**Lab Exercise 5 - Understanding and Using Joins in MySQL**

**Objective:**

To practice **joining tables** using different types of joins in MySQL:

* INNER JOIN
* LEFT JOIN
* RIGHT JOIN
* FULL OUTER JOIN (simulated)
* CROSS JOIN (Cartesian product)

**Software Required:**

* MySQL Server / MySQL Workbench / phpMyAdmin
* SQL editor or terminal access

**Part A: Setup – Creating Tables**

**Task 1: Create Database**

CREATE DATABASE JoinLab;

USE JoinLab;

**Task 2: Create Tables**

**Table: Departments**

CREATE TABLE Departments (

dept\_id INT PRIMARY KEY,

dept\_name VARCHAR(50)

);

**Table: Employees**

CREATE TABLE Employees (

emp\_id INT PRIMARY KEY,

emp\_name VARCHAR(50),

dept\_id INT,

salary DECIMAL(10,2),

FOREIGN KEY (dept\_id) REFERENCES Departments(dept\_id)

);

**Task 3: Insert Sample Data**

**Insert into Departments**

INSERT INTO Departments VALUES

(1, 'HR'),

(2, 'Finance'),

(3, 'Engineering'),

(4, 'Sales');

**Insert into Employees**

INSERT INTO Employees VALUES

(101, 'Alice', 1, 50000),

(102, 'Bob', 2, 55000),

(103, 'Carol', 3, 60000),

(104, 'David', NULL, 45000),

(105, 'Eve', 5, 70000); -- dept\_id 5 doesn't exist

**Part B: Practicing Joins**

**Task 4: INNER JOIN (Only Matching Rows)**

SELECT emp\_name, dept\_name

FROM Employees

INNER JOIN Departments ON Employees.dept\_id = Departments.dept\_id;

**Task 5: LEFT JOIN (All from Left Table + Matches)**

SELECT emp\_name, dept\_name

FROM Employees

LEFT JOIN Departments ON Employees.dept\_id = Departments.dept\_id;

**Task 6: RIGHT JOIN (All from Right Table + Matches)**

SELECT emp\_name, dept\_name

FROM Employees

RIGHT JOIN Departments ON Employees.dept\_id = Departments.dept\_id;

**Task 7: FULL OUTER JOIN (Simulated Using UNION)**

SELECT emp\_name, dept\_name

FROM Employees

LEFT JOIN Departments ON Employees.dept\_id = Departments.dept\_id

UNION

SELECT emp\_name, dept\_name

FROM Employees

RIGHT JOIN Departments ON Employees.dept\_id = Departments.dept\_id;

**Part C: Cross Join (Cartesian Product)**

**Task 8: CROSS JOIN**

SELECT emp\_name, dept\_name

FROM Employees

CROSS JOIN Departments;

This returns **m × n rows**, where m = number of employees, n = number of departments.

**Learning Outcomes:**

After completing this lab, you will:

* Understand different types of joins and when to use them
* Perform inner, outer, and cross joins in MySQL
* Analyze how data is retrieved and matched across tables
* Recognize unmatched records using outer joins