# Scenario 1:

CREATE OR REPLACE FUNCTION CalculateAge(p\_DOB DATE) RETURN NUMBER IS

v\_Age NUMBER;

BEGIN

v\_Age := ROUND((SYSDATE - p\_DOB) / 365);

RETURN v\_Age;

END CalculateAge;

/

# Scenario 2:

CREATE OR REPLACE FUNCTION CalculateMonthlyInstallment(

p\_LoanAmount NUMBER,

p\_InterestRate NUMBER,

p\_LoanDurationYears NUMBER

) RETURN NUMBER IS

v\_MonthlyInstallment NUMBER;

v\_MonthlyInterestRate NUMBER;

v\_NumPayments NUMBER;

BEGIN

v\_MonthlyInterestRate := p\_InterestRate / 1200;

v\_NumPayments := p\_LoanDurationYears \* 12;

v\_MonthlyInstallment := p\_LoanAmount \* v\_MonthlyInterestRate / (1 - POWER(1 + v\_MonthlyInterestRate, -v\_NumPayments));

RETURN v\_MonthlyInstallment;

END CalculateMonthlyInstallment;

/

# Scenario 3:

CREATE OR REPLACE FUNCTION HasSufficientBalance(

p\_AccountID NUMBER,

p\_Amount NUMBER

) RETURN BOOLEAN IS

v\_Balance NUMBER;

BEGIN

SELECT Balance INTO v\_Balance FROM Accounts WHERE AccountID = p\_AccountID;

RETURN v\_Balance >= p\_Amount;

EXCEPTION

WHEN NO\_DATA\_FOUND THEN

RETURN FALSE;

END HasSufficientBalance;

/