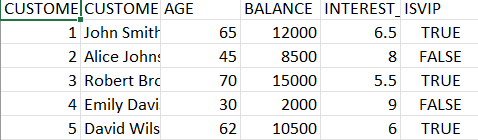
**Exercise 1: Control Structures**

**Database Setup**

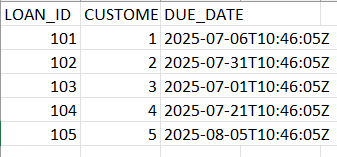
To implement the following scenarios, I first created the required tables customers and loans using PL/SQL and inserted sample data. I also downloaded the table data, and will attach it below for reference.

**Attached Tables:**

* **customers\_table\_output.csv**

****

* **loans\_table\_output.csv**

****

These tables include fields such as customer\_id, customer\_name, age, balance, interest\_rate, IsVIP, and due\_date, which are used in the PL/SQL programs below.

**Scenario 1: Discount for Senior Citizens**

**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.

DECLARE

CURSOR cur\_customers IS

SELECT customer\_id, interest\_rate

FROM customers

WHERE age > 60;

BEGIN

FOR rec IN cur\_customers LOOP

UPDATE customers

SET interest\_rate = interest\_rate - 1

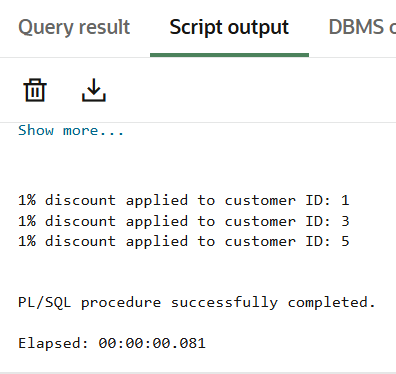
WHERE customer\_id = rec.customer\_id;

DBMS\_OUTPUT.PUT\_LINE('1% discount applied to customer ID: ' || rec.customer\_id);

END LOOP;

END;

/



**Scenario 2: Promote to VIP**

**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.

DECLARE

CURSOR cur\_vip\_check IS

SELECT customer\_id, balance

FROM customers

WHERE balance > 10000;

BEGIN

FOR rec IN cur\_vip\_check LOOP

UPDATE customers

SET IsVIP = 'TRUE'

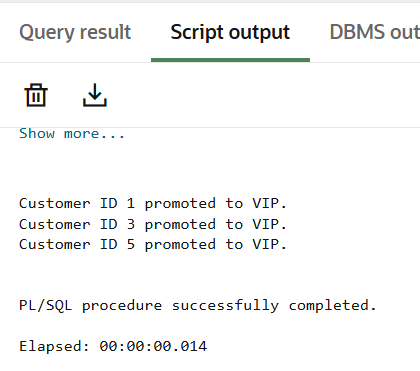
WHERE customer\_id = rec.customer\_id;

DBMS\_OUTPUT.PUT\_LINE('Customer ID ' || rec.customer\_id || ' promoted to VIP.');

END LOOP;

END;

/



**Scenario 3: Loan Payment Reminders**

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

DECLARE

CURSOR cur\_due\_loans IS

SELECT loan\_id, customer\_id, due\_date

FROM loans

WHERE due\_date BETWEEN SYSDATE AND SYSDATE + 30;

BEGIN

FOR rec IN cur\_due\_loans LOOP

DBMS\_OUTPUT.PUT\_LINE('Reminder: Loan ID ' || rec.loan\_id ||

' for Customer ID ' || rec.customer\_id ||

' is due on ' || TO\_CHAR(rec.due\_date, 'DD-MON-YYYY'));

END LOOP;

END;

/

