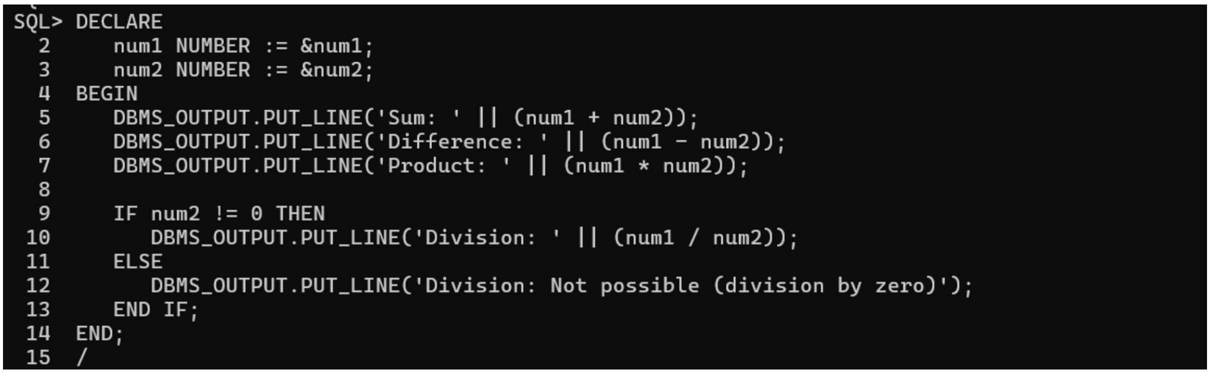
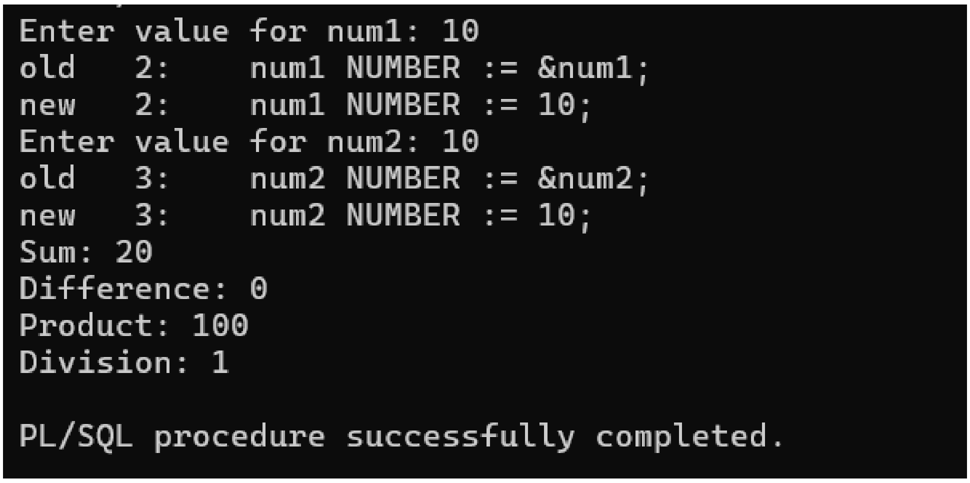
ORACLE LAB ASSIGNMENT – 7

Q.1 Write a PL/SQL program which takes two number as input and perform arithmetic operations



