Drop database if exists relationship;

Create database relationship;

Use relationship;

#-----------------------------------

-- Cars table first

CREATE TABLE Cars (

car\_id INT PRIMARY KEY,

car\_model VARCHAR(100),

car\_registration\_number VARCHAR(50)

);

-- Employees table

CREATE TABLE Employees (

employee\_id INT PRIMARY KEY,

employee\_name VARCHAR(100),

car\_id INT UNIQUE, -- Each employee gets exactly one car

FOREIGN KEY (car\_id) REFERENCES Cars(car\_id)

);

-- Insert Cars

INSERT INTO Cars VALUES

(1, 'Toyota Corolla', 'MH12AB1234'),

(2, 'Honda City', 'MH14XY5678'),

(3, 'Hyundai i20', 'MH01ZZ4321');

-- Insert Employees (each has one unique car)

INSERT INTO Employees VALUES

(101, 'Amit Sharma', 1),

(102, 'Riya Mehta', 2),

(103, 'Rahul Verma', 3);

SELECT e.employee\_id, e.employee\_name, c.car\_model, c.car\_registration\_number

FROM Employees e

JOIN Cars c ON e.car\_id = c.car\_id;

#------------------------------------------------------------------------

# 1 to many

CREATE TABLE Authors (

author\_id INT PRIMARY KEY,

author\_name VARCHAR(100)

);

CREATE TABLE Books (

book\_id INT PRIMARY KEY,

book\_title VARCHAR(200),

author\_id INT,

FOREIGN KEY (author\_id) REFERENCES Authors(author\_id)

);

-- Insert Authors

INSERT INTO Authors VALUES

(1, 'J.K. Rowling'),

(2, 'Ruskin Bond');

-- Insert Books

INSERT INTO Books VALUES

(101, 'Harry Potter and the Sorcerer''s Stone', 1),

(102, 'Harry Potter and the Chamber of Secrets', 1),

(103, 'The Room on the Roof', 2);

SELECT a.author\_name, b.book\_title

FROM Authors a

JOIN Books b ON a.author\_id = b.author\_id;

#----------------------------------------------------------------------------------

CREATE TABLE Students (

student\_id INT PRIMARY KEY,

student\_name VARCHAR(100)

);

CREATE TABLE Courses (

course\_id INT PRIMARY KEY,

course\_name VARCHAR(100)

);

-- Junction table

CREATE TABLE Enrollments (

student\_id INT,

course\_id INT,

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

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

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

);

-- Students

INSERT INTO Students VALUES

(1, 'Amit'),

(2, 'Riya'),

(3, 'Rahul');

-- Courses

INSERT INTO Courses VALUES

(101, 'Mathematics'),

(102, 'Science');

-- Enrollments (Many-to-Many)

INSERT INTO Enrollments VALUES

(1, 101), -- Amit → Mathematics

(1, 102), -- Amit → Science

(2, 101), -- Riya → Mathematics

(3, 102); -- Rahul → Science

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

FROM Enrollments e

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

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