WEEK-2

Nipuna Amanapu

Superset id: 6432842

**Exercise 1: Control Structures**

**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.

Solution:

SET SERVEROUTPUT ON;

BEGIN

  EXECUTE IMMEDIATE 'DROP TABLE customers';

EXCEPTION

  WHEN OTHERS THEN

    IF SQLCODE != -942 THEN

      RAISE;

    END IF;

END;

/

CREATE TABLE customers (

  customer\_id NUMBER PRIMARY KEY,

  name VARCHAR2(50),

  age NUMBER,

  loan\_interest\_rate NUMBER(5,2)

);

INSERT INTO customers VALUES (1, 'Alice', 45, 7.5);

INSERT INTO customers VALUES (2, 'Bob', 62, 8.0);

INSERT INTO customers VALUES (3, 'Charlie', 70, 7.2);

INSERT INTO customers VALUES (4, 'Diana', 30, 6.5);

COMMIT;

BEGIN

  FOR rec IN (SELECT \* FROM customers) LOOP

    IF rec.age > 60 THEN

      UPDATE customers

      SET loan\_interest\_rate = loan\_interest\_rate - 1

      WHERE customer\_id = rec.customer\_id;

      DBMS\_OUTPUT.PUT\_LINE('Discount applied to ' || rec.name ||

                           ' | New rate: ' ||

                           TO\_CHAR(rec.loan\_interest\_rate - 1));

    END IF;

  END LOOP;

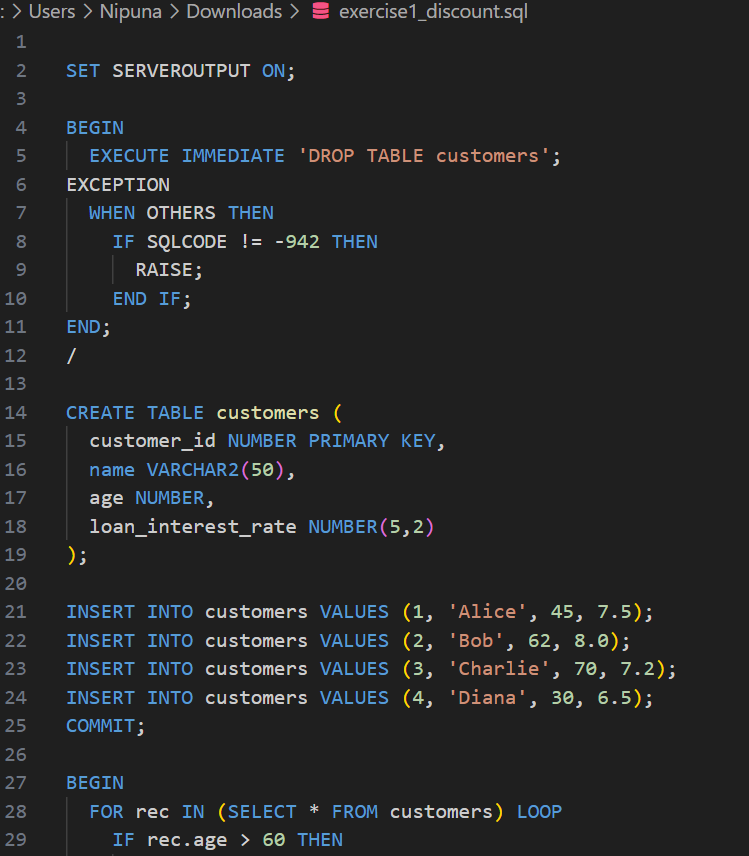
END;