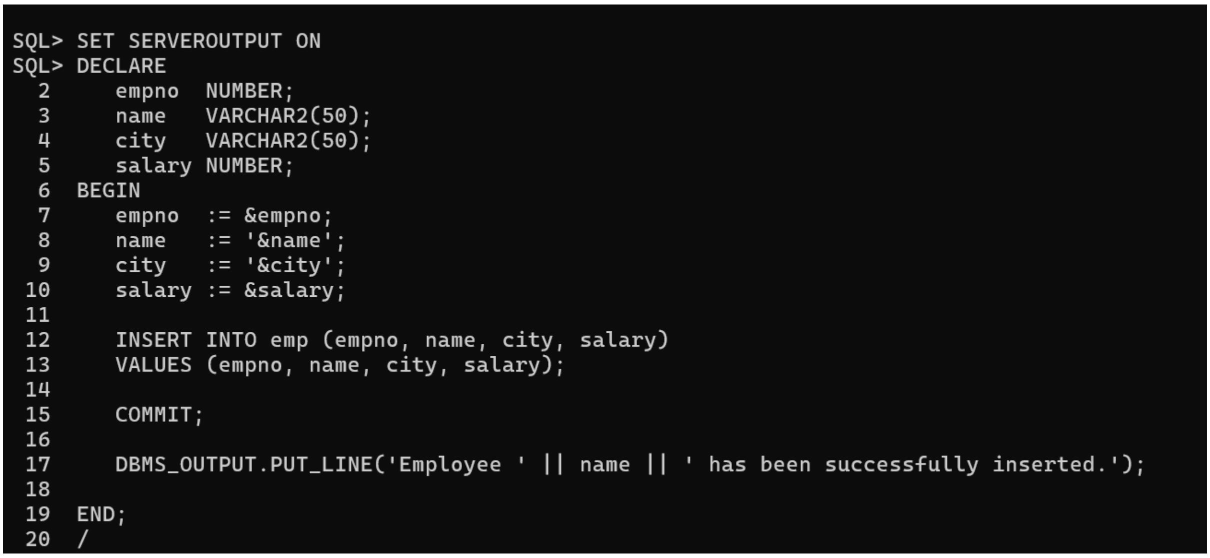
```
SQL> DECLARE
        num1 NUMBER := &num1;
        num2 NUMBER := &num2;
     BEGIN
        DBMS_OUTPUT.PUT_LINE('Sum: ' || (num1 + num2));
        DBMS_OUTPUT.PUT_LINE('Difference: ' || (num1 - num2));
        DBMS_OUTPUT.PUT_LINE('Product: ' || (num1 * num2));

        IF num2 != 0 THEN
           DBMS_OUTPUT.PUT_LINE('Division: ' || (num1 / num2));
        ELSE
           DBMS_OUTPUT.PUT_LINE('Division: Not possible (division by zero)');
        END IF;
     END;
     /
```

- OUTPUT



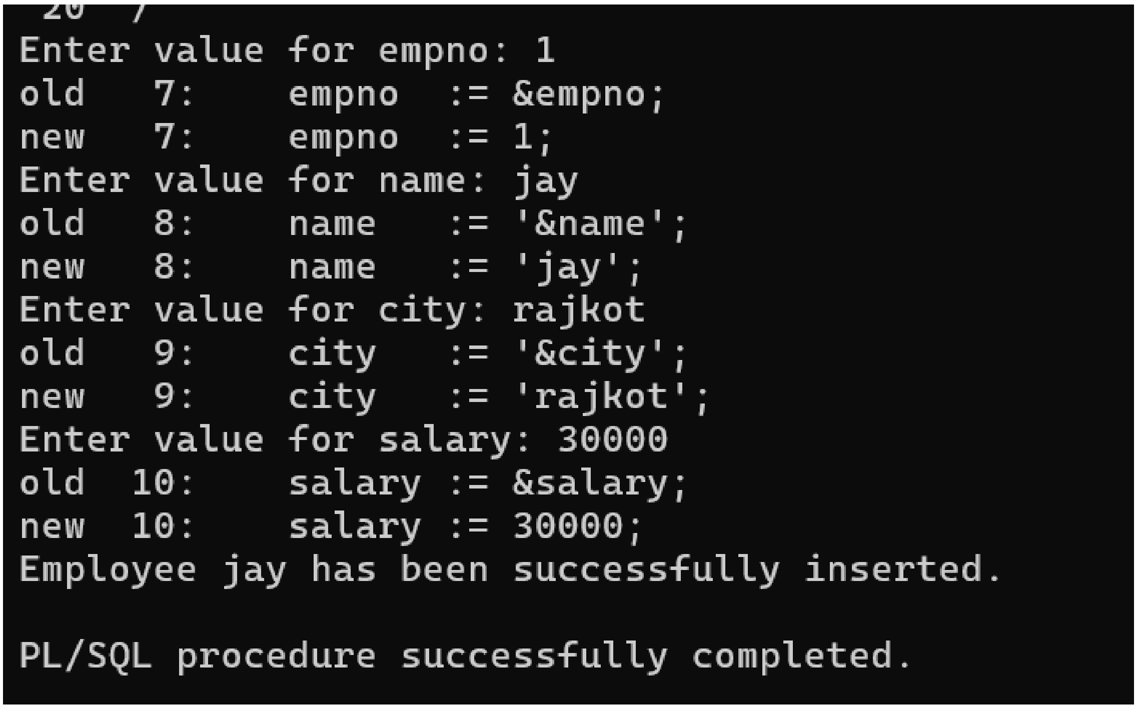
```
Enter value for num1: 10
old   2:    num1 NUMBER := &num1;
new   2:    num1 NUMBER := 10;
Enter value for num2: 10
old   3:    num2 NUMBER := &num2;
new   3:    num2 NUMBER := 10;
Sum: 20
Difference: 0
Product: 100
Division: 1

PL/SQL procedure successfully completed.
```

