Exercise 1: Control Structures

Code:

**Create customers table:**

CREATE TABLE customers (

customer\_id NUMBER PRIMARY KEY,

name VARCHAR2(100),

age NUMBER,

loan\_interest NUMBER(5,2),

balance NUMBER(10,2),

isVIP VARCHAR2(5)

);

**Create loans table:**

CREATE TABLE loans (

loan\_id NUMBER PRIMARY KEY,

customer\_id NUMBER,

due\_date DATE,

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

);

**Insert sample data:**

-- Customers

INSERT INTO customers VALUES (1, 'Alice', 65, 8.5, 15000, 'FALSE');

INSERT INTO customers VALUES (2, 'Bob', 45, 9.0, 9500, 'FALSE');

INSERT INTO customers VALUES (3, 'Charlie', 70, 7.5, 12000, 'FALSE');

-- Loans

INSERT INTO loans VALUES (101, 1, SYSDATE + 15);

INSERT INTO loans VALUES (102, 2, SYSDATE + 35);

INSERT INTO loans VALUES (103, 3, SYSDATE + 10);

COMMIT;

BEGIN

  FOR rec IN (

    SELECT customer\_id, loan\_interest

    FROM customers

    WHERE age > 60

  ) LOOP

    UPDATE customers

    SET loan\_interest = rec.loan\_interest - 1

    WHERE customer\_id = rec.customer\_id;

  END LOOP;

  COMMIT;

END;

/

BEGIN

  FOR rec IN (

    SELECT customer\_id

    FROM customers

    WHERE balance > 10000

  ) LOOP

    UPDATE customers

    SET isVIP = 'TRUE'

    WHERE customer\_id = rec.customer\_id;

  END LOOP;

  COMMIT;

END;

/

SET SERVEROUTPUT ON;

BEGIN

  FOR rec IN (

    SELECT l.loan\_id, l.due\_date, c.name

    FROM loans l

    JOIN customers c ON l.customer\_id = c.customer\_id

    WHERE l.due\_date <= SYSDATE + 30

  ) LOOP

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

                         ' for customer ' || rec.name ||

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

  END LOOP;

END;

/

Output:

A screenshot of a computer

AI-generated content may be incorrect.