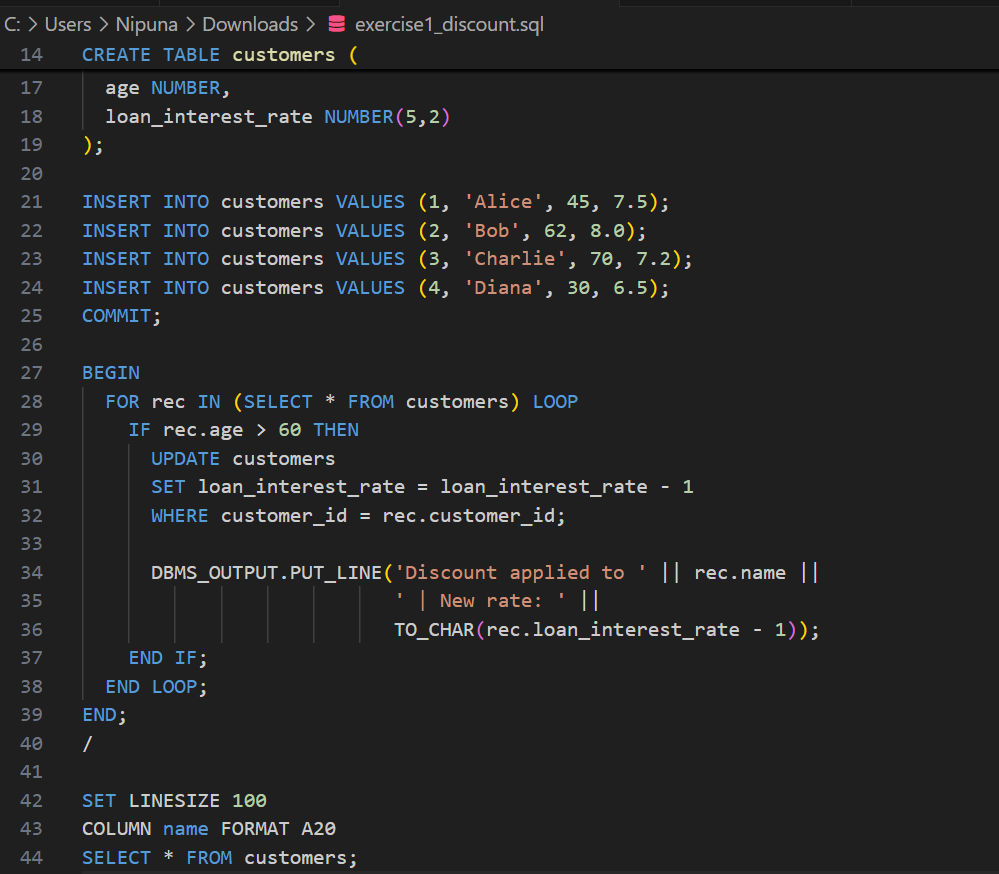
/

SET LINESIZE 100

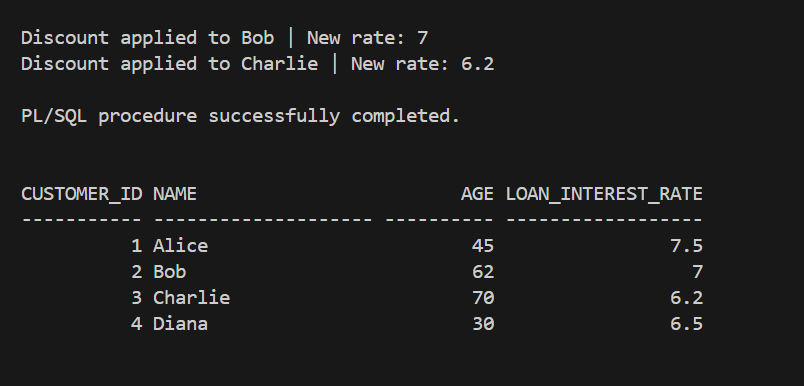
COLUMN name FORMAT A20

SELECT \* FROM customers;





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.

