**DBMS Practical No: 2**

**# Write SQL queries using Insert, Select, Update, delete with**

**operators, functions, and set operator etc. Use of SQL**

**objects such as Table, View, Index, Sequence, Synonym,**

**different constraints etc.**

**Step 1: Set up your database**

Before you can perform SQL operations, you need a database. You can use an existing database or create a new one. For demonstration purposes, let's assume you have a database named "mydb."

sql

**CREATE DATABASE mydb;**

**USE mydb;**

**Step 2: Create a Table**

To store data, you'll need a table. Let's create a simple table for an example:

sql

**CREATE TABLE employees (**

**employee\_id INT AUTO\_INCREMENT PRIMARY KEY,**

**first\_name VARCHAR(50),**

**last\_name VARCHAR(50),**

**salary DECIMAL(10, 2)**

**);**

This creates a table named "employees" with columns for employee information.

**Step 3: Insert Data**

Now, let's add some data to the "employees" table:

sql

**INSERT INTO employees (first\_name, last\_name, salary)**

**VALUES ('John', 'Doe', 50000.00),**

**('Jane', 'Smith', 60000.00),**

**('Bob', 'Johnson', 55000.00);**

This SQL statement inserts three rows of employee data into the "employees" table.

**Step 4: Select Data**

You can retrieve data from the table using the SELECT statement. For example, to retrieve all employee names and salaries:

sql

**SELECT first\_name, last\_name, salary**

**FROM employees;**

**Step 5: Update Data**

If you need to modify existing data, you can use the UPDATE statement. For example, to increase John's salary:

sql

**UPDATE employees**

**SET salary = salary + 5000**

**WHERE first name = 'John';**

**This increases John's salary by $5000.**

**Step 6: Delete Data**

To remove data from the table, you can use the DELETE statement. For example, to delete Bob's record:

sql

**DELETE FROM employees**

**WHERE first\_name = 'Bob';**

**Step 7: SQL Operators and Functions**

You can use various operators and functions in your SQL queries. Here are some examples:

Arithmetic Operators: +, -, \*, /

Comparison Operators: =, <, >, <=, >=, <>

Logical Operators: AND, OR, NOT

Functions: SUM(), AVG(), COUNT(), MAX(), MIN(), CONCAT()

For instance, you can calculate the average salary of all employees:

sql

**SELECT AVG(salary)**

**FROM employees;**

**Step 8: SQL Objects**

SQL allows you to create various objects, including Views, Indexes, Sequences, and Synonyms.

**Views: Virtual tables based on the result of a SELECT query.**

**Indexes: Improve the speed of data retrieval from tables.**

**Sequences: Generate unique numeric values.**

**Synonyms: Alias for database objects.**

sql

**CREATE VIEW high\_paid\_employees AS**

**SELECT first\_name, last\_name, salary**

**FROM employees**

**WHERE salary > 55000;**