Q.2 Write PL/SQL block that will take values from the user (empno, name, city, and salary), insert them into the emp table, and display a message confirming that the data has been successfully entered.



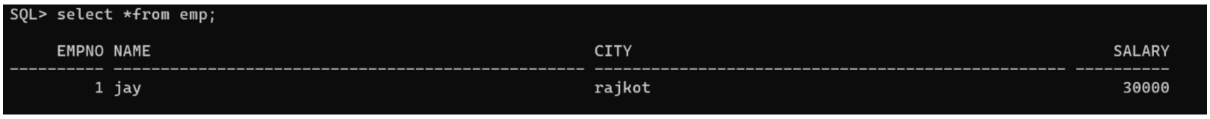
```
SQL> SET SERVEROUTPUT ON
SQL> DECLARE
        empno  NUMBER;
        name   VARCHAR2(50);
        city   VARCHAR2(50);
        salary NUMBER;
     BEGIN
        empno  := &empno;
        name   := '&name';
        city   := '&city';
        salary := &salary;

        INSERT INTO emp (empno, name, city, salary)
        VALUES (empno, name, city, salary);

        COMMIT;

        DBMS_OUTPUT.PUT_LINE('Employee ' || name || ' has been successfully inserted.');

     END;
     /
```

- INSERT VALUE



```
 20  /
Enter value for empno: 1
old   7:    empno  := &empno;
new   7:    empno  := 1;
Enter value for name: jay
old   8:    name   := '&name';
new   8:    name   := 'jay';
Enter value for city: rajkot
old   9:    city   := '&city';
new   9:    city   := 'rajkot';
Enter value for salary: 30000
old  10:    salary := &salary;
new  10:    salary := 30000;
Employee jay has been successfully inserted.

PL/SQL procedure successfully completed.
```

- OUTPUT



```
SQL> select *from emp;

     EMPNO NAME                                                CITY                                                   SALARY
---------- --------------------------------------------------  -------------------------------------------------- ----------
         1 jay                                                 rajkot                                                  30000
```

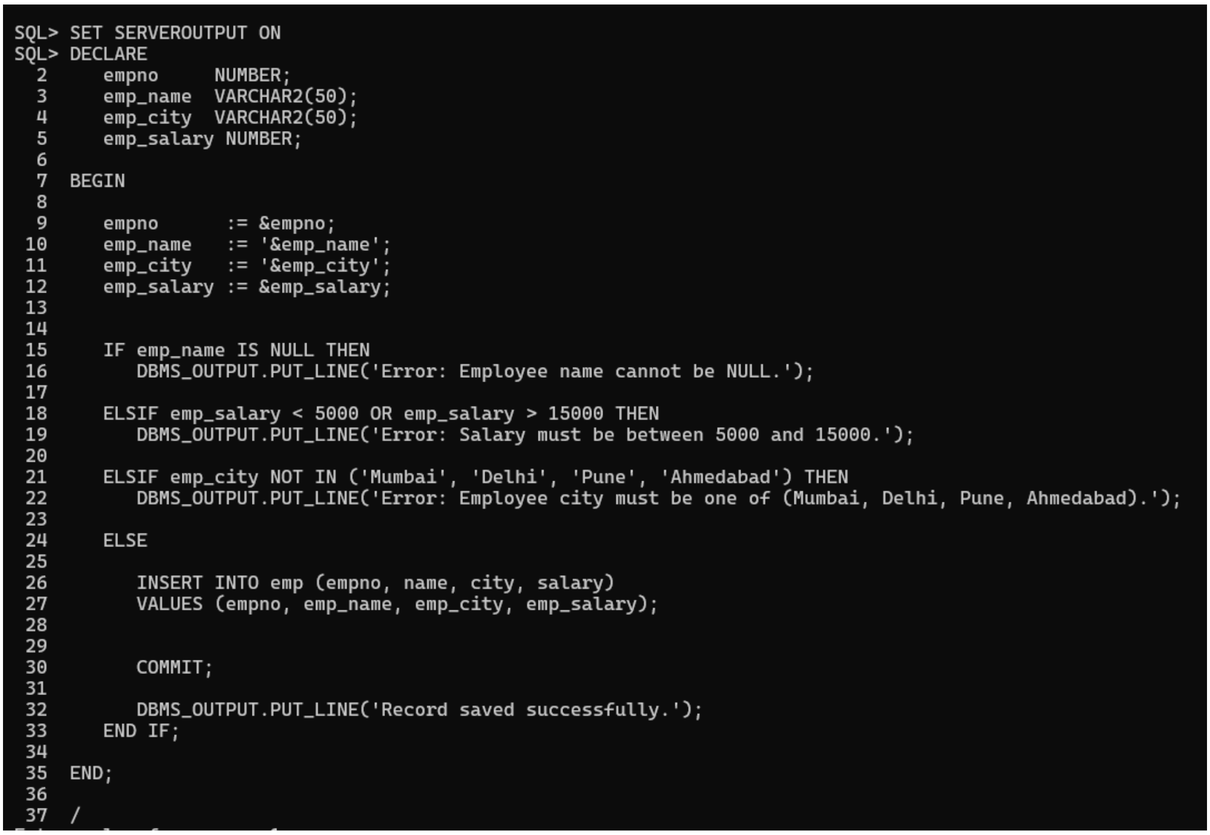
Q.3 Write a PL/SQL block that accepts employee data (Employee Name, Employee City, and Salary) as input, and before inserting the data into the Emp table, ensure the following conditions are met.

