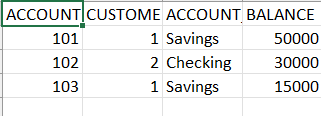
**Exercise 3: Stored Procedures**

**Database Setup**

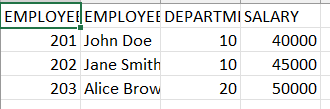
To implement the following scenarios, I first created the required tables accounts and employees using PL/SQL and inserted sample data. These tables include fields such as account\_id, account\_type, balance, employee\_id, department\_id, and salary, which are used in the PL/SQL procedures below.

**Attached Tables:**

* **accounts\_table\_output.csv**

****

* **employees\_table\_output.csv**

****

**Scenario 1: Monthly Interest for 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.

CREATE OR REPLACE PROCEDURE ProcessMonthlyInterest IS

BEGIN

UPDATE accounts

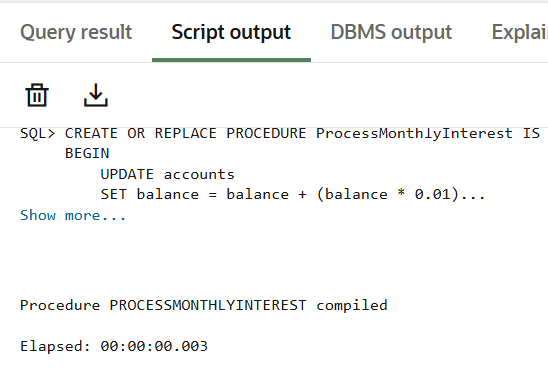
SET balance = balance + (balance \* 0.01)

WHERE account\_type = 'Savings';

DBMS\_OUTPUT.PUT\_LINE('Monthly interest applied to all savings accounts.');

END;

/



**Scenario 2: Employee Bonus Based on 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.

CREATE OR REPLACE PROCEDURE UpdateEmployeeBonus (

p\_department\_id IN employees.department\_id%TYPE,

p\_bonus\_percent IN NUMBER

) IS

BEGIN

UPDATE employees

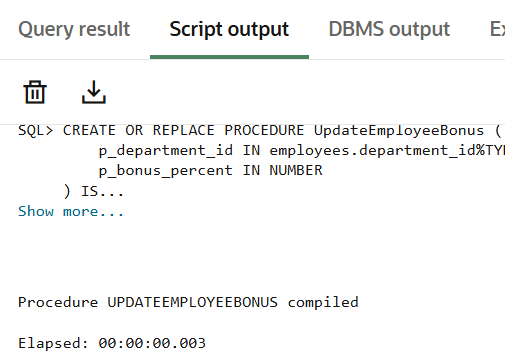
SET salary = salary + (salary \* (p\_bonus\_percent / 100))

WHERE department\_id = p\_department\_id;

DBMS\_OUTPUT.PUT\_LINE('Bonus applied to department ID: ' || p\_department\_id);

END;

/



**Scenario 3: Fund Transfer Between 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.

CREATE OR REPLACE PROCEDURE TransferFunds (

p\_from\_account IN accounts.account\_id%TYPE,

p\_to\_account IN accounts.account\_id%TYPE,

p\_amount IN NUMBER

) IS

v\_balance NUMBER;

BEGIN

SELECT balance INTO v\_balance

FROM accounts

WHERE account\_id = p\_from\_account;

IF v\_balance < p\_amount THEN

RAISE\_APPLICATION\_ERROR(-20001, 'Insufficient balance in source account.');

ELSE

UPDATE accounts

SET balance = balance - p\_amount

WHERE account\_id = p\_from\_account;

UPDATE accounts

SET balance = balance + p\_amount

WHERE account\_id = p\_to\_account;

DBMS\_OUTPUT.PUT\_LINE('₹' || p\_amount || ' transferred from Account ' || p\_from\_account || ' to Account ' || p\_to\_account);

END IF;

END;

/

