in students that refers to courses).

**Access XAMPP**

* **For Windows -** XAMPP on Windows is typically installed in C:\xampp.
* **Steps:**
  + Open Command Prompt:
    - Press Win + R, type cmd, and hit Enter.
  + Navigate to XAMPP directory:
    - cd C:\xampp
  + Access MySQL from terminal:
    - cd C:\xampp\mysql\bin
    - mysql -u root -p
* **For MAC -** XAMPP is installed in /Applications/XAMPP
* **Steps:**
  + Open Terminal:
    - You can search "Terminal" via Spotlight (Cmd + Space).
  + Navigate to XAMPP directory:
    - cd /Applications/XAMPP
  + Access MySQL CLI:
    - sudo /Applications/XAMPP/xamppfiles/bin/mysql -u root -p

**Database**

* **Show all database**
  + SHOW DATABASES;

You’ll see output like:

A screenshot of a computer program

Description automatically generated

* **Create database**
  + CREATE DATABASE <database\_name>;
* **Delete database**
  + DROP DATABASE <database\_name>;
* **Use database**
  + USE <database\_name>**;**

**Table**

* **Show all tables**
  + SHOW TABLES;
* **Show table structure**
  + DESCRIBE <table\_name>
* **Delete a table**
  + DROP TABLE <table\_name>;
* **Rename a table**
  + RENAME TABLE <table old\_name> TO <table new\_name>;
* **Create a table**
  + *CREATE TABLE students (*

*id INT AUTO\_INCREMENT PRIMARY KEY,*

*name VARCHAR(100),*

*age INT,*

*email VARCHAR(100)*

*);*

**Data Manipulation**

* **Insert data**
  + INSERT INTO <table\_name> (col1, col2) VALUES (val1, val2);
* **View table data**
  + SELECT \*

FROM <table\_name>;

* **Update data**
  + UPDATE <table\_name>

SET col1 = val

WHERE condition;

* **Delete data**
  + DELETE FROM <table\_name>

WHERE condition;

**Activity – Lesson 1**

1. Create a database named **school\_db**.
2. Show all databases.
3. Use the database you have created.
4. Create a table named **students** with the following columns:

|  |  |  |
| --- | --- | --- |
| Column name | Data Type | Notes |
| id | INT | Auto increment, primary key |
| name | VARCHAR(100) |  |
| age | INT |  |
| email | VARCHAR(100) |  |
| course | VARCHAR(100) |  |

1. Show table structure
2. Insert 3 students into the table.
   1. Alice Johnson / 20 / [alice@example.com](mailto:alice@example.com) / BSCS
   2. Bob Smith / 22 / [bob@example.com](mailto:bob@example.com) / BSIT
   3. Clara Davis / 21 / [clara@example.com](mailto:clara@example.com) / BSEMC
3. Display all records in the **students** table.
4. Display only names and emails.
5. Display students older than 20.
6. Change Clara’s course to Data Science.
7. Delete the student named Bob Smith.