Write and execute PL/SQL procedure to transfer fund from one account to another.

Aishi De

```
SQL> CREATE TABLE BankAccounts (
  2      AccountNumber NUMBER PRIMARY KEY,
  3      AccountHolder VARCHAR2(100),
  4      Balance NUMBER
  5  );

Table created.

SQL>
SQL> INSERT INTO BankAccounts (AccountNumber, AccountHolder, Balance)
  2  VALUES (101, 'John Doe', 5000);

1 row created.

SQL> INSERT INTO BankAccounts (AccountNumber, AccountHolder, Balance)
  2  VALUES (102, 'Jane Smith', 3000);

1 row created.

SQL> COMMIT;

Commit complete.

SQL> SET SERVEROUTPUT ON;
SQL>
SQL> CREATE OR REPLACE PROCEDURE TransferFunds(
  2      source_account IN NUMBER,
  3      dest_account IN NUMBER,
  4      transfer_amount IN NUMBER
  5  ) IS
  6      source_balance NUMBER;
  7      dest_balance NUMBER;
  8  BEGIN
  9      BEGIN
 10          SELECT Balance INTO source_balance
 11          FROM BankAccounts
 12          WHERE AccountNumber = source_account;
 13      EXCEPTION
 14          WHEN NO_DATA_FOUND THEN
 15              DBMS_OUTPUT.PUT_LINE('Error: Source account does not exist.');
 16              RETURN;
 17      END;
 18
 19      BEGIN
 20          SELECT Balance INTO dest_balance
 21          FROM BankAccounts
 22          WHERE AccountNumber = dest_account;
 23      EXCEPTION
 24          WHEN NO_DATA_FOUND THEN
```

```
24         WHEN NO_DATA_FOUND THEN
25             DBMS_OUTPUT.PUT_LINE('Error: Destination account does not exist.');
26             RETURN;
27     END;
28
29     IF source_balance < transfer_amount THEN
30         DBMS_OUTPUT.PUT_LINE('Error: Insufficient funds in the source account.');
31         RETURN;
32     END IF;
33
34     UPDATE BankAccounts
35     SET Balance = Balance - transfer_amount
36     WHERE AccountNumber = source_account;
37
38     UPDATE BankAccounts
39     SET Balance = Balance + transfer_amount
40     WHERE AccountNumber = dest_account;
41
42     COMMIT;
43
44     DBMS_OUTPUT.PUT_LINE('Transfer successful: ' || transfer_amount || ' has been transferred f
rom account ' || source_account || ' to account ' || dest_account);
45  EXCEPTION
46     WHEN OTHERS THEN
47         DBMS_OUTPUT.PUT_LINE('Error: An unexpected error occurred.');
48         ROLLBACK;
49  END;
50  /

Procedure created.

SQL> BEGIN
  2       TransferFunds(101, 102, 1000); -- Transfer 1000 from account 101 to 102
  3  END;
  4  /
Transfer successful: 1000 has been transferred from account 101 to account 102

PL/SQL procedure successfully completed.
```