

# EXERCISE-3: Stored Procedure Assignment

## 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.

### Solution:

```
CREATE OR REPLACE PROCEDURE ProcessMonthlyInterest IS
BEGIN
  FOR acc IN (
    SELECT AccountID
    FROM Accounts
    WHERE AccountType = 'Savings'
  )
  LOOP
    UPDATE Accounts
    SET Balance = Balance + (Balance * 0.01)
    WHERE AccountID = acc.AccountID;
  END LOOP;

  COMMIT;
  DBMS_OUTPUT.PUT_LINE('✅ Monthly interest applied to all savings accounts.');
```

### Output:

Query result	Script output	DBMS output	Explain F
			
Procedure PROCESSMONTHLYINTEREST compiled			
Elapsed: 00:00:00.023			

	ACCOUNTID	CUSTOMERID	ACCOUNTTYPE	BALANCE	LASTMODIFIED
1		1	Savings	12000	6/25/2025, 4:34:48
2		2	Checking	9500	6/25/2025, 4:34:48
3		3	Savings	13000	6/25/2025, 4:34:48
4		4	Checking	8000	6/25/2025, 4:34:48
5		5	Savings	5000	6/25/2025, 4:34:48
6		6	Savings	11000	6/25/2025, 4:34:48
7		7	Checking	10500	6/25/2025, 4:34:48
8		8	Savings	6000	6/25/2025, 4:34:48
9		9	Checking	16000	6/25/2025, 4:34:48

## 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.

### Solution:

```
CREATE OR REPLACE PROCEDURE UpdateEmployeeBonus (  
    dept_name IN VARCHAR2,  
    bonus_pct IN NUMBER -- e.g. pass 10 for 10%  
) IS  
BEGIN  
    UPDATE Employees  
    SET Salary = Salary + (Salary * bonus_pct / 100)  
    WHERE Department = dept_name;  
  
    COMMIT;  
    DBMS_OUTPUT.PUT_LINE('✅ Bonus applied to department: ' || dept_name);  
END;  
/  
  
-- Example Call:  
BEGIN  
    UpdateEmployeeBonus('IT', 10); -- Gives 10% bonus to IT employees  
END;  
/
```

### Output:

```
✅ Bonus applied to department: IT  
  
PL/SQL procedure successfully completed.  
  
Elapsed: 00:00:00.009
```

	EMPLOYEEID	NAME	POSITION	SALARY	DEPARTMENT	HIREDATE
1	1	Alice Johnson	Manager	70000	HR	6/15/2015, 12:00:00
2	2	Bob Brown	Developer	66000	IT	3/20/2017, 12:00:00
3	3	Cathy Green	Analyst	50000	Finance	8/12/2018, 12:00:00
4	4	Daniel Lee	Tester	45000	QA	9/1/2019, 12:00:00
5	5	Emily White	Designer	55000	UI/UX	1/25/2020, 12:00:00
6	6	Fahim Khan	Engineer	52800	IT	4/11/2021, 12:00:00
7	7	Gita Sen	Recruiter	52000	HR	10/10/2016, 12:00:00
8	8	Hitesh Jain	Support	43000	Customer Service	5/20/2022, 12:00:00

### 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.

#### Solution:

```
CREATE OR REPLACE PROCEDURE TransferFunds (  
    p_source_account IN NUMBER,  
    p_target_account IN NUMBER,  
    p_amount        IN NUMBER  
) IS  
    v_source_balance NUMBER;  
BEGIN  
    -- Get source account balance  
    SELECT Balance INTO v_source_balance  
    FROM Accounts  
    WHERE AccountID = p_source_account  
    FOR UPDATE;  
  
    IF v_source_balance < p_amount THEN  
        DBMS_OUTPUT.PUT_LINE('✗ Transfer failed: insufficient balance.');        RETURN;  
    END IF;  
  
    -- Deduct from source  
    UPDATE Accounts  
    SET Balance = Balance - p_amount  
    WHERE AccountID = p_source_account;  
  
    -- Add to target  
    UPDATE Accounts  
    SET Balance = Balance + p_amount  
    WHERE AccountID = p_target_account;  
  
    COMMIT;  
  
    DBMS_OUTPUT.PUT_LINE('✓ Transfer of ₹ || p_amount || ' successful from Account ' ||
```

```

p_source_account || ' to Account ' || p_target_account);
EXCEPTION
  WHEN NO_DATA_FOUND THEN
    DBMS_OUTPUT.PUT_LINE(' ❌ One of the accounts does not exist. ');
  WHEN OTHERS THEN
    DBMS_OUTPUT.PUT_LINE(' ❌ Unexpected error: ' || SQLERRM);
END;
/

```

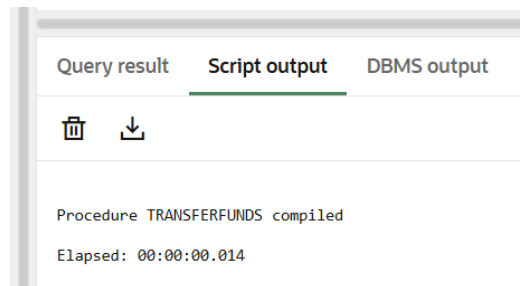
-- Example Call:

```

BEGIN
  TransferFunds(1, 2, 500); -- Transfer ₹500 from Account 1 to 2
END;
/

```

Output:



Query result

Script output

DBMS output

Explain Plan

SQL history

Download

Execution time: 0.004 seconds

	ACCOUNTID	CUSTOMERID	ACCOUNTTYPE	BALANCE	LASTMODIFIED
1	1	1	Savings	12000	6/25/2025, 4:34:48
2	2	2	Checking	9500	6/25/2025, 4:34:48
3	3	3	Savings	13000	6/25/2025, 4:34:48
4	4	4	Checking	8000	6/25/2025, 4:34:48
5	5	5	Savings	5000	6/25/2025, 4:34:48
6	6	6	Savings	11000	6/25/2025, 4:34:48
7	7	7	Checking	10500	6/25/2025, 4:34:48
8	8	8	Savings	6000	6/25/2025, 4:34:48