Solution:

SET SERVEROUTPUT ON;

CREATE TABLE vip\_customers (

  customer\_id NUMBER PRIMARY KEY,

  name VARCHAR2(50),

  balance NUMBER(10,2),

  is\_vip VARCHAR2(5) DEFAULT 'FALSE'

);

INSERT INTO vip\_customers VALUES (1, 'Alice', 7500.00, 'FALSE');

INSERT INTO vip\_customers VALUES (2, 'Bob', 12000.50, 'FALSE');

INSERT INTO vip\_customers VALUES (3, 'Charlie', 30000.00, 'FALSE');

INSERT INTO vip\_customers VALUES (4, 'Diana', 5000.00, 'FALSE');

COMMIT;

BEGIN

  FOR rec IN (SELECT \* FROM vip\_customers) LOOP

    IF rec.balance > 10000 THEN

      UPDATE vip\_customers

      SET is\_vip = 'TRUE'

      WHERE customer\_id = rec.customer\_id;

      DBMS\_OUTPUT.PUT\_LINE('VIP promoted: ' || rec.name ||

                           ' | Balance: $' || TO\_CHAR(rec.balance));

    END IF;

  END LOOP;

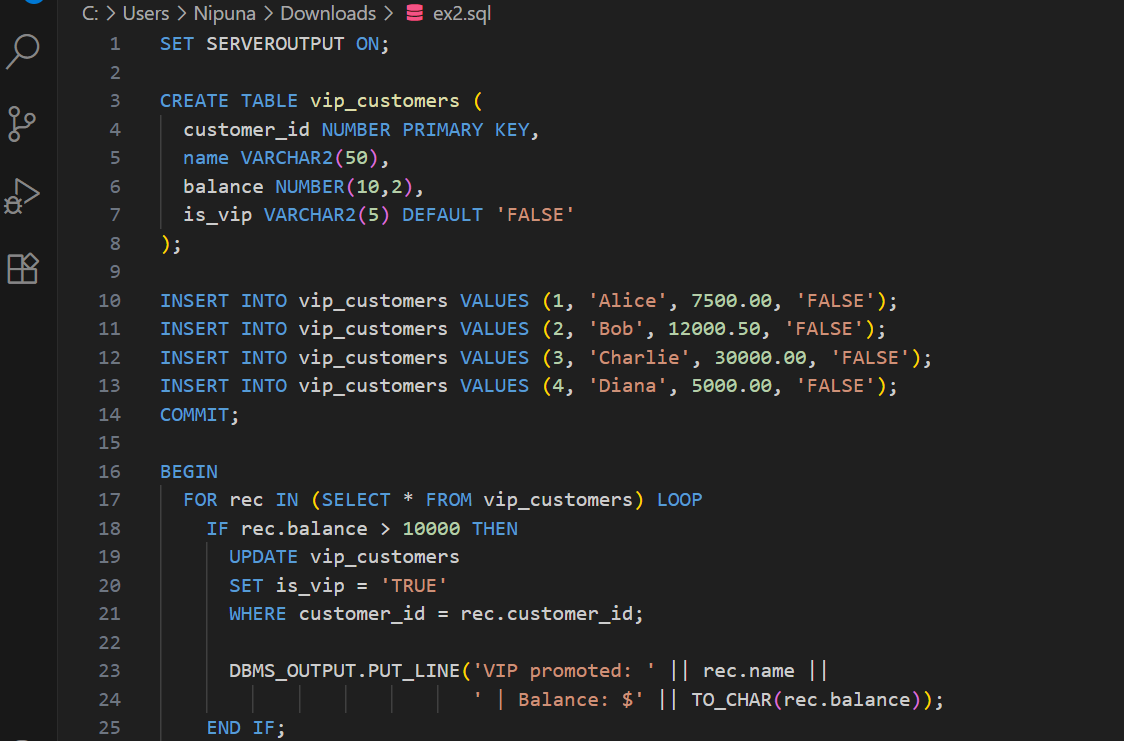
END;