• The salary must be between 5000 and 15000.

• The employee's city must be (Mumbai, Delhi, Pune, Ahmedabad).

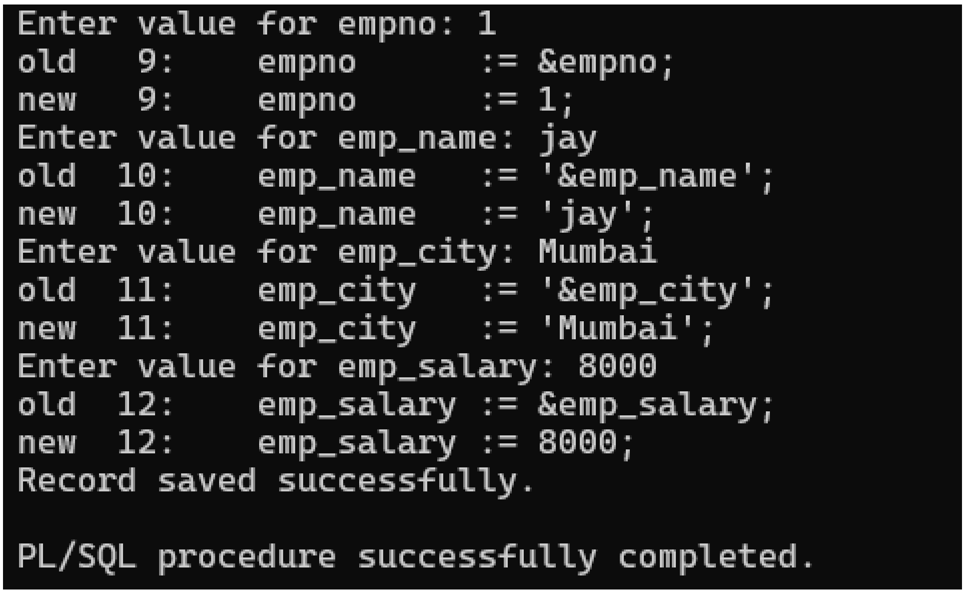
• The employee's name must not be NULL.

If any of the above conditions fail, display an appropriate error message. Otherwise, insert the data into the Emp table and display a success message: "Record saved successfully.

