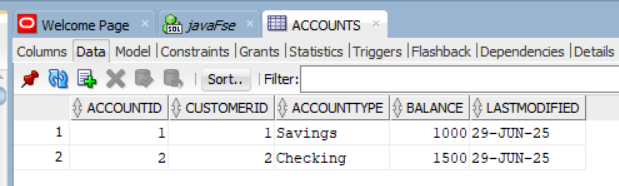
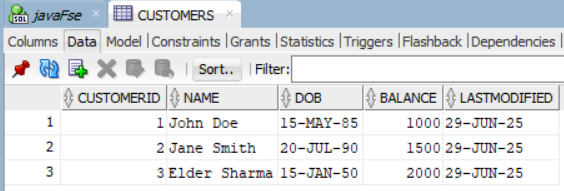
**Exercise 1: Control Structures**

**Tables Created:**

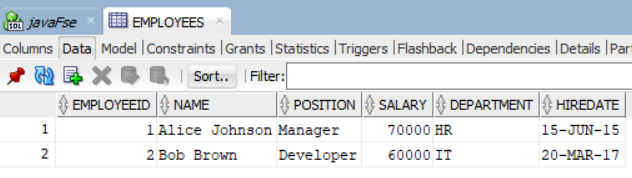
* **ACCOUNTS –**

****

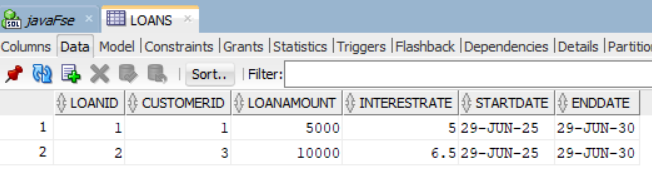
* **CUSTOMERS –**

****

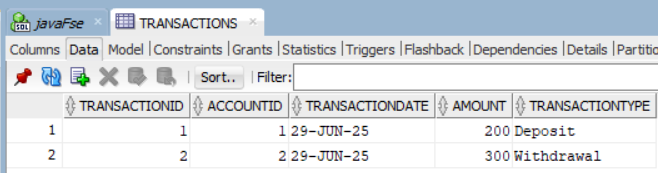
* **EMPLOYEES –**

****

* **LOANS –**

****

* **TRANSACTIONS –**

****

**Scenario 1:** The bank wants to apply a discount to loan interest rates for customers above 60 years old.

**Question:** Write a PL/SQL block that loops through all customers, checks their age, and if they are above 60, apply a 1% discount to their current loan interest rates.

**PL/SQL BLOCK :**

BEGIN

FOR res IN (

SELECT c.Name, c.CustomerID, l.LoanID, l.InterestRate

FROM Loans l

JOIN Customers c ON l.CustomerID = c.CustomerID

WHERE TRUNC(MONTHS\_BETWEEN(SYSDATE, c.DOB) / 12) > 60

) LOOP

UPDATE Loans

SET InterestRate = res.InterestRate - 1

WHERE LoanID = res.LoanID;

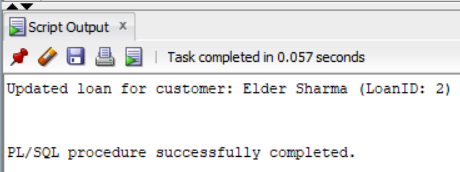
DBMS\_OUTPUT.PUT\_LINE('Updated loan for customer: ' || res.Name || ' (LoanID: ' || res.LoanID || ')');

END LOOP;

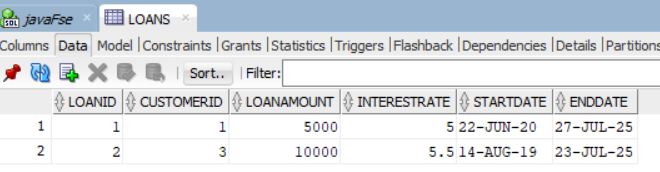
COMMIT;

END;

**OUTPUT:**

****

**TABLE AFTER RUNNING THE BLOCK:**

****

**Scenario 2:** A customer can be promoted to VIP status based on their balance.

**Question:** Write a PL/SQL block that iterates through all customers and sets a flag IsVIP to TRUE for those with a balance over $10,000.

**PL/SQL BLOCK:**

ALTER TABLE Customers ADD IsVIP VARCHAR2(5); //Alter Table to Add IsVIP Column

BEGIN

FOR cust IN (

SELECT CustomerID, Balance FROM Customers

) LOOP

IF cust.Balance > 10000 THEN

UPDATE Customers

SET IsVIP = 'TRUE'

WHERE CustomerID = cust.CustomerID;

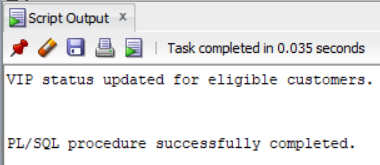
END IF;

END LOOP;

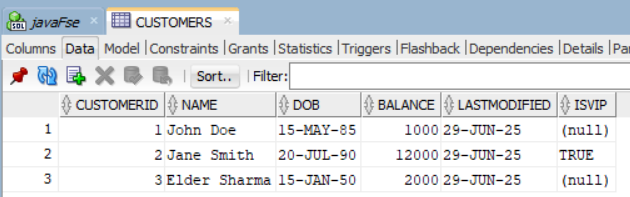
COMMIT;

DBMS\_OUTPUT.PUT\_LINE('VIP status updated for eligible customers.');

END;

**OUTPUT:  
**

**TABLE AFTER RUNNING THE BLOCK:**

****

**Scenario 3:** The bank wants to send reminders to customers whose loans are due within the next 30 days.

**Question:** Write a PL/SQL block that fetches all loans due in the next 30 days and prints a reminder message for each customer.

**PL/SQL BLOCK:**

BEGIN

FOR res IN (

SELECT c.Name, c.CustomerID, l.LoanID, l.EndDate

FROM Loans l

JOIN Customers c ON l.CustomerID = c.CustomerID

WHERE l.EndDate BETWEEN SYSDATE AND SYSDATE + 30

) LOOP

DBMS\_OUTPUT.PUT\_LINE(

'Reminder: Loan (LoanID: ' || res.LoanID || ') for customer "' || res.Name ||

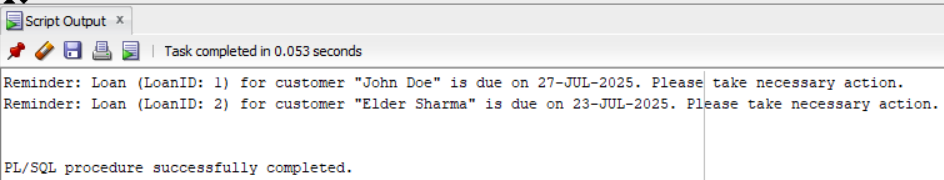
'" is due on ' || TO\_CHAR(res.EndDate, 'DD-MON-YYYY') || '. Please take necessary action.'

);

END LOOP;

END;

**OUTPUT:**

****