**Lab Exercise 6 - Relationships and Foreign Keys in MySQL**

**Objective:**

To understand and practice establishing relationships between tables in SQL using:

* **Foreign keys**
* **One-to-One**
* **One-to-Many**
* **Many-to-Many** relationships

**Software Required:**

* MySQL Server / MySQL Workbench / phpMyAdmin
* SQL client or command-line terminal

**Part A: Create Database and Use It**

CREATE DATABASE RelationshipLab;

USE RelationshipLab;

**Part B: One-to-One Relationship**

**Scenario: Each employee has one address.**

**Task 1: Create Tables**

CREATE TABLE Employee (

emp\_id INT PRIMARY KEY,

emp\_name VARCHAR(50)

);

CREATE TABLE Address (

address\_id INT PRIMARY KEY,

emp\_id INT UNIQUE,

city VARCHAR(50),

country VARCHAR(50),

FOREIGN KEY (emp\_id) REFERENCES Employee(emp\_id)

);

**Task 2: Insert Data**

INSERT INTO Employee VALUES (1, 'Alice'), (2, 'Bob');

INSERT INTO Address VALUES (101, 1, 'New York', 'USA'), (102, 2, 'London', 'UK');

**Task 3: Query**

SELECT e.emp\_name, a.city, a.country

FROM Employee e

JOIN Address a ON e.emp\_id = a.emp\_id;

**Part C: One-to-Many Relationship**

**Scenario: A department can have many employees.**

**Task 4: Create Tables**

CREATE TABLE Department (

dept\_id INT PRIMARY KEY,

dept\_name VARCHAR(50)

);

CREATE TABLE Employee2 (

emp\_id INT PRIMARY KEY,

emp\_name VARCHAR(50),

dept\_id INT,

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

);

**Task 5: Insert Data**

INSERT INTO Department VALUES (1, 'HR'), (2, 'Engineering');

INSERT INTO Employee2 VALUES

(1, 'Alice', 1),

(2, 'Bob', 2),

(3, 'Carol', 2);

**Task 6: Query**

SELECT e.emp\_name, d.dept\_name

FROM Employee2 e

JOIN Department d ON e.dept\_id = d.dept\_id;

**Part D: Many-to-Many Relationship**

**Scenario: A student can enroll in many courses, and a course can have many students.**

**Task 7: Create Tables**

CREATE TABLE Student (

student\_id INT PRIMARY KEY,

student\_name VARCHAR(50)

);

CREATE TABLE Course (

course\_id INT PRIMARY KEY,

course\_name VARCHAR(50)

);

CREATE TABLE Enrollment (

student\_id INT,

course\_id INT,

PRIMARY KEY (student\_id, course\_id),

FOREIGN KEY (student\_id) REFERENCES Student(student\_id),

FOREIGN KEY (course\_id) REFERENCES Course(course\_id)

);

**Task 8: Insert Data**

INSERT INTO Student VALUES (1, 'Alice'), (2, 'Bob');

INSERT INTO Course VALUES (101, 'Math'), (102, 'Science');

INSERT INTO Enrollment VALUES

(1, 101),

(1, 102),

(2, 101);

**Task 9: Query**

SELECT s.student\_name, c.course\_name

FROM Enrollment e

JOIN Student s ON e.student\_id = s.student\_id

JOIN Course c ON e.course\_id = c.course\_id;

**Learning Outcomes:**

By the end of this lab, you will be able to:

* Create and use **foreign keys** to link tables
* Implement **1:1**, **1:N**, and **M:N** relationships in MySQL
* Use **junction tables** for many-to-many relationships
* Write SQL joins to retrieve related data