**Exercise 1: Control Structures**

**Reference tables for the following 3 scenarios:**

SQL script for creation and insertion of values into tables:

Table 1 - Customers:

CREATE TABLE customers(

    c\_id NUMBER,

    name VARCHAR2(50),

    age NUMBER,

    balance NUMBER,

    isVIP BOOLEAN,

    PRIMARY KEY(c\_id)

);

INSERT INTO customers VALUES (1, 'Alice', 65, 12000, 'FALSE');

INSERT INTO customers VALUES (2, 'Bob', 45, 9500, 'FALSE');

INSERT INTO customers VALUES (3, 'Charlie', 70, 3000, 'FALSE');

INSERT INTO customers VALUES (4, 'David', 28, 15000, 'FALSE');

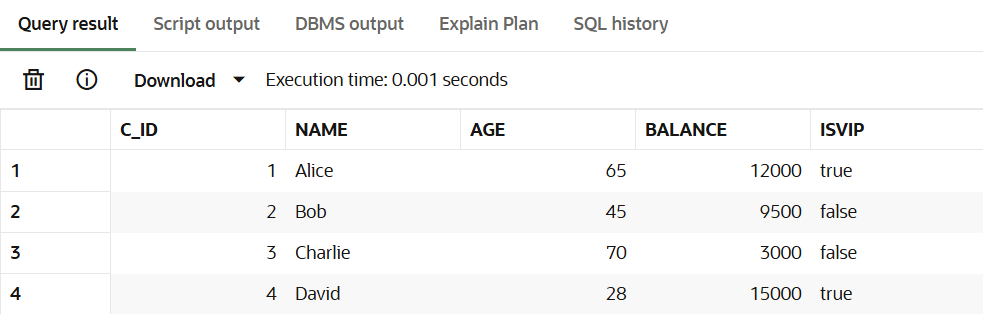


Table 2 – Loans

CREATE TABLE loans(

    l\_id NUMBER,

    c\_id NUMBER,

    interest\_rate NUMBER,

    due\_date DATE,

    PRIMARY KEY(l\_id),

    FOREIGN KEY(c\_id) REFERENCES customers(c\_id)

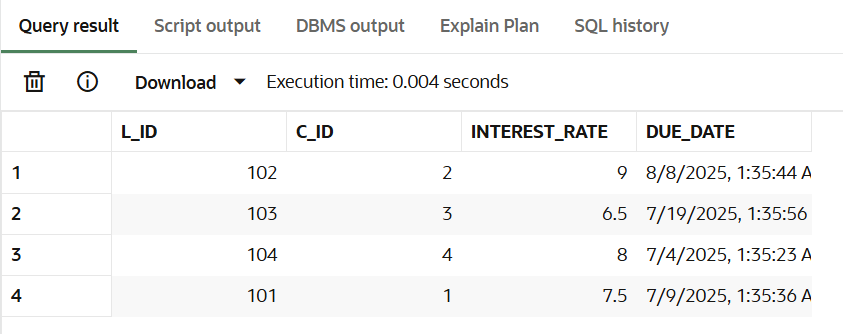
);

INSERT INTO Loans VALUES (101, 1, 8.5, SYSDATE + 10s);

INSERT INTO Loans VALUES (102, 2, 9.0, SYSDATE + 40);

INSERT INTO Loans VALUES (103, 3, 7.5, SYSDATE + 20);

INSERT INTO Loans VALUES (104, 4, 8.0, SYSDATE + 5);



**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.

**Code:**

BEGIN

    FOR i IN (SELECT c\_id FROM customers WHERE age > 60) LOOP

        UPDATE loans

        SET interest\_rate = interest\_rate - 1

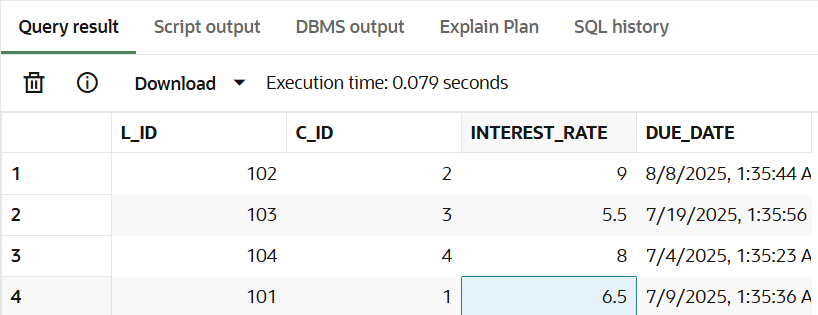
        WHERE c\_id = i.c\_id;

    END LOOP;

    COMMIT;

END;

**Output:**

****

**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.

**Code:**

BEGIN

    FOR i IN (SELECT c\_id from customers WHERE balance> 10000) LOOP

        UPDATE customers

        SET isVIP = TRUE

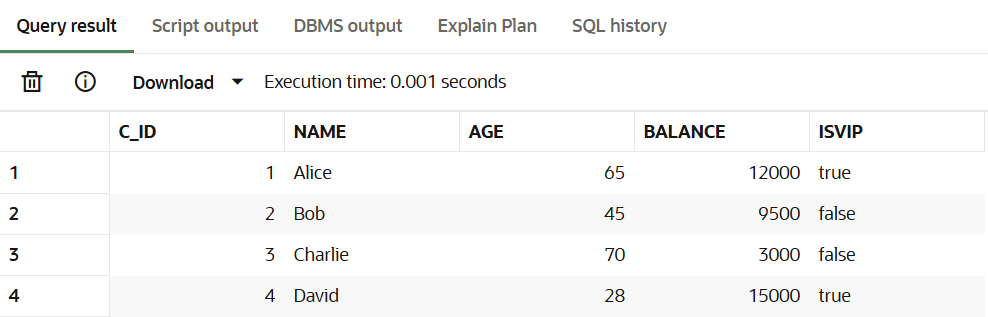
        WHERE c\_id = i.c\_id;

    END LOOP;

    COMMIT;

END;

**Output:**

****

**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.

**Code:**

BEGIN

    FOR i IN (SELECT c\_id, due\_date FROM loans WHERE SYSDATE - due\_date < 30) LOOP

        dbms\_output.put\_line('Customer ID' || TO\_CHAR(i.c\_id) || ' has loan deue in ' || TO\_CHAR(ROUND(i.due\_date - SYSDATE)) || ' days');

    END LOOP;

END;

**Output:**

