**EXERCISE 4 : FUNCTIONS**

DELIMITER //

-- ==========================================

-- Scenario 1: CalculateAge

-- ==========================================

CREATE FUNCTION CalculateAge(dob DATE)

RETURNS INT

DETERMINISTIC

BEGIN

RETURN TIMESTAMPDIFF(YEAR, dob, CURDATE());

END;

//

-- ==========================================

-- Scenario 2: CalculateMonthlyInstallment

-- ==========================================

-- Formula: EMI = P \* r \* (1 + r)^n / ((1 + r)^n - 1)

-- Where:

-- P = loan amount

-- r = monthly interest rate (annual rate / 12 / 100)

-- n = loan duration in months

CREATE FUNCTION CalculateMonthlyInstallment(

loan\_amount DECIMAL(10,2),

annual\_rate DECIMAL(5,2),

duration\_years INT

)

RETURNS DECIMAL(10,2)

DETERMINISTIC

BEGIN

DECLARE r DECIMAL(10,6);

DECLARE n INT;

DECLARE emi DECIMAL(10,2);

SET r = annual\_rate / 12 / 100;

SET n = duration\_years \* 12;

SET emi = loan\_amount \* r \* POW(1 + r, n) / (POW(1 + r, n) - 1);

RETURN emi;

END;

//

-- ==========================================

-- Scenario 3: HasSufficientBalance

-- ==========================================

CREATE FUNCTION HasSufficientBalance(

acc\_id INT,

required\_amount DECIMAL(10,2)

)

RETURNS BOOLEAN

DETERMINISTIC

BEGIN

DECLARE acc\_balance DECIMAL(10,2);

SELECT Balance INTO acc\_balance

FROM Accounts

WHERE AccountID = acc\_id;

RETURN acc\_balance >= required\_amount;

END;

//

DELIMITER ;

**OUTPUT:**

USE bankdb;

SELECT CalculateAge('1990-06-10');

SELECT CalculateMonthlyInstallment(50000, 7.5, 5);

SELECT HasSufficientBalance(1, 200);





