**Exercise 2: Error Handling**

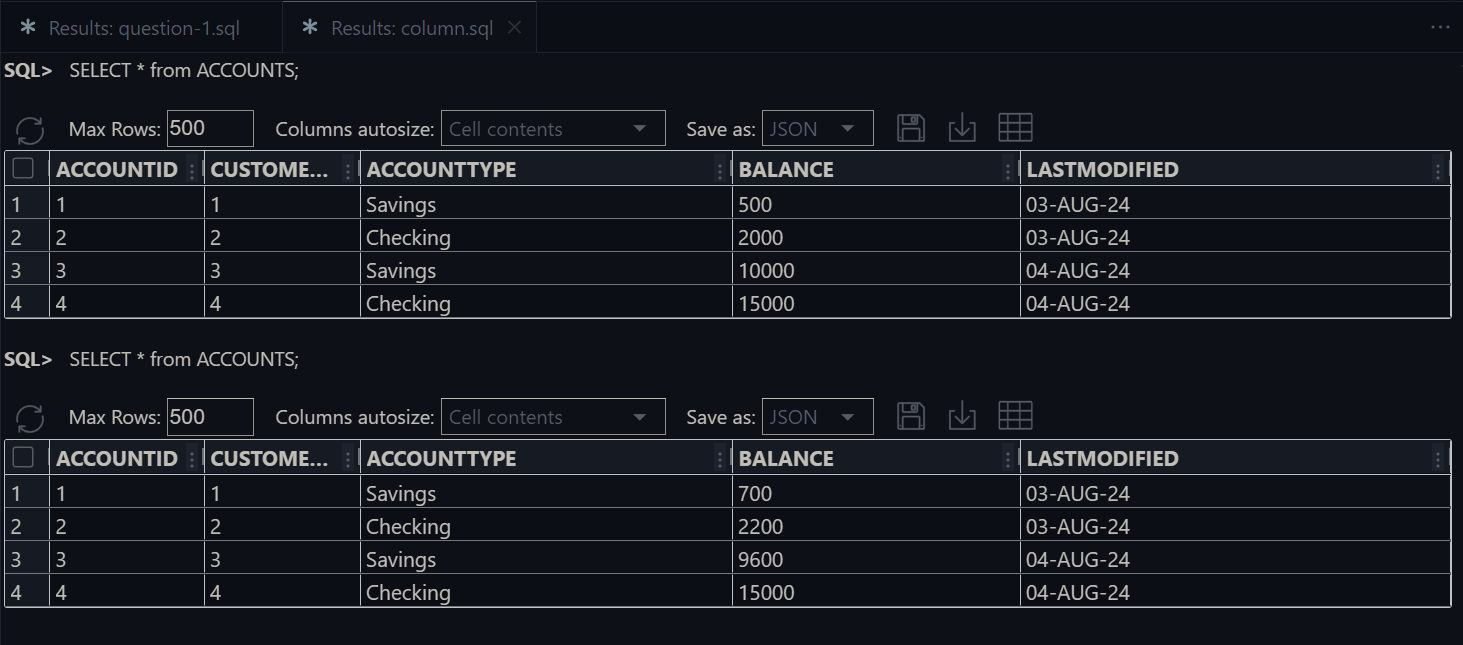
· Scenario 1: Handle exceptions during fund transfers between accounts.

Question: Write a stored procedure SafeTransferFunds that transfers funds between two accounts. Ensure that if any error occurs (e.g., insufficient funds), an appropriate error message is logged and the transaction is rolled back.

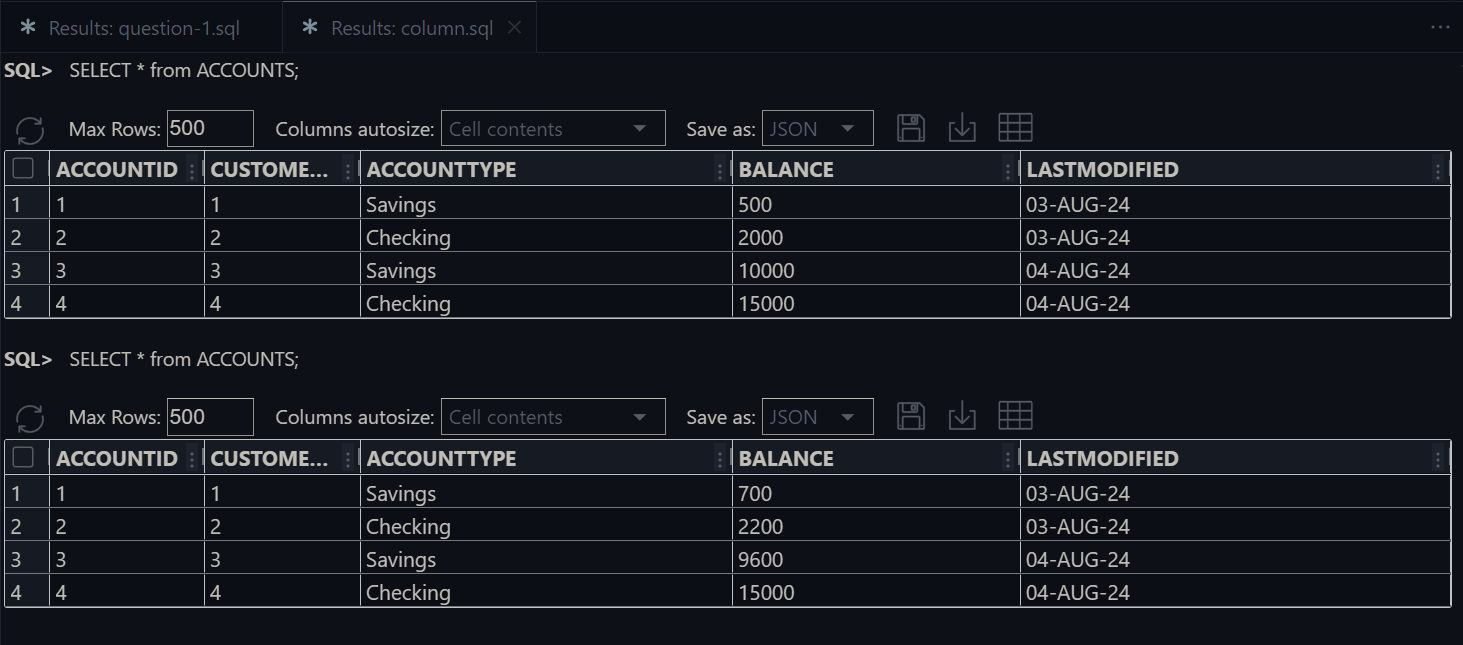
* Procedure Declaration: Defines SafeTransferFunds with parameters for source account ID, target account ID, and transfer amount. Retrieve and locks source account balance, deducts the amount and add it to target account, handling the exception. Lastly, test the procedure with valid and invalid input to verify it.

