**Assignment 2**

**1.** Create Tables with Keys and Constraints.

CREATE TABLE Branch (branch\_name VARCHAR(50) PRIMARY KEY, branch\_city VARCHAR(50) NOT NULL, assets DECIMAL(15, 2) NOT NULL);

CREATE TABLE Customer (c\_id VARCHAR(10) PRIMARY KEY, cname VARCHAR(50) NOT NULL, street VARCHAR(100), city VARCHAR(50) NOT NULL);

CREATE TABLE Account (account\_no VARCHAR(10) PRIMARY KEY, branch\_name VARCHAR(50), balance DECIMAL(12, 2) NOT NULL, FOREIGN KEY (branch\_name) REFERENCES Branch(branch\_name));

**2.** Create a View with Joins and Filters.

CREATE OR REPLACE VIEW vw\_Pune\_High\_Balance AS SELECT c.c\_id, c.cname, a.account\_no, a.balance FROM Customer c JOIN Depositor d ON c.c\_id = d.c\_id JOIN Account a ON d.account\_no = a.account\_no WHERE a.branch\_name LIKE '%Pune%' AND a.balance > 20000;

**3.** Insert Data into Specific Columns.

INSERT INTO Customer (c\_id, cname, city) VALUES ('C015', 'Simran Singh', 'Delhi');

**4.** Select Data Using Multiple Joins.

SELECT c.cname FROM Customer c JOIN Borrower b ON c.c\_id = b.c\_id JOIN Loan l ON b.loan\_no = l.loan\_no JOIN Branch br ON l.branch\_name = br.branch\_name WHERE br.branch\_city = 'Mumbai';

**5.** Update Records Based on a Condition.

UPDATE Loan SET amount = amount + 100000 WHERE loan\_no IN (SELECT loan\_no FROM Borrower WHERE c\_id = '1211');

**6.** Delete Records Using a Subquery.

DELETE FROM Depositor WHERE account\_no IN (SELECT account\_no FROM Account WHERE balance BETWEEN 30000 AND 80000);

**7.** Use an Aggregate Function (AVG).

SELECT AVG(balance) AS average\_balance FROM Account WHERE branch\_name = 'XYZ';

**8.** Filter and Sort Data (LIKE, ORDER BY).

SELECT cname, city, street FROM Customer WHERE cname LIKE 'R%' ORDER BY cname ASC;

**9.** Combine Results with a Set Operator (UNION).

SELECT c.cname FROM Customer c JOIN Depositor d ON c.c\_id = d.c\_id JOIN Account a ON d.account\_no = a.account\_no WHERE a.branch\_name = 'XYZ' UNION SELECT c.cname FROM Customer c JOIN Depositor d ON c.c\_id = d.c\_id JOIN Account a ON d.account\_no = a.account\_no WHERE a.branch\_name = 'ABC';

**10.** Emulate a MINUS Operation with a Subquery.

SELECT c.\* FROM Customer c WHERE c.c\_id IN (SELECT d.c\_id FROM Depositor d JOIN Account a ON d.account\_no = a.account\_no WHERE a.branch\_name = 'ABC') AND c.c\_id NOT IN (SELECT b.c\_id FROM Borrower b);

**11.** Count the Total Number of Customers in a Branch.

SELECT COUNT(DISTINCT d.c\_id) AS total\_customers FROM Depositor d JOIN Account a ON d.account\_no = a.account\_no WHERE a.branch\_name = 'ABC';