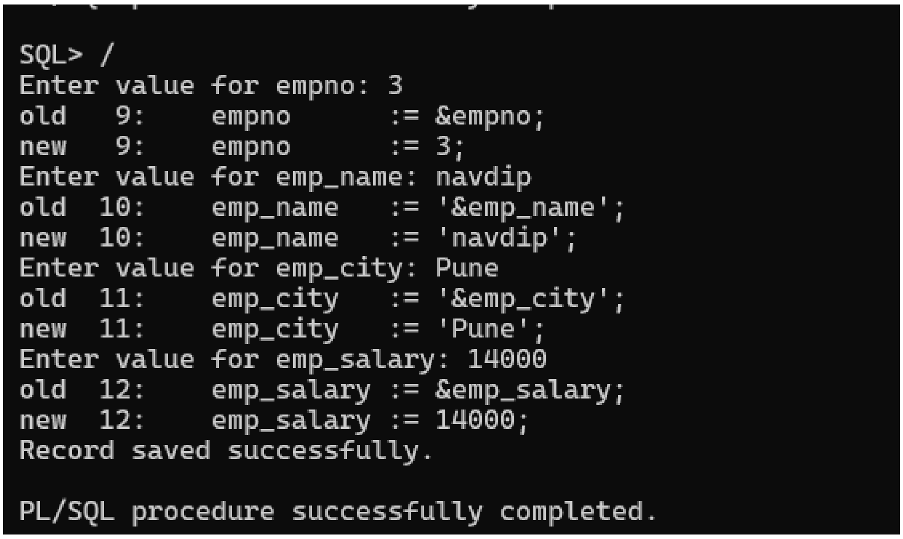


```
SQL> SET SERVEROUTPUT ON
SQL> DECLARE
  2     empno     NUMBER;
  3     emp_name  VARCHAR2(50);
  4     emp_city  VARCHAR2(50);
  5     emp_salary NUMBER;
  6
  7  BEGIN
  8
  9     empno      := &empno;
 10     emp_name   := '&emp_name';
 11     emp_city   := '&emp_city';
 12     emp_salary := &emp_salary;
 13
 14
 15     IF emp_name IS NULL THEN
 16        DBMS_OUTPUT.PUT_LINE('Error: Employee name cannot be NULL.');
 17
 18     ELSIF emp_salary < 5000 OR emp_salary > 15000 THEN
 19        DBMS_OUTPUT.PUT_LINE('Error: Salary must be between 5000 and 15000.');
 20
 21     ELSIF emp_city NOT IN ('Mumbai', 'Delhi', 'Pune', 'Ahmedabad') THEN
 22        DBMS_OUTPUT.PUT_LINE('Error: Employee city must be one of (Mumbai, Delhi, Pune, Ahmedabad).');
 23
 24     ELSE
 25
 26        INSERT INTO emp (empno, name, city, salary)
 27        VALUES (empno, emp_name, emp_city, emp_salary);
 28
 29
 30        COMMIT;
 31
 32        DBMS_OUTPUT.PUT_LINE('Record saved successfully.');
 33     END IF;
 34
 35  END;
 36
 37  /
```

- INSERT VALUE

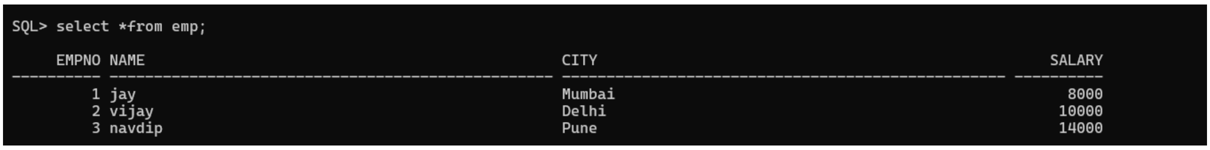


```
Enter value for empno: 1
old   9:    empno      := &empno;
new   9:    empno      := 1;
Enter value for emp_name: jay
old  10:    emp_name   := '&emp_name';
new  10:    emp_name   := 'jay';
Enter value for emp_city: Mumbai
old  11:    emp_city   := '&emp_city';
new  11:    emp_city   := 'Mumbai';
Enter value for emp_salary: 8000
old  12:    emp_salary := &emp_salary;
new  12:    emp_salary := 8000;
Record saved successfully.

PL/SQL procedure successfully completed.
```



