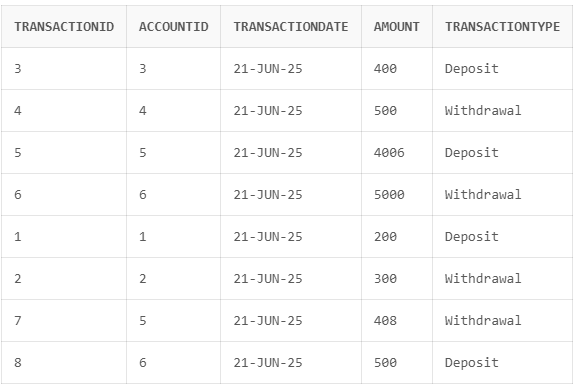
**Exercise 3: Stored Procedures**

**Initial tables :**

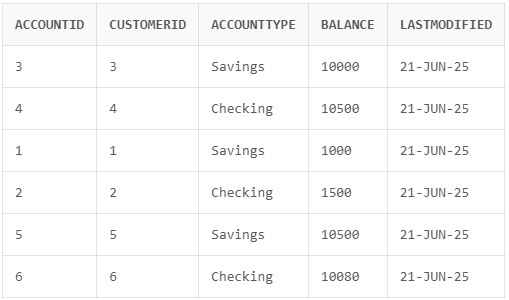
**Employees :**

****

**Transactions:**

****

**Accounts :**

****

**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 :**

CREATE OR REPLACE PROCEDURE ProcessMonthlyInterest AS

BEGIN

FOR rec IN (

SELECT AccountID, Balance

FROM Accounts

WHERE AccountType = 'Savings'

) LOOP

UPDATE Accounts

SET Balance = Balance + (rec.Balance \* 0.01),

LastModified = SYSDATE

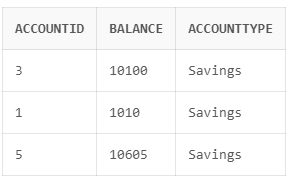
WHERE AccountID = rec.AccountID;

END LOOP;

COMMIT;

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 :**

CREATE OR REPLACE PROCEDURE UpdateEmployeeBonus (

dept\_name IN VARCHAR2,

bonus\_percent IN NUMBER -- e.g., pass 10 for 10%

) AS



BEGIN

UPDATE Employees

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

WHERE Department = dept\_name;

COMMIT;

END;

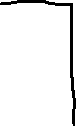
**Output :**

BEGIN

UpdateEmployeeBonus('IT', 10);

END;





**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 :**

CREATE OR REPLACE PROCEDURE TransferFunds (

from\_account IN NUMBER,

to\_account IN NUMBER,

amount IN NUMBER

) AS

insufficient\_balance EXCEPTION;

curr\_balance NUMBER;

BEGIN

SELECT Balance INTO curr\_balance

FROM Accounts

WHERE AccountID = from\_account

FOR UPDATE;

IF curr\_balance < amount THEN

RAISE insufficient\_balance;

END IF;

UPDATE Accounts

SET Balance = Balance - amount,

LastModified = SYSDATE

WHERE AccountID = from\_account;

UPDATE Accounts

SET Balance = Balance + amount,

LastModified = SYSDATE

WHERE AccountID = to\_account;

INSERT INTO Transactions

VALUES (9, from\_account, SYSDATE, amount, 'Withdrawal');

INSERT INTO Transactions

VALUES (10, to\_account, SYSDATE, amount, 'Deposit');

COMMIT;

EXCEPTION

WHEN insufficient\_balance THEN

ROLLBACK;

DBMS\_OUTPUT.PUT\_LINE('Transfer failed: Insufficient funds.');

WHEN NO\_DATA\_FOUND THEN

ROLLBACK;

DBMS\_OUTPUT.PUT\_LINE('Transfer failed: Account not found.');

END;

**Output :**

BEGIN

TransferFunds(1, 2, 500);

END;

