**Create Table :**

CREATE TABLE SavingsAccounts (

AccountID INT PRIMARY KEY,

AccountHolder VARCHAR(100),

Balance DECIMAL(10,2)

);

**Insert Values :**

INSERT INTO SavingsAccounts VALUES

(1, 'Alice', 1000.00),

(2, 'Bob', 2000.00),

(3, 'Charlie', 3000.00);

**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.

**Ans :**

DELIMITER //

CREATE PROCEDURE ProcessMonthlyInterest()

BEGIN

UPDATE SavingsAccounts

SET Balance = Balance + (Balance \* 0.01);

END //

DELIMITER ;

**Output :**

| **AccountID** | **AccountHolder** | **Balance** |
| --- | --- | --- |
| 1 | Alice | 1010.00 |
| 2 | Bob | 2020.00 |
| 3 | Charlie | 3030.00 |

**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.

**Ans :**

CREATE TABLE Employees (

EmpID INT PRIMARY KEY,

EmpName VARCHAR(100),

Department VARCHAR(50),

Salary DECIMAL(10,2)

);

INSERT INTO Employees VALUES

(101, 'John', 'HR', 50000),

(102, 'Jane', 'IT', 60000),

(103, 'Mark', 'IT', 55000),

(104, 'Lucy', 'HR', 52000);

DELIMITER //

CREATE PROCEDURE UpdateEmployeeBonus(

IN deptName VARCHAR(50),

IN bonusPercent DECIMAL(5,2)

)

BEGIN

UPDATE Employees

SET Salary = Salary + (Salary \* bonusPercent / 100)

WHERE Department = deptName;

END //

DELIMITER ;

**Output :**

| EmpID | EmpName | Department | Salary |
| --- | --- | --- | --- |
| 101 | John | HR | 50000.00 |
| 102 | Jane | IT | 66000.00 |
| 103 | Mark | IT | 60500.00 |
| 104 | Lucy | HR | 52000.00 |
|  |  |  |  |

**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.

**Ans :**

CREATE TABLE BankAccounts (

AccountID INT PRIMARY KEY,

AccountHolder VARCHAR(100),

Balance DECIMAL(10,2)

);

INSERT INTO BankAccounts VALUES

(201, 'Ravi', 10000.00),

(202, 'Sneha', 8000.00);

DELIMITER //

CREATE PROCEDURE TransferFunds(

IN fromAcc INT,

IN toAcc INT,

IN amt DECIMAL(10,2)

)

BEGIN

DECLARE fromBal DECIMAL(10,2);

-- Check balance

SELECT Balance INTO fromBal FROM BankAccounts WHERE AccountID = fromAcc;

IF fromBal >= amt THEN

-- Deduct from source

UPDATE BankAccounts

SET Balance = Balance - amt

WHERE AccountID = fromAcc;

-- Add to destination

UPDATE BankAccounts

SET Balance = Balance + amt

WHERE AccountID = toAcc;

ELSE

SIGNAL SQLSTATE '45000'

SET MESSAGE\_TEXT = 'Insufficient balance for transfer';

END IF;

END //

DELIMITER ;

**Output :**

| AccountID | AccountHolder | Balance |
| --- | --- | --- |
| 201 | Ravi | 8000.00 |
| 202 | Sneha | 10000.00 |