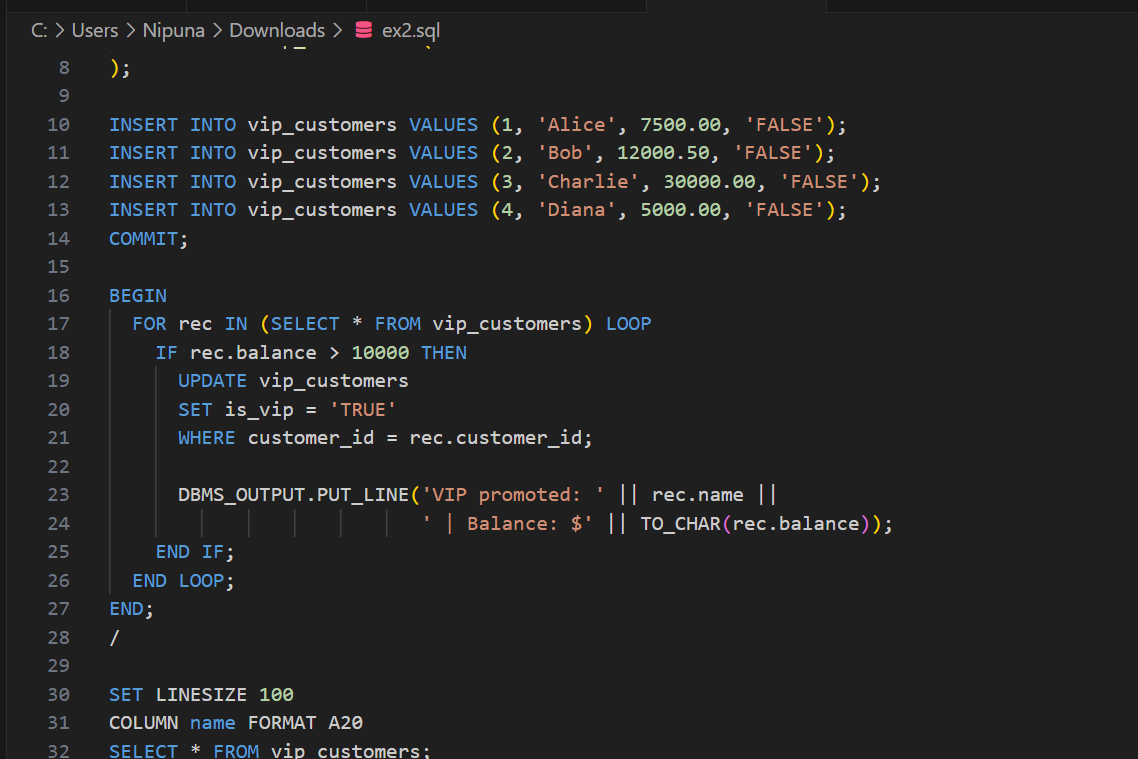
/

SET LINESIZE 100

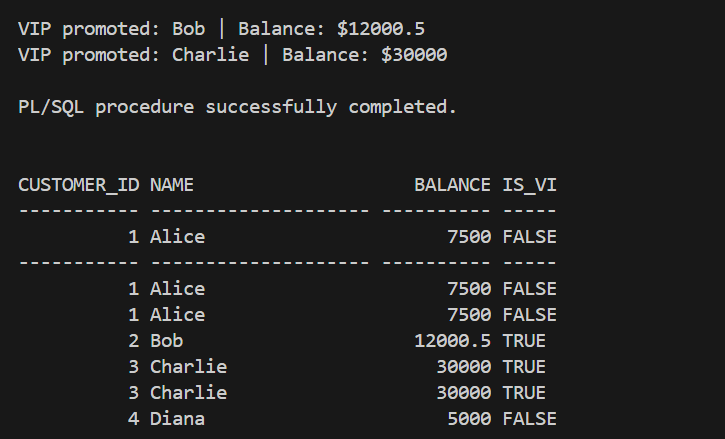
COLUMN name FORMAT A20

SELECT \* FROM vip\_customers;





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.

