**PL/SQL Exercises**

* **Exercise 3: Stored Procedures**
* ***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.

**Code:**

*PL/SQL Procedure-*

CREATE OR REPLACE PROCEDURE ProcessMonthlyInterest IS

BEGIN

UPDATE Accounts

SET Balance = Balance + (Balance \* 0.01)

WHERE AccountType = 'Savings';

END;

/

*Execution Block-*

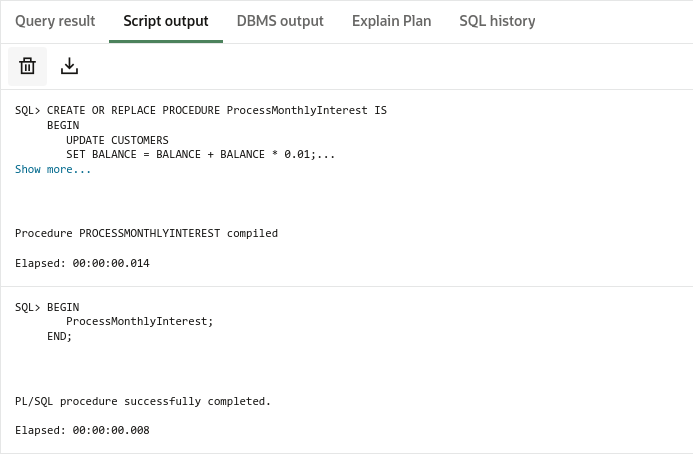
BEGIN

ProcessMonthlyInterest;

END;

/

**Output:**

****

* ***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.

**Code:**

*PL/SQL Procedure-*

CREATE OR REPLACE PROCEDURE UpdateEmployeeBonus (

dept\_name IN VARCHAR2,

bonus\_percent IN NUMBER

) IS

BEGIN

UPDATE Employees

SET Salary = Salary + (Salary \* bonus\_percent / 100)

WHERE Department = dept\_name;

DBMS\_OUTPUT.PUT\_LINE('Bonus added to ' || dept\_name || ' Employees!');

END;

/

*Execution Block-*

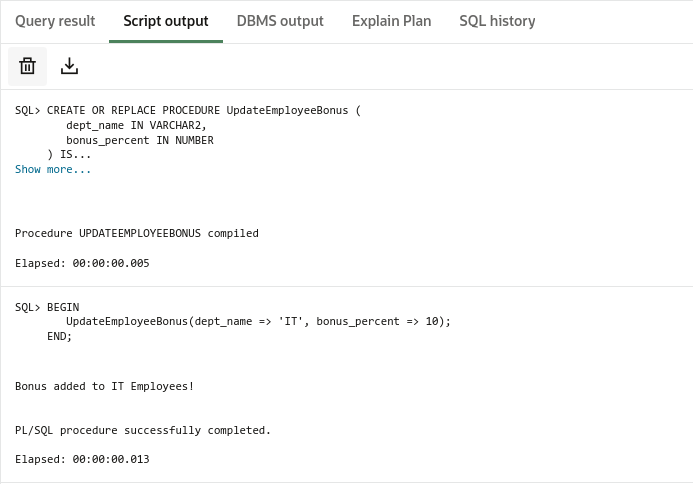
BEGIN

UpdateEmployeeBonus(dept\_name => 'IT', bonus\_percent => 10);

END;

/

**Output:**

****

* ***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.

**Code:**

*PL/SQL Procedure-*

CREATE OR REPLACE PROCEDURE TransferFunds (

src\_account IN NUMBER,

dest\_account IN NUMBER,

funds IN NUMBER

) IS

src\_balance NUMBER;

BEGIN

-- Get the current balance of the source account

SELECT Balance INTO src\_balance

FROM Accounts

WHERE AccountID = src\_account;

-- Check if sufficient funds are available

IF src\_balance >= funds THEN

-- Deduct from source account

UPDATE Accounts

SET Balance = Balance - funds

WHERE AccountID = src\_account;

-- Add to destination account

UPDATE Accounts

SET Balance = Balance + funds

WHERE AccountID = dest\_account;

-- Confirmation message

DBMS\_OUTPUT.PUT\_LINE('Transfer of ' || funds ||

' from account ' || src\_account ||

' to account ' || dest\_account || ' completed.');

ELSE

DBMS\_OUTPUT.PUT\_LINE('Insufficient balance in source account.');

END IF;

END;

/

*Execution Block-*

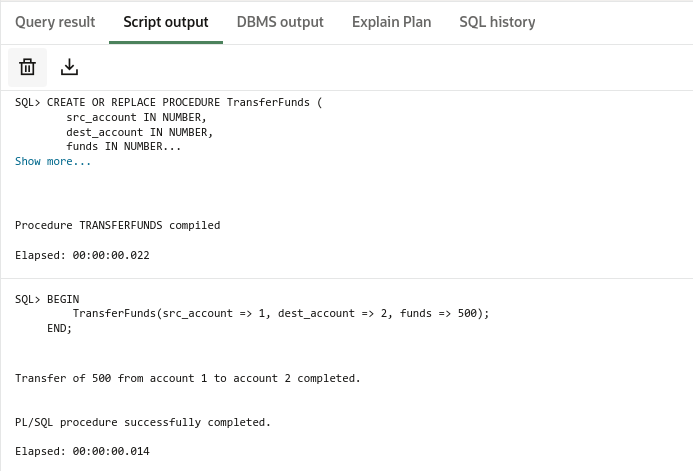
BEGIN

TransferFunds(src\_account => 1, dest\_account => 2, funds => 500);

END;

/

**Output:**

****