**LAB13**

**L1S23BSCS0195**

**D6**

**CODE:**

-- -- -- --

-- LAB 13

-- -- -- --

CREATE DATABASE LAB13;

USE LAB13;

-- -- -- --

-- Create Tables

-- -- -- --

-- Students Table

CREATE TABLE Students (

student\_id INT PRIMARY KEY,

name VARCHAR(50),

department VARCHAR(50),

CGPA DECIMAL(3, 2)

);

-- Employees Table

CREATE TABLE Employees (

employee\_id INT PRIMARY KEY,

name VARCHAR(50),

department VARCHAR(50),

basic\_salary DECIMAL(10, 2),

allowance DECIMAL(10, 2)

);

-- Departments Table

CREATE TABLE Departments (

department\_id INT PRIMARY KEY,

department\_name VARCHAR(50)

);

-- Products Table

CREATE TABLE Products (

product\_id INT PRIMARY KEY,

product\_name VARCHAR(50),

category VARCHAR(50),

price DECIMAL(10, 2)

);

-- -- -- --

-- Insert Data

-- -- -- --

-- Insert data into Students Table

INSERT INTO Students (student\_id, name, department, CGPA) VALUES

(1, 'Alice', 'Computer Science', 3.9),

(2, 'Bob', 'Electrical Engineering', 2.8),

(3, 'Charlie', 'Computer Science', 3.4),

(4, 'David', 'Mechanical Engineering', 2.5),

(5, 'Eve', 'Computer Science', 3.7);

-- Insert data into Employees Table

INSERT INTO Employees (employee\_id, name, department, basic\_salary, allowance) VALUES

(101, 'John Doe', 'Finance', 5000.00, 1500.00),

(102, 'Jane Smith', 'IT', 6000.00, 2000.00),

(103, 'Emily Johnson', 'Finance', 5500.00, 1800.00),

(104, 'Michael Brown', 'HR', 4500.00, 1200.00),

(105, 'Sarah Davis', 'IT', 7000.00, 2500.00);

-- Insert data into Departments Table

INSERT INTO Departments (department\_id, department\_name) VALUES

(1, 'Finance'),

(2, 'IT'),

(3, 'HR'),

(4, 'Mechanical Engineering'),

(5, 'Electrical Engineering'),

(6, 'Computer Science');

-- Insert data into Products Table

INSERT INTO Products (product\_id, product\_name, category, price) VALUES

(1, 'Laptop', 'Electronics', 1200.00),

(2, 'Smartphone', 'Electronics', 800.00),

(3, 'Tablet', 'Electronics', 400.00),

(4, 'Headphones', 'Accessories', 150.00),

(5, 'Charger', 'Accessories', 30.00);

-- -- -- --

-- Views

-- -- -- --

-- Question 1:

CREATE VIEW StudentView AS

SELECT student\_id, name, CGPA

FROM Students

WHERE CGPA > 3.0;

-- Question 2:

CREATE VIEW EmployeeSalaryView AS

SELECT employee\_id, name, department, (basic\_salary + allowance) AS total\_salary

FROM Employees;

-- Question 3:

CREATE VIEW StudentViewWithGrade AS

SELECT student\_id, name,

CASE

WHEN CGPA >= 3.7 THEN 'Excellent'

WHEN CGPA >= 3.0 THEN 'Good'

ELSE 'Average'

END AS grade

FROM Students;

-- -- -- --

-- Nested Queries

-- -- -- --

-- Question 1:

SELECT name

FROM Students

WHERE CGPA = (SELECT MAX(CGPA) FROM Students);

-- Question 2:

SELECT name

FROM Employees e1

WHERE (basic\_salary + allowance) > (

SELECT AVG(basic\_salary + allowance)

FROM Employees e

);

-- Question 3:

SELECT department

FROM Employees

GROUP BY department

ORDER BY SUM(basic\_salary + allowance) DESC

LIMIT 1;

-- -- -- --

-- Correlated Queries

-- -- -- --

-- Question 1:

SELECT name

FROM Students s1

WHERE CGPA > (

SELECT AVG(CGPA)

FROM Students s2

WHERE s1.department = s2.department

);

-- Question 2:

SELECT name

FROM Employees e1

WHERE (basic\_salary + allowance) > (

SELECT AVG(basic\_salary + allowance)

FROM Employees e

WHERE e1.department = e.department

);

-- Question 3:

SELECT product\_name

FROM Products p1

WHERE price > (

SELECT AVG(price)

FROM Products p2

WHERE p1.category = p2.category

);

