**Exercise – 1 Control Statements**

**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**

CREATE TABLE CUSTOMER (

  customer\_id INT,

  customer\_name VARCHAR(15),

  age INT,

  interest\_rate NUMBER(5,2),

  loan\_amount NUMBER(10,2)

);

INSERT INTO CUSTOMER VALUES (1, 'Adam', 51, 7.50, 15000);

INSERT INTO CUSTOMER VALUES (2, 'Ben', 62, 6.80, 20000);

INSERT INTO CUSTOMER VALUES (3, 'Charlie', 60, 7.00, 12000);

INSERT INTO CUSTOMER VALUES (4, 'David', 46, 6.75, 13000);

INSERT INTO CUSTOMER VALUES (5, 'Eva', 32, 7.10, 8000);

INSERT INTO CUSTOMER VALUES (6, 'Frank', 65, 8.00, 25000);

INSERT INTO CUSTOMER VALUES (7, 'Emma', 25, 7.90, 5000);

select \* from customer;

BEGIN

  UPDATE CUSTOMER

  SET interest\_rate = interest\_rate - 1

  WHERE age > 60;

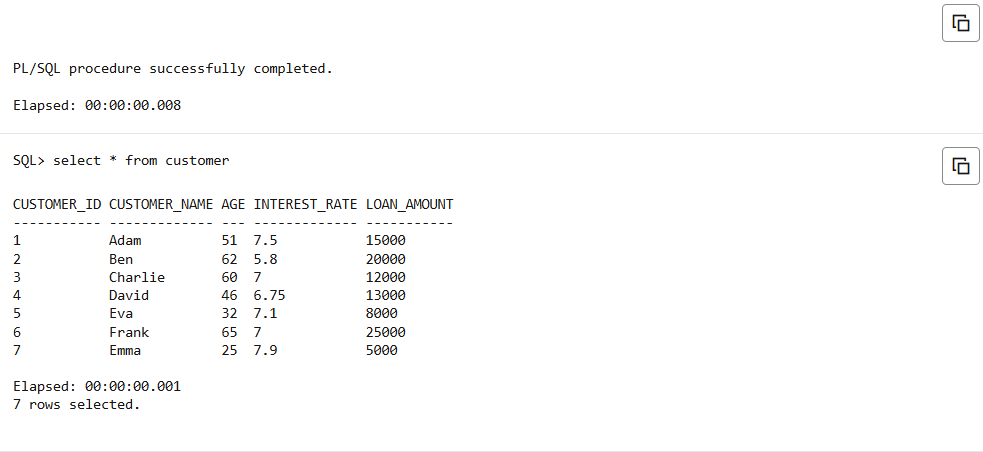
  COMMIT;

END;

/

select \* from customer;

**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**

ALTER TABLE CUSTOMER ADD (IsVIP VARCHAR2(5) DEFAULT 'FALSE');

select \* from customer;

BEGIN

  UPDATE CUSTOMER

  SET IsVIP = 'TRUE'

  WHERE loan\_amount > 10000;

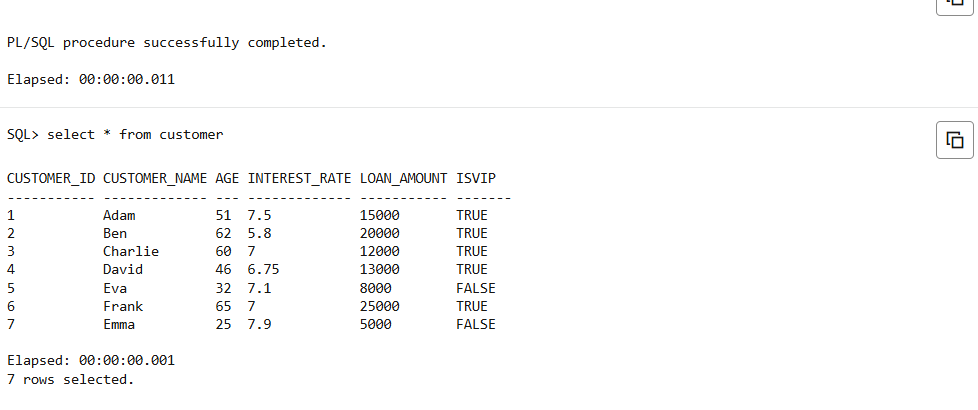
  COMMIT;

END;

/

select \* from customer;

**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**

ALTER TABLE CUSTOMER ADD (loan\_due\_date DATE);

TRUNCATE TABLE CUSTOMER;

ALTER TABLE CUSTOMER DROP COLUMN ISVIP;

INSERT INTO CUSTOMER VALUES (1, 'Adam', 51, 7.50, 15000, TO\_DATE('2025-05-25', 'YYYY-MM-DD'));

INSERT INTO CUSTOMER VALUES (2, 'Ben', 62, 6.80, 20000, TO\_DATE('2025-05-29', 'YYYY-MM-DD'));

INSERT INTO CUSTOMER VALUES (3, 'Charlie', 60, 7.00, 12000, TO\_DATE('2025-04-31', 'YYYY-MM-DD'));

INSERT INTO CUSTOMER VALUES (4, 'David', 46, 6.75, 13000, TO\_DATE('2025-06-10', 'YYYY-MM-DD'));

INSERT INTO CUSTOMER VALUES (5, 'Eva', 32, 7.10, 8000, TO\_DATE('2025-05-28', 'YYYY-MM-DD'));

INSERT INTO CUSTOMER VALUES (6, 'Frank', 65, 8.00, 25000, TO\_DATE('2025-06-05', 'YYYY-MM-DD'));

INSERT INTO CUSTOMER VALUES (7, 'Emma', 25, 7.90, 5000, TO\_DATE('2025-04-30', 'YYYY-MM-DD'));

select \* from customer;

SET SERVEROUTPUT ON;

BEGIN

  FOR cust IN (

    SELECT customer\_id, customer\_name, loan\_due\_date

    FROM CUSTOMER

    WHERE loan\_due\_date BETWEEN SYSDATE AND SYSDATE + 30

  )

  LOOP

    DBMS\_OUTPUT.PUT\_LINE('Reminder: Loan for customer ' || cust.customer\_name ||

    ' (ID: ' || cust.customer\_id || ') is due soon.');

  END LOOP;

END;

/