**Exercise 4: Functions**

**Scenario 1: Calculate customer age**

Query:

CREATE OR REPLACE FUNCTION CalculateAge(p\_date\_of\_birth DATE)

RETURN NUMBER

IS

v\_age NUMBER;

BEGIN

v\_age := TRUNC(MONTHS\_BETWEEN(SYSDATE, p\_date\_of\_birth) / 12);

RETURN v\_age;

END CalculateAge;

**Scenario 2: Calculate monthly installment for a loan**

Query:

CREATE OR REPLACE FUNCTION CalculateMonthlyInstallment(p\_loan\_amount NUMBER, p\_interest\_rate NUMBER, p\_loan\_duration NUMBER)

RETURN NUMBER

IS

v\_monthly\_installment NUMBER;

BEGIN

v\_monthly\_installment := p\_loan\_amount \* (p\_interest\_rate / 100) / (1 - POWER(1 + p\_interest\_rate / 100, -p\_loan\_duration \* 12));

RETURN v\_monthly\_installment;

END CalculateMonthlyInstallment;

**Scenario 3: Check if a customer has sufficient balance**

Query:

CREATE OR REPLACE FUNCTION HasSufficientBalance(p\_account\_id NUMBER, p\_amount NUMBER)

RETURN BOOLEAN

IS

v\_balance NUMBER;

BEGIN

SELECT Balance

INTO v\_balance

FROM Accounts

WHERE AccountID = p\_account\_id;

IF v\_balance >= p\_amount THEN

RETURN TRUE;

ELSE

RETURN FALSE;

END IF;

END HasSufficientBalance;