Solution:

SET SERVEROUTPUT ON;

CREATE TABLE loan\_customers (

  customer\_id NUMBER PRIMARY KEY,

  name VARCHAR2(50),

  loan\_amount NUMBER(10, 2),

  due\_date DATE

);

INSERT INTO loan\_customers VALUES (1, 'Alice', 10000, SYSDATE + 10);

INSERT INTO loan\_customers VALUES (2, 'Bob', 15000, SYSDATE + 40);

INSERT INTO loan\_customers VALUES (3, 'Charlie', 20000, SYSDATE + 25);

INSERT INTO loan\_customers VALUES (4, 'Diana', 12000, SYSDATE - 5);

COMMIT;

BEGIN

  FOR rec IN (

    SELECT \* FROM loan\_customers

    WHERE due\_date BETWEEN SYSDATE AND SYSDATE + 30

  ) LOOP

    DBMS\_OUTPUT.PUT\_LINE('Reminder: ' || rec.name ||

                         ' has a loan of $' || TO\_CHAR(rec.loan\_amount) ||

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

  END LOOP;

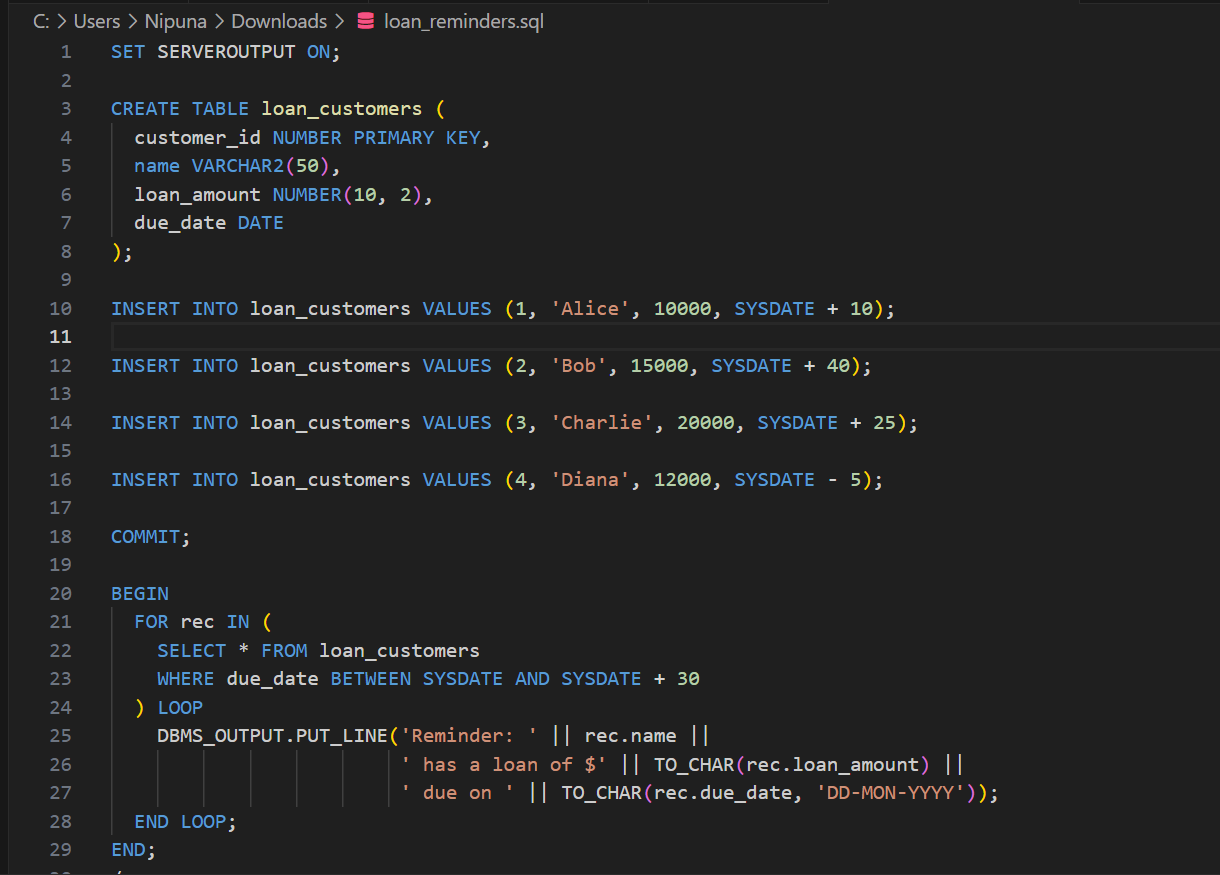
END;

