## **Exercise 2: Error Handling**

## **Scenario 1: Fund Transfer Between Accounts**

### **Objective**

Create a **stored procedure SafeTransferFunds** to:

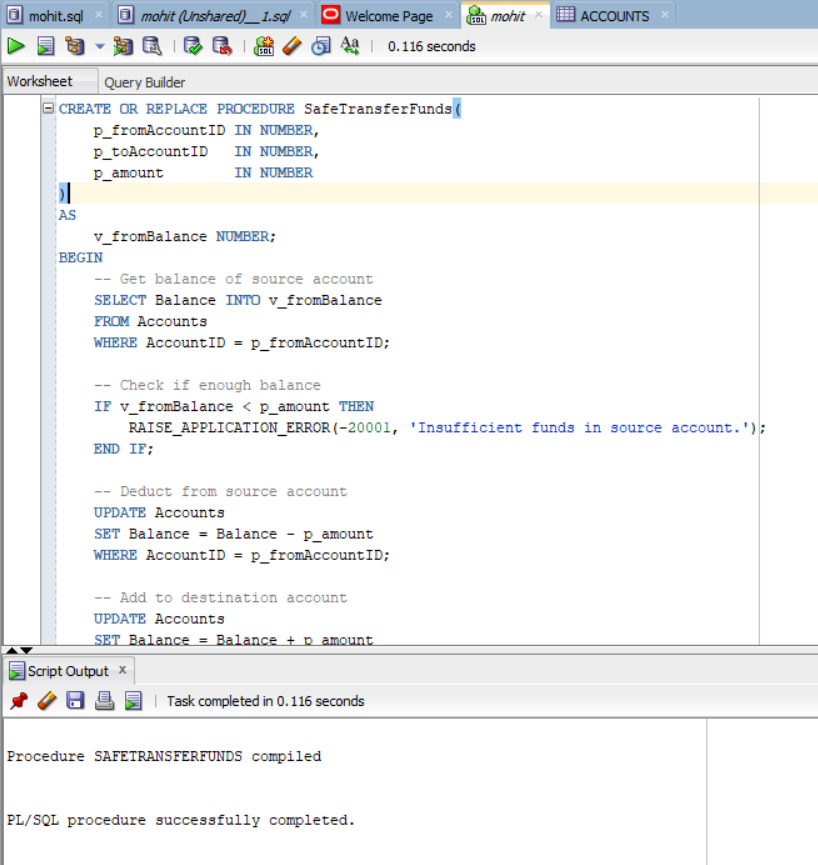
* Transfer a specified amount from one account to another.
* If an error occurs (like insufficient funds), handle it gracefully:
  + **Log an error message via DBMS\_OUTPUT.PUT\_LINE**
  + **Rollback the transaction** to maintain data integrity.

## **Explanation**

* Use **EXCEPTION handling** in PL/SQL.
* Check if the source account has enough balance before proceeding.
* If not, raise an application error.
* Use **SAVEPOINT** and **ROLLBACK** to manage transactions safely.

## **Stored Procedure Code**

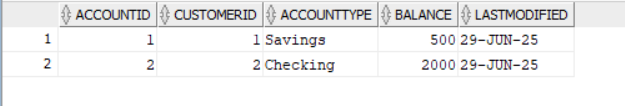
CREATE OR REPLACE PROCEDURE SafeTransferFunds(  
 p\_fromAccountID IN NUMBER,  
 p\_toAccountID IN NUMBER,  
 p\_amount IN NUMBER  
)  
AS  
 v\_fromBalance NUMBER;



### **How to Call the Procedure**

EXEC SafeTransferFunds(1, 2, 500);

### Output:



## **Expected Behavior**

|  |  |
| --- | --- |
| **Case** | **Result** |
| Sufficient balance | Transfers amount and commits transaction |
| Insufficient balance / error | Rolls back and logs error message |

## **Example Output (in DBMS Output):**

* **If successful**:

Funds transferred successfully.

* **If insufficient funds**:

Error occurred: ORA-20001: Insufficient funds in source account.