SQL Assignment – Bank Management Scenario

---

1. Table Creation Queries

1.1 Create Bank Table

```
CREATE TABLE Bank (
    branch_id INT PRIMARY KEY,
    branch_name VARCHAR(100),
    branch_city VARCHAR(100)
);
```

1.2 Create Account Holder Table

```
CREATE TABLE Account_Holder (
    account_holder_id INT PRIMARY KEY,
    account_no VARCHAR(20),
    account_holder_name VARCHAR(100),
    city VARCHAR(100),
    contact VARCHAR(15),
    date_of_account_created DATE,
    account_status VARCHAR(20),
    account_type VARCHAR(20),
    balance DECIMAL(10,2)
);
```

1.3 Create Loan Table

```
CREATE TABLE Loan (
    loan_no INT PRIMARY KEY,
    branch_id INT,
    account_holder_id INT,
    loan_amount DECIMAL(10,2),
    loan_type VARCHAR(50),
```

```
    FOREIGN KEY (branch_id) REFERENCES Bank(branch_id),
    FOREIGN KEY (account_holder_id) REFERENCES Account_Holder(account_holder_id)
);
```

---

2. SQL Transaction for Fund Transfer (Account A to Account B)

```
START TRANSACTION;

-- Deduct from Account A
UPDATE Account_Holder
SET balance = balance - 100
WHERE account_no = 'A';

-- Add to Account B
UPDATE Account_Holder
SET balance = balance + 100
WHERE account_no = 'B';

COMMIT;
```

---

3. Fetch Account Holders from Same City

```
SELECT *
FROM Account_Holder AH1
WHERE EXISTS (
    SELECT 1
    FROM Account_Holder AH2
    WHERE AH1.city = AH2.city
      AND AH1.account_holder_id <> AH2.account_holder_id
);
```

---

4. Accounts Created After 15th of Any Month

```
SELECT account_no, account_holder_name
FROM Account_Holder
WHERE DAY(date_of_account_created) > 15;
```

---

5. City Name and Branch Count

```
SELECT branch_city AS city, COUNT(*) AS Count_Branch
FROM Bank
GROUP BY branch_city;
```

---

6. Join Query: Loan Details of Account Holders

```
SELECT AH.account_holder_id, AH.account_holder_name, L.branch_id, L.loan_amount
FROM Account_Holder AH
JOIN Loan L ON AH.account_holder_id = L.account_holder_id;
```

Guide: How to Answer During Viva

- **Table Creation:** "I created 3 tables with correct relationships using Primary and Foreign keys."
- **Transaction Query:** "I safely transferred $100 between two accounts using SQL transactions."
- **Same City Query:** "I used a subquery with EXISTS to find other users from the same city."
- **15th Day Filter:** "I applied the DAY() function to get accounts made after 15th."
- **Branch Count:** "I used GROUP BY with COUNT() and gave it an alias Count_Branch."
- **Join Query:** "I joined the loan and account holder tables on account_holder_id to fetch the correct details."