**Exercise 3: Stored Procedures**

**Create Tables:**

**Accounts Table:**

CREATE TABLE Accounts (

AccountID INT PRIMARY KEY,

Customer\_Id INT,

AccountType VARCHAR(20), -- e.g., 'Savings', 'Checking'

Balance DECIMAL(12,2)

);

**Employees Table:**

CREATE TABLE Employees (

EmployeeID INT PRIMARY KEY,

Name VARCHAR(100),

Department VARCHAR(50),

Salary DECIMAL(10,2)

);

**Insert into Accounts**

INSERT INTO Accounts (AccountID, Customer\_Id, AccountType, Balance)

VALUES

(101, 1, 'Savings', 5000.00),

(102, 2, 'Savings', 10000.00),

(103, 3, 'Checking', 2000.00),

(104, 1, 'Savings', 7000.00), -- Same customer, second savings

(105, 2, 'Checking', 3000.00);

**Insert into Employees:**

INSERT INTO Employees (EmployeeID, Name, Department, Salary)

VALUES

(1, 'Alice', 'HR', 50000.00),

(2, 'Bob', 'Sales', 60000.00),

(3, 'Charlie', 'HR', 55000.00),

(4, 'David', 'IT', 70000.00);

**Scenario 1: ProcessMonthlyInterest**

DELIMITER //

CREATE PROCEDURE ProcessMonthlyInterest()

BEGIN

-- Apply 1% interest to all Savings accounts

UPDATE Accounts

SET Balance = Balance \* 1.01

WHERE AccountType = 'Savings';

END //

DELIMITER ;

**Scenario 2: UpdateEmployeeBonus (with parameter)**

DELIMITER //

CREATE PROCEDURE UpdateEmployeeBonus(

IN dept\_name VARCHAR(50),

IN bonus\_percent DECIMAL(5,2)

)

BEGIN

-- Give bonus to employees of the specified department

UPDATE Employees

SET Salary = Salary + (Salary \* bonus\_percent / 100)

WHERE Department = dept\_name;

END //

DELIMITER ;

**Scenario 3: TransferFunds**

DELIMITER //

CREATE PROCEDURE TransferFunds(

IN fromAccount INT,

IN toAccount INT,

IN amount DECIMAL(10,2)

)

BEGIN

DECLARE fromBalance DECIMAL(12,2);

-- Get balance of source account

SELECT Balance INTO fromBalance

FROM Accounts

WHERE AccountID = fromAccount;

-- Check if sufficient funds exist

IF fromBalance >= amount THEN

-- Deduct from source

UPDATE Accounts

SET Balance = Balance - amount

WHERE AccountID = fromAccount;

-- Add to destination

UPDATE Accounts

SET Balance = Balance + amount

WHERE AccountID = toAccount;

ELSE

SIGNAL SQLSTATE '45000'

SET MESSAGE\_TEXT = 'Insufficient balance in source account';

END IF;

END //

DELIMITER ;

**Running the procedures:**

SET SQL\_SAFE\_UPDATES = 0;

-- Process Monthly Interest

CALL ProcessMonthlyInterest();

-- Update Bonus for HR by 10%

CALL UpdateEmployeeBonus('HR', 10);

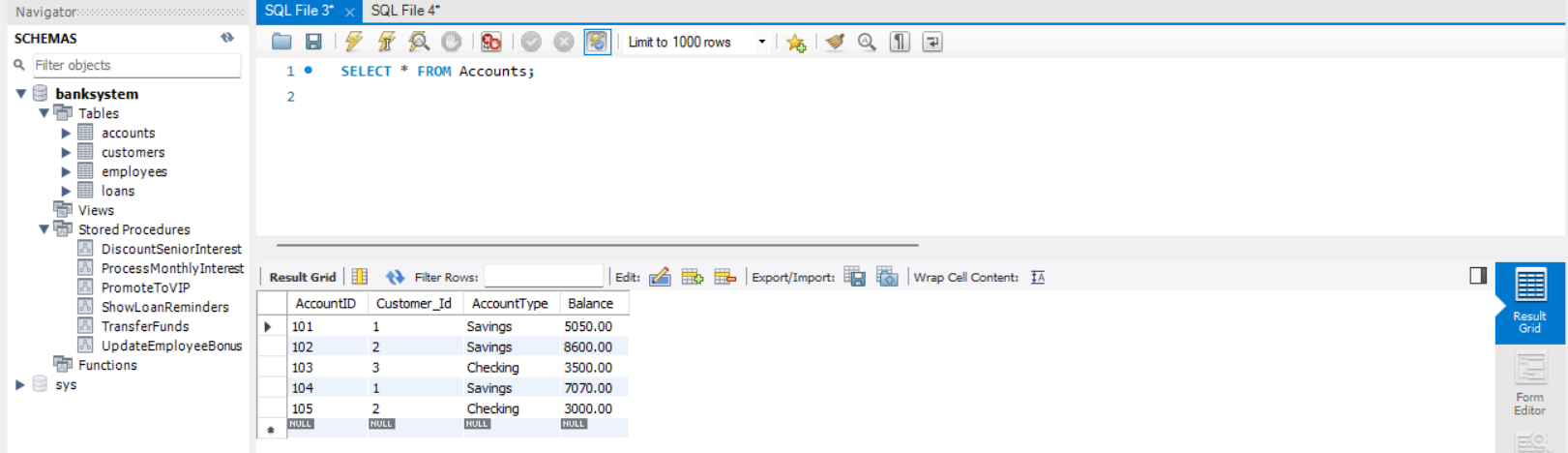
-- Transfer ₹1500 from Account 102 to 103

CALL TransferFunds(102, 103, 1500);

**Verify output changes after implementing the procedures:**

-- Check updated balances

SELECT \* FROM Accounts;



-- Check updated salaries

SELECT \* FROM Employees;

