Exercise 1: Control Structures

Scenario 1: (Ex1-Scenario1.sql)

```
@InitializeData.sql
SET ECHO ON;
SET SERVEROUTPUT ON SIZE UNLIMITED;
-- Start spooling output to a file
SPOOL output-Ex1-Scenario1.txt;
-- Declare a variable for input
VARIABLE input VARCHAR2(30);

-- Scenario 1
-- Insert sample customers older than 60 years
INSERT INTO CUSTOMERS (
   CUSTOMERID,
   NAME,
   DOB,
   BALANCE,
   LASTMODIFIED
) VALUES (
   1001,
   'John Doe',
   TO_DATE('1950-01-01', 'YYYY-MM-DD'),
   5000,
   SYSDATE
);

INSERT INTO CUSTOMERS (
   CUSTOMERID,
   NAME,
```

```
    DOB,
    BALANCE,
    LASTMODIFIED
) VALUES (
    1002,
    'Jane Smith',
    TO_DATE('1955-01-01', 'YYYY-MM-DD'),
    6000,
    SYSDATE
);

-- Insert loans for these customers
INSERT INTO LOANS (
    LOANID,
    CUSTOMERID,
    LOANAMOUNT,
    INTERESTRATE,
    STARTDATE,
    ENDDATE
) VALUES (
    2001,
    1001,
    10000,
    5,
    SYSDATE - 100,
    SYSDATE + 365
);

INSERT INTO LOANS (
```

```
    LOANID,
    CUSTOMERID,
    LOANAMOUNT,
    INTERESTRATE,
    STARTDATE,
    ENDDATE
) VALUES (
    2002,
    1002,
    15000,
    5,
    SYSDATE - 200,
    SYSDATE + 365
);

-- Update the interest rate for customers older than 60
BEGIN
    FOR CUSTOMER_REC IN (
        SELECT
            CUSTOMERID,
            TRUNC(MONTHS_BETWEEN(SYSDATE, DOB) / 12) AS AGE
        FROM
            CUSTOMERS
    ) LOOP
        IF CUSTOMER_REC.AGE > 60 THEN
            UPDATE LOANS
            SET
                INTERESTRATE = INTERESTRATE - 1
            WHERE
```

```
            CUSTOMERID = CUSTOMER_REC.CUSTOMERID;
      END IF;
   END LOOP;
END;
/

-- Select from loans to see the changes
SELECT
   *
FROM
   LOANS;

-- Stop spooling
SPOOL OFF;

@DropData.sql
```

Scenario 2: (Ex1-Scenario2.sql)

```
@InitializeData.sql

SET ECHO ON;

SET SERVEROUTPUT ON SIZE UNLIMITED;

-- Start spooling output to a file
SPOOL output-Ex1-Scenario2.txt;

VARIABLE input VARCHAR2(30);
```

```
-- Scenario 2

-- Insert sample customers with varying balances
INSERT INTO CUSTOMERS (
    CUSTOMERID,
    NAME,
    DOB,
    BALANCE,
    LASTMODIFIED
) VALUES (
    1003,
    'Kyle',
    TO_DATE('1980-01-01', 'YYYY-MM-DD'),
    15000,
    SYSDATE
);

INSERT INTO CUSTOMERS (
    CUSTOMERID,
    NAME,
    DOB,
    BALANCE,
    LASTMODIFIED
) VALUES (
    1004,
    'Zach',
    TO_DATE('1985-01-01', 'YYYY-MM-DD'),
    8000,
    SYSDATE
```

```
);

ALTER TABLE CUSTOMERS
   ADD (
      ISVIP VARCHAR2(5)
   );

BEGIN
   FOR CUSTOMER_REC IN (
      SELECT
         CUSTOMERID,
         BALANCE
      FROM
         CUSTOMERS
   ) LOOP
      IF CUSTOMER_REC.BALANCE > 10000 THEN
         UPDATE CUSTOMERS
         SET
            ISVIP='TRUE'
         WHERE
            CUSTOMERID = CUSTOMER_REC.CUSTOMERID;
      ELSE
         UPDATE CUSTOMERS
         SET
            ISVIP='FALSE'
         WHERE
            CUSTOMERID = CUSTOMER_REC.CUSTOMERID;
      END IF;
   END LOOP;
```

```
END;
/

-- Select from customers to see the changes
SELECT
   *
FROM
   CUSTOMERS;

-- Stop spooling
SPOOL OFF

@DropData.sql
```

Scenario 3: (Ex1-Scenario3.sql)

```
@InitializeData.sql

SET ECHO ON

SET SERVEROUTPUT on SIZE UNLIMITED

SPOOL output-Ex1-Scenario3.txt

VARIABLE input VARCHAR2(30)

-- Scenario 3

-- Insert sample customer
INSERT INTO CUSTOMERS (
```

```
    CUSTOMERID,
    NAME,
    DOB,
    BALANCE,
    LASTMODIFIED
) VALUES (
    1005,
    'Charlie Davis',
    TO_DATE('1990-01-01', 'YYYY-MM-DD'),
    7000,
    SYSDATE
);

-- Insert loan for the customer with end date within the next 30 days
INSERT INTO LOANS (
    LOANID,
    CUSTOMERID,
    LOANAMOUNT,
    INTERESTRATE,
    STARTDATE,
    ENDDATE
) VALUES (
    2003,
    1005,
    20000,
    4,
    SYSDATE - 300,
    SYSDATE + 10
);
```

```
BEGIN
   FOR LOAN_REC IN (
      SELECT
         L.LOANID,
         L.CUSTOMERID,
         C.NAME,
         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 ID '
                           || LOAN_REC.LOANID
                           || ' for customer '
                           || LOAN_REC.NAME
                           || ' is due on '
                           || TO_CHAR(LOAN_REC.ENDDATE, 'YYYY-MM-DD'));
   END LOOP;
END;
/

SPOOL OFF

@DropData.sql
```