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(

p\_from\_accountID IN NUMBER,

p\_to\_accountID IN NUMBER,

p\_amount IN NUMBER

) AS

transaction\_no NUMBER;

new\_transaction\_no NUMBER;

rec1 Accounts%ROWTYPE;

BEGIN

SELECT \* INTO rec1 FROM Accounts WHERE AccountID=p\_from\_accountID;

IF rec1.Balance>p\_amount

THEN

UPDATE Accounts

SET Balance=Balance-p\_amount

WHERE AccountID=p\_from\_accountID;

SELECT MAX(TransactionID) INTO transaction\_no FROM Transactions;

INSERT INTO Transactions VALUES(transaction\_no+1,p\_from\_accountID,SYSDATE,p\_amount,'Withdrawal');

UPDATE Accounts

SET Balance=Balance+p\_amount

WHERE AccountID=p\_to\_accountID;

SELECT MAX(TransactionID) INTO new\_transaction\_no FROM Transactions;

INSERT INTO Transactions VALUES(new\_transaction\_no+1,p\_to\_accountID,SYSDATE,p\_amount,'Deposit');

ELSE

DBMS\_OUTPUT.PUT\_LINE('Insufficient funds in account '||p\_from\_accountID);

END IF;

COMMIT;

END;

/

BEGIN

TransferFunds(&account\_no\_1,&account\_no\_2,&amount);

END;

/

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Before executing the procedure

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

SELECT \* FROM Accounts;

ACCOUNTID CUSTOMERID ACCOUNTTYPE BALANCE LASTMODIF

---------- ---------- -------------------- ---------- ---------

1 1 Savings 1010 19-JUN-25

2 2 Current 1500 19-JUN-25

3 3 Savings 1717 19-JUN-25

4 4 Savings 2020 19-JUN-25

5 5 Current 1300 19-JUN-25

6 6 Savings 10100 19-JUN-25

6 rows selected.

SQL> SELECT \* FROM Transactions;

TRANSACTIONID ACCOUNTID TRANSACTI AMOUNT TRANSACTIO

------------- ---------- --------- ---------- ----------

1 1 19-JUN-25 200 Deposit

2 2 19-JUN-25 300 Withdrawal

3 3 19-JUN-25 500 Deposit

4 4 19-JUN-25 200 Deposit

5 5 19-JUN-25 400 Withdrawal

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

After executing the procedure

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

SQL> ACCEPT account\_no\_1 NUMBER PROMPT 'Enter the account ID from which the money has to be transferred : '

Enter the account ID from which the money has to be transferred : 6

SQL> ACCEPT account\_no\_2 NUMBER PROMPT 'Enter the account ID to which the mo

ney has to be transferred : '

Enter the account ID to which the money has to be transferred : 2

SQL> ACCEPT amount NUMBER PROMPT 'Enter the amount : '

Enter the amount : 500

SQL> BEGIN

2 TransferFunds(&account\_no\_1,&account\_no\_2,&amount);

3 END;

4 /

old 2: TransferFunds(&account\_no\_1,&account\_no\_2,&amount);

new 2: TransferFunds( 6, 2, 500);

PL/SQL procedure successfully completed.

SQL> SELECT \* FROM Accounts;

ACCOUNTID CUSTOMERID ACCOUNTTYPE BALANCE LASTMODIF

---------- ---------- -------------------- ---------- ---------

1 1 Savings 1010 19-JUN-25

2 2 Current 2000 19-JUN-25

3 3 Savings 1717 19-JUN-25

4 4 Savings 2020 19-JUN-25

5 5 Current 1300 19-JUN-25

6 6 Savings 9600 19-JUN-25

6 rows selected.

SQL> SELECT \* FROM Transactions;

TRANSACTIONID ACCOUNTID TRANSACTI AMOUNT TRANSACTIO

------------- ---------- --------- ---------- ----------

1 1 19-JUN-25 200 Deposit

2 2 19-JUN-25 300 Withdrawal

3 3 19-JUN-25 500 Deposit

4 4 19-JUN-25 200 Deposit

5 5 19-JUN-25 400 Withdrawal

6 6 20-JUN-25 500 Withdrawal

7 2 20-JUN-25 500 Deposit

7 rows selected.