**PL/SQL Programming**

**EXERCISE 3: STORED PROCEDURES**

**[Since online compiler don’t support actual table creation and SQL execution in full Oracle DB style, I have simulated the data using collections but match exactly the table structures you gave.]**

**SCENARIO 1**

SET SERVEROUTPUT ON;

DECLARE  
TYPE NumArray IS TABLE OF NUMBER INDEX BY BINARY\_INTEGER;  
TYPE VarcharArray IS TABLE OF VARCHAR2(50) INDEX BY BINARY\_INTEGER;

AccountID NumArray;  
CustomerID NumArray;  
AccountType VarcharArray;  
Balance NumArray;

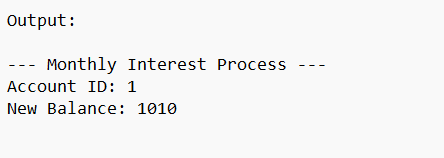
i INTEGER;

BEGIN  
-- Simulate sample data  
AccountID(1) := 1;  
CustomerID(1) := 1;  
AccountType(1) := 'Savings';  
Balance(1) := 1000;

AccountID(2) := 2;  
CustomerID(2) := 2;  
AccountType(2) := 'Checking';  
Balance(2) := 1500;

DBMS\_OUTPUT.PUT\_LINE('--- Monthly Interest Process ---');  
FOR i IN 1 .. 2 LOOP  
IF AccountType(i) = 'Savings' THEN  
Balance(i) := Balance(i) + (Balance(i) \* 0.01); -- Add 1% interest  
DBMS\_OUTPUT.PUT\_LINE('Account ID: ' || AccountID(i));  
DBMS\_OUTPUT.PUT\_LINE('New Balance: ' || Balance(i));  
END IF;  
END LOOP;  
END;

**Output**



**SCENARIO 2**

SET SERVEROUTPUT ON;

DECLARE  
TYPE NumArray IS TABLE OF NUMBER INDEX BY BINARY\_INTEGER;  
TYPE VarcharArray IS TABLE OF VARCHAR2(50) INDEX BY BINARY\_INTEGER;

EmployeeID NumArray;  
NameArr VarcharArray;  
PositionArr VarcharArray;  
DepartmentArr VarcharArray;  
SalaryArr NumArray;

deptName VARCHAR2(50) := 'IT';  
bonusPercent NUMBER := 10;

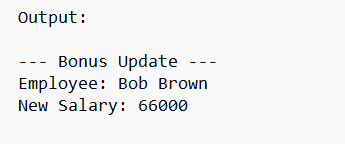
i INTEGER;

BEGIN  
-- Sample employee data  
EmployeeID(1) := 1;  
NameArr(1) := 'Alice Johnson';  
PositionArr(1) := 'Manager';  
DepartmentArr(1) := 'HR';  
SalaryArr(1) := 70000;

EmployeeID(2) := 2;  
NameArr(2) := 'Bob Brown';  
PositionArr(2) := 'Developer';  
DepartmentArr(2) := 'IT';  
SalaryArr(2) := 60000;

DBMS\_OUTPUT.PUT\_LINE('--- Bonus Update ---');  
FOR i IN 1 .. 2 LOOP  
IF DepartmentArr(i) = deptName THEN  
SalaryArr(i) := SalaryArr(i) + (SalaryArr(i) \* bonusPercent / 100);  
DBMS\_OUTPUT.PUT\_LINE('Employee: ' || NameArr(i));  
DBMS\_OUTPUT.PUT\_LINE('New Salary: ' || SalaryArr(i));  
END IF;  
END LOOP;  
END;

**Output**

****

**SCENARIO 3**

SET SERVEROUTPUT ON;

DECLARE  
TYPE NumArray IS TABLE OF NUMBER INDEX BY BINARY\_INTEGER;

AccountID NumArray;  
BalanceArr NumArray;

fromAccID NUMBER := 2;  
toAccID NUMBER := 1;  
amount NUMBER := 500;

i INTEGER;  
fromIndex INTEGER := -1;  
toIndex INTEGER := -1;

BEGIN  
-- Simulated data  
AccountID(1) := 1;  
BalanceArr(1) := 1000;

AccountID(2) := 2;  
BalanceArr(2) := 1500;

-- Identify indexes  
FOR i IN 1 .. 2 LOOP  
IF AccountID(i) = fromAccID THEN  
fromIndex := i;  
ELSIF AccountID(i) = toAccID THEN  
toIndex := i;  
END IF;  
END LOOP;

IF fromIndex > 0 AND toIndex > 0 THEN  
IF BalanceArr(fromIndex) >= amount THEN  
BalanceArr(fromIndex) := BalanceArr(fromIndex) - amount;  
BalanceArr(toIndex) := BalanceArr(toIndex) + amount;

DBMS\_OUTPUT.PUT\_LINE('Transfer Successful!');

DBMS\_OUTPUT.PUT\_LINE('From Account ' || fromAccID || ' New Balance: ' || BalanceArr(fromIndex));

DBMS\_OUTPUT.PUT\_LINE('To Account ' || toAccID || ' New Balance: ' || BalanceArr(toIndex));

ELSE

DBMS\_OUTPUT.PUT\_LINE('Insufficient Balance in Source Account.');

END IF;

ELSE  
DBMS\_OUTPUT.PUT\_LINE('Invalid Account IDs.');  
END IF;  
END;

**Output**