**The Output :**

1. Before Transaction-



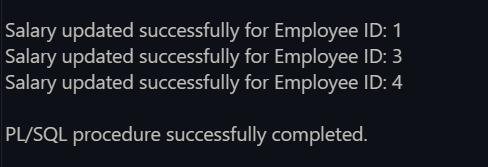
1. After transaction –



* **Scenario 2:** Manage errors when updating employee salaries.
* **Question:** Write a stored procedure **UpdateSalary** that increases the salary of an employee by a given percentage. If the employee ID does not exist, handle the exception and log an error message.

This procedure attempts to collect the current salary of the employee using their ID. If the employee exists, the procedure update salary by adding percentage.It handles exception and commit transaction if successfull.

**THE OUTPUT :**

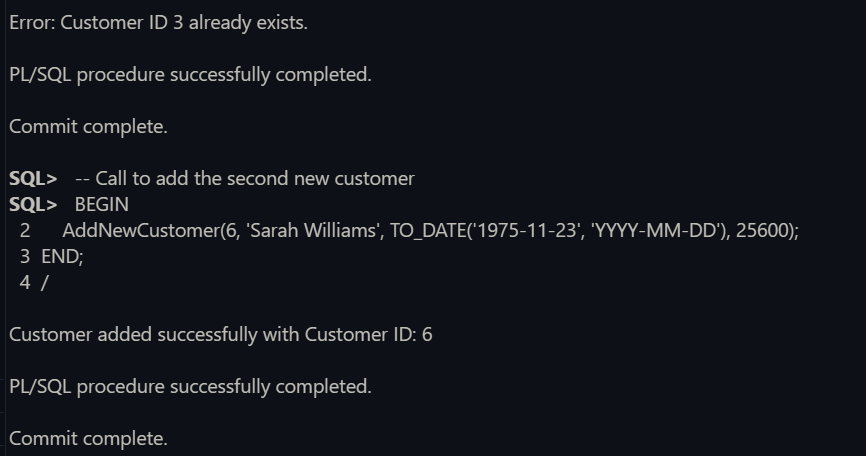
****

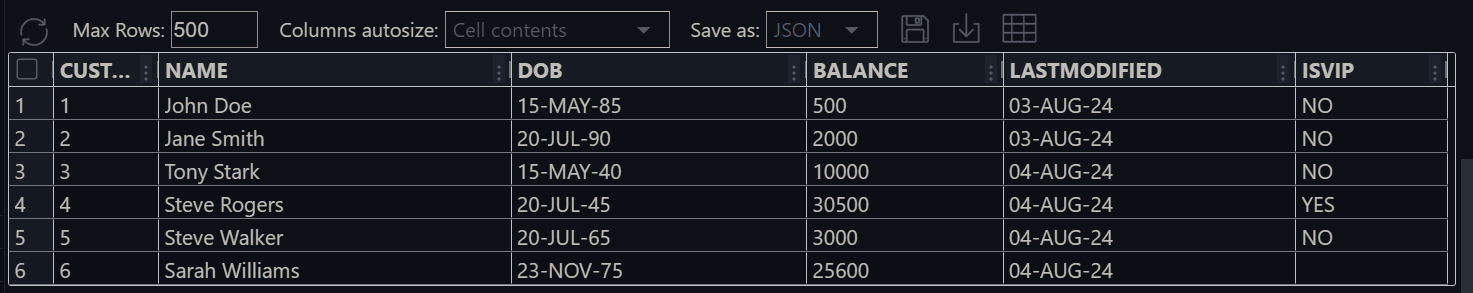
Before and After Salary Update respectively–



* **Scenario 3:** Ensure data integrity when adding a new customer.
* **Question:** Write a stored procedure **AddNewCustomer** that inserts a new customer into the Customers table. If a customer with the same ID already exists, handle the exception by logging an error and preventing the insertion.

**The OUTPUT** ---



Here add a Customer in ID 3 and ID 6. The ID 3 already have a customer detail so it does not change the ID 3 and add new Details in ID 6 .