**6-1: Cross Joins and Natural Join**

D.suryakiran

192325057

CSA0560

CREATE TABLE departments (

department\_id INT PRIMARY KEY,

department\_name VARCHAR(50),

location\_id INT

);

**-- Insert data into the departments table**

INSERT INTO departments (department\_id, department\_name, location\_id) VALUES (10, 'HR', 100);

INSERT INTO departments (department\_id, department\_name, location\_id) VALUES (20, 'IT', 101);

INSERT INTO departments (department\_id, department\_name, location\_id) VALUES (30, 'Finance', 102);

INSERT INTO departments (department\_id, department\_name, location\_id) VALUES (50, 'Sales', 103);

CREATE TABLE employees (

employee\_id INT PRIMARY KEY,

last\_name VARCHAR(50),

department\_id INT

);

-- Insert data into the employees table

INSERT INTO employees (employee\_id, last\_name, department\_id) VALUES (1, 'Smith', 10);

INSERT INTO employees (employee\_id, last\_name, department\_id) VALUES (2, 'Johnson', 20);

INSERT INTO employees (employee\_id, last\_name, department\_id) VALUES (3, 'Williams', 30);

INSERT INTO employees (employee\_id, last\_name, department\_id) VALUES (4, 'Brown', 50);

CREATE TABLE locations (

location\_id INT PRIMARY KEY,

city VARCHAR(50)

**);**

**-- Insert data into the locations table**

INSERT INTO locations (location\_id, city) VALUES (100, 'New York');

INSERT INTO locations (location\_id, city) VALUES (101, 'Los Angeles');

INSERT INTO locations (location\_id, city) VALUES (102, 'Chicago');

INSERT INTO locations (location\_id, city) VALUES (103, 'San Francisco');

**1. Create a cross-join that displays the last name and department name from the employees and departments tables.**

SELECT employees.last\_name, departments.department\_name

FROM employees

CROSS JOIN departments;

**2. Create a query that uses a natural join to join the departments table and the locations table. Display the department id, department name, location id, and city.**

SELECT departments.department\_id, departments.department\_name, locations.location\_id, locations.city

FROM departments

NATURAL JOIN locations;

**3. Create a query that uses a natural join to join the departments table and the locations table. Restrict the output to only department IDs of 20 and 50. Display the department id, department name, location id, and city**

SELECT departments.department\_id, departments.department\_name, locations.location\_id, locations.city

FROM departments

NATURAL JOIN locations

WHERE departments.department\_id IN (20, 50);