Exercise 3: Stored Procedures

Program :

CREATE DATABASE bank\_db;

CREATE TABLE Account (

AccountID INT PRIMARY KEY,

AccountType VARCHAR(20),

Balance DECIMAL(10, 2)

);

DROP TABLE IF EXISTS Employee;

CREATE TABLE Employee (

EmployeeID INT PRIMARY KEY,

Name VARCHAR(50),

DepartmentID INT,

Salary DECIMAL(10, 2)

);

DROP TABLE IF EXISTS CustomerAccount;

CREATE TABLE CustomerAccount (

AccountID INT PRIMARY KEY,

CustomerID INT,

Balance DECIMAL(10, 2)

);

INSERT INTO Account VALUES

(1, 'Savings', 10000),

(2, 'Current', 15000),

(3, 'Savings', 20000);

INSERT INTO Employee VALUES

(101, 'Alice', 1, 50000),

(102, 'Bob', 1, 60000),

(103, 'Charlie', 2, 55000);

INSERT INTO CustomerAccount VALUES

(1, 201, 5000),

(2, 202, 3000);

DELIMITER //

CREATE PROCEDURE ProcessMonthlyInterest()

BEGIN

UPDATE Account

SET Balance = Balance + (Balance \* 0.01)

WHERE AccountType = 'Savings';

END //

CREATE PROCEDURE UpdateEmployeeBonus(IN dept\_id INT, IN bonus\_percent DECIMAL(5,2))

BEGIN

UPDATE Employee

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

WHERE DepartmentID = dept\_id;

END //

CREATE PROCEDURE TransferFunds(

IN from\_account INT,

IN to\_account INT,

IN amount DECIMAL(10,2)

)

BEGIN

DECLARE from\_balance DECIMAL(10,2);

SELECT Balance INTO from\_balance

FROM CustomerAccount

WHERE AccountID = from\_account;

IF from\_balance >= amount THEN

UPDATE CustomerAccount

SET Balance = Balance - amount

WHERE AccountID = from\_account;

UPDATE CustomerAccount

SET Balance = Balance + amount

WHERE AccountID = to\_account;

ELSE

SIGNAL SQLSTATE '45000'

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

END IF;

END //

DELIMITER ;

CALL ProcessMonthlyInterest();

CALL UpdateEmployeeBonus(1, 10);

CALL TransferFunds(1, 2, 1000);

SELECT \* FROM Account;

SELECT \* FROM Employee;

SELECT \* FROM CustomerAccount;