**Exercise 3: Stored Procedures**

**Scenario 1:** The bank needs to process monthly interest for all savings accounts.

* + **Question:** Write a stored procedure **ProcessMonthlyInterest** that calculates and updates the balance of all savings accounts by applying an interest rate of 1% to the current balance.

**SCENARIO 1:**

CREATE PROCEDURE ProcessMonthlyInterest()

BEGIN

DECLARE done INT DEFAULT 0;

DECLARE account\_id INT;

DECLARE balance DECIMAL(10,2);

DECLARE interest\_rate DECIMAL(10,2) DEFAULT 0.01;

-- Cursor to select all savings accounts

DECLARE cur CURSOR FOR

SELECT account\_id, balance FROM savings\_accounts;

-- Handler for cursor

DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;

OPEN cur;

read\_loop: LOOP

FETCH cur INTO account\_id, balance;

IF done THEN

LEAVE read\_loop;

END IF;

-- Calculate new balance with interest

SET balance = balance + (balance \* interest\_rate);

-- Update the balance

UPDATE savings\_accounts

SET balance = balance

WHERE account\_id = account\_id;

END LOOP;

CLOSE cur;

END$$

DELIMITER ;

**Scenario 2:** The bank wants to implement a bonus scheme for employees based on their performance.

* + **Question:** Write a stored procedure **UpdateEmployeeBonus** that updates the salary of employees in a given department by adding a bonus percentage passed as a parameter.

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

BEGIN

-- Update the salary by adding the bonus percentage

UPDATE employees

SET salary = salary + (salary \* (bonus\_percentage / 100))

WHERE department\_id = dept\_id;

END;

**Scenario 3:** Customers should be able to transfer funds between their accounts.

* + **Question:** Write a stored procedure **TransferFunds** that transfers a specified amount from one account to another, checking that the source account has sufficient balance before making the transfer.

**TransferFunds**

CREATE PROCEDURE TransferFunds(

IN source\_account\_id INT,

IN target\_account\_id INT,

IN amount DECIMAL(10,2)

)

BEGIN

DECLARE source\_balance DECIMAL(10,2);

-- Check the balance of the source account

SELECT balance INTO source\_balance FROM accounts WHERE account\_id = source\_account\_id;

IF source\_balance >= amount THEN

-- Deduct amount from source account

UPDATE accounts

SET balance = balance - amount

WHERE account\_id = source\_account\_id;

-- Add amount to target account

UPDATE accounts

SET balance = balance + amount

WHERE account\_id = target\_account\_id;

ELSE

-- Handle insufficient funds

SIGNAL SQLSTATE '45000'

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

END IF;

END;