**Exercise 3: Stored Procedures**

Step 1: Create accounts table

CREATE TABLE accounts (

account\_id NUMBER PRIMARY KEY,

customer\_id NUMBER,

account\_type VARCHAR2(20),

balance NUMBER

);

Insert sample values

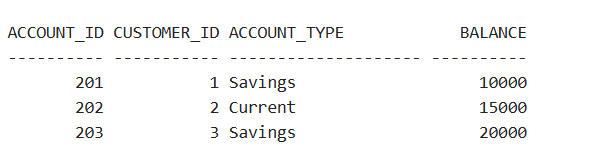
INSERT INTO accounts VALUES (201, 1, 'Savings', 10000);

INSERT INTO accounts VALUES (202, 2, 'Current', 15000);

INSERT INTO accounts VALUES (203, 3, 'Savings', 20000);

COMMIT;

* SELECT \* FROM ACCOUNTS;



Step 2: Create employees table

CREATE TABLE employees (

emp\_id NUMBER PRIMARY KEY,

emp\_name VARCHAR2(50),

department VARCHAR2(50),

salary NUMBER

);

Insert sample values

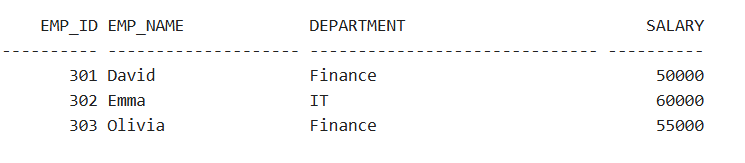
INSERT INTO employees VALUES (301, 'David', 'Finance', 50000);

INSERT INTO employees VALUES (302, 'Emma', 'IT', 60000);

INSERT INTO employees VALUES (303, 'Olivia', 'Finance', 55000);

COMMIT;

* SELECT \* FROM EMPLOYEES;



**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.

CREATE OR REPLACE PROCEDURE ProcessMonthlyInterest AS

BEGIN

UPDATE accounts

SET balance = balance + (balance \* 0.01)

WHERE account\_type = 'Savings';

COMMIT;

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

END;

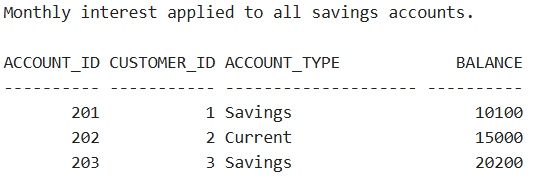
/

SET SERVEROUTPUT ON;

EXEC ProcessMonthlyInterest;

SELECT \* FROM accounts;

**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.

CREATE OR REPLACE PROCEDURE UpdateEmployeeBonus (dept\_name IN VARCHAR2,

bonus\_pct IN NUMBER) AS

BEGIN

UPDATE employees

SET salary = salary + (salary \* bonus\_pct / 100)

WHERE department = dept\_name

COMMIT;

DBMS\_OUTPUT.PUT\_LINE('Bonus applied to ' || dept\_name || ' department.');

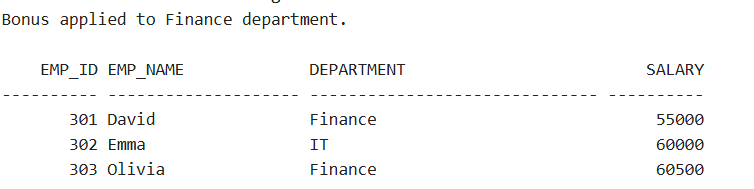
END;

/

EXEC UpdateEmployeeBonus('Finance', 10);

SELECT \* FROM employees;

**OUPUT:**

****

**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.

CREATE OR REPLACE PROCEDURE TransferFunds (

from\_acc IN NUMBER,

to\_acc IN NUMBER,

amount IN NUMBER

) AS

from\_balance NUMBER;

BEGIN

-- Get source account balance

SELECT balance INTO from\_balance

FROM accounts

WHERE account\_id = from\_acc;

-- Check sufficient balance

IF from\_balance >= amount THEN

-- Deduct from source

UPDATE accounts

SET balance = balance - amount

WHERE account\_id = from\_acc;

UPDATE accounts

SET balance = balance + amount

WHERE account\_id = to\_acc;

COMMIT;

DBMS\_OUTPUT.PUT\_LINE('Transfer successful: ' || amount || ' transferred from Account ' || from\_acc || ' to Account ' || to\_acc);

ELSE

DBMS\_OUTPUT.PUT\_LINE('Insufficient balance in Account ' || from\_acc);

END IF;

END;

/

EXEC TransferFunds(201, 203, 2000);

SELECT \* FROM accounts;

**OUTPUT:**

