**Exercise 4: Functions**

**Scenario 1:** Calculate the age of customers for eligibility checks.

**Question:** Write a function CalculateAge that takes a customer's date of birth as input and returns their age in years.

**Answer:**

DELIMITER //

CREATE FUNCTION CalculateAge(dob DATE)

RETURNS INT

DETERMINISTIC

READS SQL DATA

BEGIN

DECLARE age INT;

SET age = TIMESTAMPDIFF(YEAR, dob, CURDATE());

RETURN age;

END //

DELIMITER ;

**Scenario 2:** The bank needs to compute the monthly installment for a loan.

**Question:** Write a function **CalculateMonthlyInstallment** that takes the loan amount, interest rate, and loan duration in years as input and returns the monthly installment amount.

**Answer:**

DELIMITER //

CREATE FUNCTION CalculateMonthlyInstallment(

loanAmount DECIMAL(10,2),

interestRate DECIMAL(5,2),

loanDurationYears INT

)

RETURNS DECIMAL(10,2)

DETERMINISTIC

READS SQL DATA

BEGIN

DECLARE monthlyRate DECIMAL(5,10);

DECLARE numberOfPayments INT;

DECLARE monthlyInstallment DECIMAL(10,2);

SET monthlyRate = interestRate / 1200;

SET numberOfPayments = loanDurationYears \* 12;

SET monthlyInstallment = loanAmount \* (monthlyRate / (1 - POWER(1 + monthlyRate, -numberOfPayments)));

RETURN monthlyInstallment;

END //

DELIMITER ;

**Scenario 3:** Check if a customer has sufficient balance before making a transaction.

**Question:** Write a function **HasSufficientBalance** that takes an account ID and an amount as input and returns a boolean indicating whether the account has at least the specified amount.

**Answer:**

DELIMITER //

CREATE FUNCTION HasSufficientBalance(accountID INT, amount DECIMAL(10,2))

RETURNS TINYINT(1)

DETERMINISTIC

READS SQL DATA

BEGIN

DECLARE balance DECIMAL(10,2);

-- Retrieve the balance for the given account ID

SELECT Balance INTO balance

FROM Accounts

WHERE AccountID = accountID;

-- Return 1 (TRUE) if balance is sufficient, otherwise 0 (FALSE)

RETURN IF(balance >= amount, 1, 0);

END //

DELIMITER ;