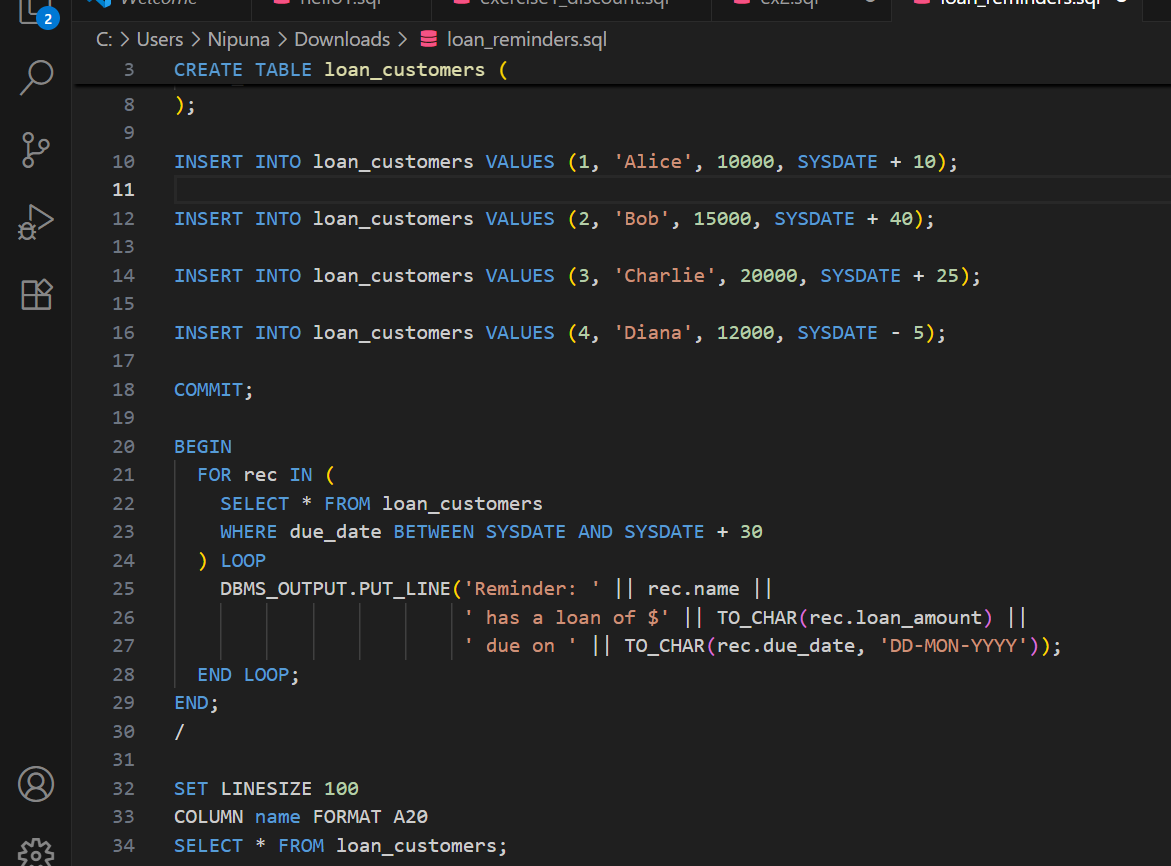
/

SET LINESIZE 100

COLUMN name FORMAT A20

SELECT \* FROM loan\_customers;





Output:

