

#### Problem Statement 5 (DML USING MYSQL)

Create following tables using a given schema and insert appropriate data into the same:

Customer (CustID, Name, Cust\_Address, Phone\_no, Age)

Branch (Branch ID, Branch\_Name, Address)

Account (Account\_no, Branch ID, CustID, date\_open, Account\_type, Balance)

1. Add the column "Email\_Address" in Customer table.

2. Change the name of column "Email\_Address" to "Email\_ID" in Customer table.

3. Display the customer details with highest balance in the account.

4. Display the customer details with lowest balance for account type= "Saving Account".

5. Display the customer details that live in Pune and have age greater than 35.

6. Display the Cust\_ID, Name and Age of the customer in ascending order of their age.

7. Display the Name and Branch ID of the customer group by the Account\_type.

-----  
-----

#### Step 1: Create the Database and Tables

Start by creating the database and then create the Customer, Branch, and Account tables according to the given schema.

-- Step 1: Create Database

```
CREATE DATABASE BankDB;
```

```
USE BankDB;
```

-- Step 2: Create Customer Table

```
CREATE TABLE Customer (  
    CustID INT PRIMARY KEY,  
    Name VARCHAR(50),  
    Cust_Address VARCHAR(100),  
    Phone_no VARCHAR(15),  
    Age INT  
);
```

-- Step 3: Create Branch Table

```
CREATE TABLE Branch (  
    Branch_ID INT PRIMARY KEY,  
    Branch_Name VARCHAR(50),  
    Address VARCHAR(100)  
);
```

-- Step 4: Create Account Table

```
CREATE TABLE Account (  
    Account_no INT PRIMARY KEY,  
    Branch_ID INT,  
    CustID INT,  
    date_open DATE,  
    Account_type VARCHAR(20),  
    Balance DECIMAL(10, 2),  
    FOREIGN KEY (Branch_ID) REFERENCES Branch(Branch_ID),  
    FOREIGN KEY (CustID) REFERENCES Customer(CustID)  
);
```

## Step 2: Insert Sample Data into Tables

Populate the Customer, Branch, and Account tables with sample data.

-- Insert sample data into Customer table

```
INSERT INTO Customer (CustID, Name, Cust_Address, Phone_no, Age)
VALUES
(1, 'Alice', 'Pune', '1234567890', 40),
(2, 'Bob', 'Mumbai', '0987654321', 30),
(3, 'Charlie', 'Pune', '1122334455', 45),
(4, 'David', 'Delhi', '2233445566', 28),
(5, 'Eva', 'Pune', '3344556677', 36);
```

-- Insert sample data into Branch table

```
INSERT INTO Branch (Branch_ID, Branch_Name, Address)
VALUES
(1, 'Main', 'Pune'),
(2, 'South', 'Mumbai'),
(3, 'East', 'Delhi');
```

-- Insert sample data into Account table

```
INSERT INTO Account (Account_no, Branch_ID, CustID, date_open, Account_type,
Balance)
VALUES
(1001, 1, 1, '2020-01-10', 'Saving Account', 15000),
(1002, 2, 2, '2019-02-15', 'Current Account', 20000),
(1003, 1, 3, '2021-03-20', 'Saving Account', 5000),
(1004, 3, 4, '2018-04-25', 'Current Account', 8000),
(1005, 1, 5, '2022-05-30', 'Saving Account', 12000);
```

## Step 3: Perform DML Operations

1. Add the Column "Email\_Address" in the Customer Table

```
ALTER TABLE Customer
ADD COLUMN Email_Address VARCHAR(100);
```

2. Change the Name of Column "Email\_Address" to "Email\_ID" in Customer Table

```
ALTER TABLE Customer
CHANGE COLUMN Email_Address Email_ID VARCHAR(100);
```

3. Display the Customer Details with the Highest Balance in the Account  
This query joins the Customer and Account tables to retrieve details of the customer with the highest balance.

```
SELECT C.*
FROM Customer C
JOIN Account A ON C.CustID = A.CustID
ORDER BY A.Balance DESC
LIMIT 1;
```

4. Display the Customer Details with the Lowest Balance for Account Type = "Saving Account"  
This query retrieves the customer with the lowest balance among "Saving Account" types.

```
SELECT C.*  
FROM Customer C  
JOIN Account A ON C.CustID = A.CustID  
WHERE A.Account_type = 'Saving Account'  
ORDER BY A.Balance ASC  
LIMIT 1;
```

5. Display the Customer Details That Live in Pune and Have Age Greater Than 35  
This query filters customers based on their address and age.

```
SELECT *  
FROM Customer  
WHERE Cust_Address = 'Pune' AND Age > 35;
```

6. Display the CustID, Name, and Age of the Customer in Ascending Order of Their Age  
This query retrieves the CustID, Name, and Age of customers ordered by age in ascending order.

```
SELECT CustID, Name, Age  
FROM Customer  
ORDER BY Age ASC;
```

7. Display the Name and Branch\_ID of the Customer Grouped by Account\_type  
This query groups by Account\_type to display the Name and Branch\_ID of each customer.

```
SELECT C.Name, A.Branch_ID  
FROM Customer C  
JOIN Account A ON C.CustID = A.CustID  
GROUP BY A.Account_type, C.Name, A.Branch_ID;
```