PRACTICAL-6 **DATE:**_____

AIM: Join Queries- Inner Join, Outer Join Subqueries- With IN clause, With EXISTS clause

INPUT:-

Step-1 : Create BankCustomers, Accounts, Transactions table.

```
// BankCustomers Table
CREATE TABLE BankCustomers (
        CustomerID INT AUTO_INCREMENT PRIMARY KEY,
        FirstName VARCHAR(50) NOT NULL,
        LastName VARCHAR(50) NOT NULL,
        DOB DATE NOT NULL,
        Gender CHAR(1),
        PhoneNumber VARCHAR(10) NOT NULL
);
// Accounts Table
CREATE TABLE Accounts (
        AccountID INT AUTO_INCREMENT PRIMARY KEY,
        CustomerID INT,
        AccountType VARCHAR(20) NOT NULL,
        Balance DECIMAL(15,2) NOT NULL,
        FOREIGN KEY (CustomerID) REFERENCES
                BankCustomers (CustomerID)
);
// Transactions Table
CREATE TABLE Transactions (
        TransactionID INT AUTO_INCREMENT PRIMARY KEY,
        AccountID INT,
        TransactionDate DATE NOT NULL,
        Amount DECIMAL(15,2) NOT NULL,
        TransactionType VARCHAR(20) NOT NULL,
        FOREIGN KEY (AccountID) REFERENCES
                Accounts (AccountID)
```

Step-2 : Insert data into tables.

```
// BankCustomers Table
INSERT INTO BankCustomers
  (FirstName, LastName, DOB, Gender, PhoneNumber) VALUES
('John', 'Doe', '1980-01-15', 'M', '123456789'),
('Jane', 'Smith', '1985-05-23', 'F', '2345678901'),
('Alan', 'Brown', '1990-08-12', 'M', '3456789012');
// Accounts Table
INSERT INTO Accounts
  (CustomerID, AccountType, Balance) VALUES
(1, 'Savings', 5000.00),
(2, 'Checking', 1500.00),
(1, 'Checking', 2000.00);
// Transactions Table
INSERT INTO Transactions
  (AccountID, TransactionDate, Amount, TransactionType)
VALUES
  (1, '2024-01-10', 500.00, 'Deposit'),
  (2, '2024-01-15', 200.00, 'Withdrawal'),
  (3, '2024-01-20', 300.00, 'Deposit');
```

Step-3 : Fetch the data

```
// BankCustomers Table
SELECT * FROM BankCustomers;
// Accounts Table
SELECT * FROM Accounts;
// Transactions Table
SELECT * FROM Transactions;
```

// Bank Customers Table

CustomerID	FirstName	LastName	DoB	Gender	PhoneNumber
1	John	Doe	1980-01-15	M	1234567890
2	Jane	Smith	1985-05-23	F	2345678901
3	Alan	Brown	1990-08-12	M	3456789012

// Accounts Table

AccountID	CustomerID	AccountType	Balance
1	1	Savings	5000.00
2	2	Checking	1500.00
3	1	Checking	2000.00

// Transactions Table

TransactionID	AccountID	TransactionDate	Amount	TransactionType
1	1	2024-01-10	500.00	Deposit
2	2	2024-01-15	200.00	Withdrawal
3	3	2024-01-20	300.00	Deposit

a). Join Queries :-

i). Inner Join :-

```
SELECT
    BankCustomers.CustomerID,
    BankCustomers.FirstName,
    BankCustomers.LastName,
    Accounts.AccountType,
    Accounts.Balance
FROM
    BankCustomers
INNER JOIN
    Accounts
ON
    BankCustomers.CustomerID =
                    Accounts.CustomerID;
```

CustomerID	FirstName	LastName	AccountType	Balance
1	John	Doe	Savings	5000.00
2	Jane	Smith	Checking	1500.00
1	John	Doe	Checking	2000.00

ii). Left Join :-

```
SELECT
    BankCustomers.CustomerID,
    BankCustomers.FirstName,
    BankCustomers.LastName,
    Accounts.AccountType,
    Accounts.Balance
FROM
    BankCustomers
LEFT JOIN
    Accounts
ON
    BankCustomers.CustomerID = Accounts.CustomerID;
```

CustomerID	FirstName	LastName	AccountType	Balance
1	John	Doe	Savings	5000.00
1	John	Doe	Checking	2000.00
2	Jane	Smith	Checking	1500.00
3	Alan	Brown	NULL	NULL

iii). Right Join :-

```
SELECT
    BankCustomers.CustomerID,
    BankCustomers.FirstName,
    BankCustomers.LastName,
    Accounts.AccountType,
    Accounts.Balance
FROM
    BankCustomers
RIGHT JOIN
    Accounts
ON
    BankCustomers.CustomerID = Accounts.CustomerID ;
```

CustomerID	FirstName	LastName	AccountType	Balance
1	John	Doe	Savings	5000.00
2	Jane	Smith	Checking	1500.00
1	John	Doe	Checking	2000.00

iv). Full-Outer Join :-

```
SELECT
    BankCustomers.CustomerID,
    BankCustomers.FirstName,
    BankCustomers.LastName,
    Accounts.AccountType,
    Accounts.Balance
FROM
    BankCustomers
FULL OUTER JOIN
    Accounts
ON
    BankCustomers.CustomerID =
                Accounts.CustomerID;
```

CustomerID	FirstName	LastName	AccountType	Balance
1	John	Doe	Savings	5000.00
2	Jane	Smith	Checking	1500.00
1	John	Doe	Checking	2000.00
3	Alan	Brocon	NULL	NULL

V). CROSS Join :-

```
SELECT
    BankCustomers.FirstName,
    BankCustomers.LastName,
    Accounts.AccountType,
    Accounts.Balance
FROM
    BankCustomers
CROSS JOIN
    Accounts;
```

FirstName	LastName	AccountType	Balance
John	Doe	Savings	5000.00
John	Doe	Checking	2000.00
Jane	Smith	Savings	5000.00
Jane	Smith	Checking	2000.00
Alan	Brown	Savings	5000.00
Alan	Brown	Checking	2000.00

vii. Self-Join :-

```
SELECT
    a1.AccountID AS AccountID1,
    a1.CustomerID AS CustomerID1,
    a1.AccountType AS AccountType1,
    a1.Balance AS Balance1,
    a2.AccountID AS AccountID2,
    a2.CustomerID AS CustomerID2,
    a2.AccountType AS AccountType2,
    a2.Balance AS Balance2
FROM
    Accounts a1
INNER JOIN
    Accounts a2
ON
    a1.CustomerID = a2.CustomerID
    AND a1.Balance > a2.Balance;
```

AccountID1	CustomerID1	AccountType1	Balance1	AccountID2	CustomerID2
1	1	Savings	5000.00	3	1

AccountType2	Balance2
Checking	2000.00

b). Subqueries with IN and Exists Clause :-

i). IN Clause :-

```
SELECT
    FirstName,
    LastName,
FROM
    BankCustomers
WHERE
    CustomerID IN (
        SELECT DISTINCT CustomerID
        FROM Accounts
        WHERE AccountID IN(
            SELECT AccounID
            FROM Transactions
        )
    );
```

FirstName	LastName
John	Doe
Jane	Smith

ii). Exists Clause :-

```
SELECT
    FirstName,
    LastName
From
    BankCustomers bc
WHERE
    EXISTS (
        Select 1
        FROM Accounts a
```

```
    JOIN Transactions t ON a.AccountID = t.AccountID
    WHERE bc.CustomerID = a.CustomerID
);
```

FirstName	LastName
John	Doe
Jane	Smith