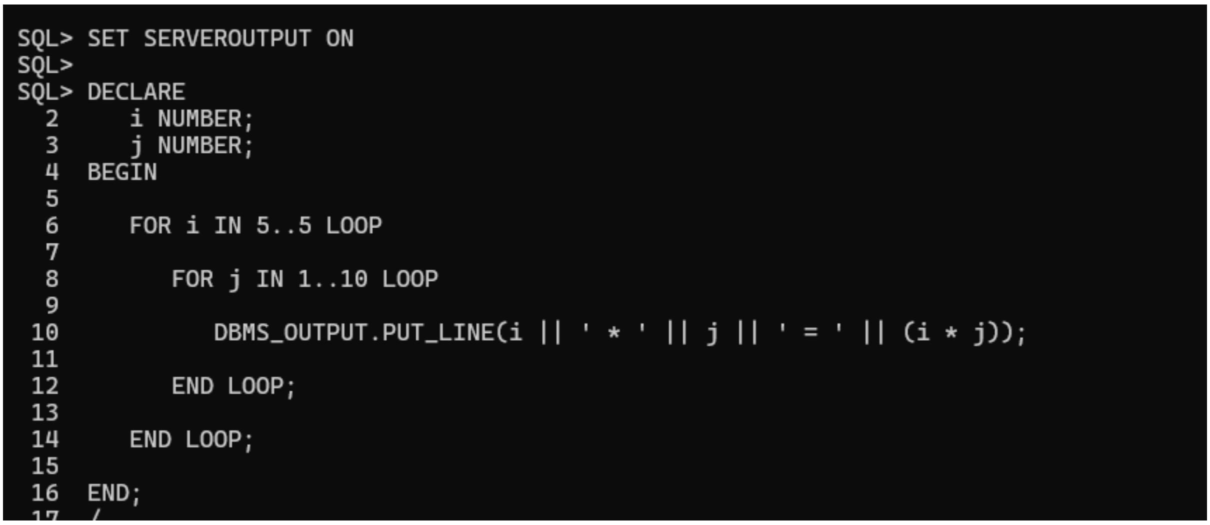
```
SQL> /
Enter value for empno: 3
old   9:    empno      := &empno;
new   9:    empno      := 3;
Enter value for emp_name: navdip
old  10:    emp_name   := '&emp_name';
new  10:    emp_name   := 'navdip';
Enter value for emp_city: Pune
old  11:    emp_city   := '&emp_city';
new  11:    emp_city   := 'Pune';
Enter value for emp_salary: 14000
old  12:    emp_salary := &emp_salary;
new  12:    emp_salary := 14000;
Record saved successfully.

PL/SQL procedure successfully completed.
```

- OUTPUT

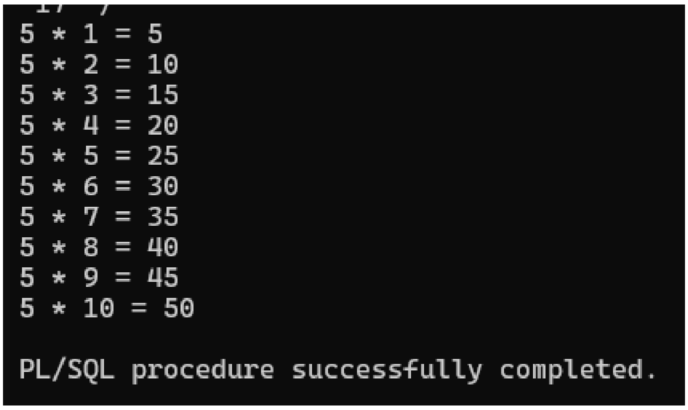


```
SQL> select *from emp;

     EMPNO NAME                                               CITY                                               SALARY
---------- -------------------------------------------------- -------------------------------------------------- ----------
         1 jay                                                Mumbai                                               8000
         2 vijay                                              Delhi                                               10000
         3 navdip                                             Pune                                                14000
```

Q.4 Write PL/SQL block using nested loops to generate and print a multiplication table.



```
SQL> SET SERVEROUTPUT ON
SQL>
SQL> DECLARE
       i NUMBER;
       j NUMBER;
     BEGIN

       FOR i IN 5..5 LOOP

          FOR j IN 1..10 LOOP

             DBMS_OUTPUT.PUT_LINE(i || ' * ' || j || ' = ' || (i * j));

          END LOOP;

       END LOOP;

     END;
     /
```



```
5 * 1 = 5
5 * 2 = 10
5 * 3 = 15
5 * 4 = 20
5 * 5 = 25
5 * 6 = 30
5 * 7 = 35
5 * 8 = 40
5 * 9 = 45
5 * 10 = 50

PL/SQL procedure successfully completed.
```