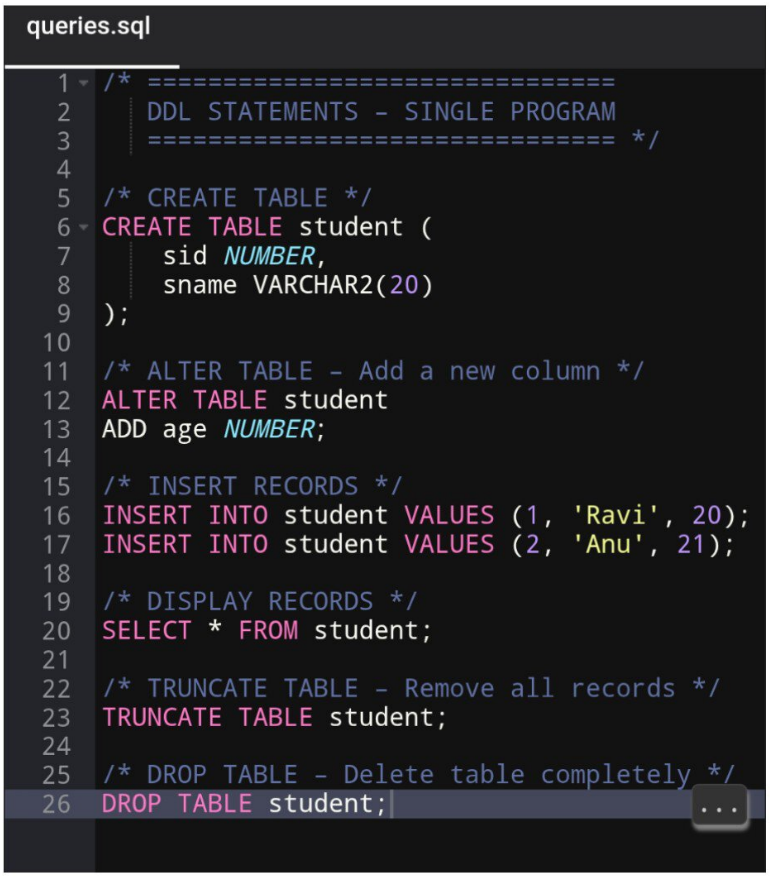
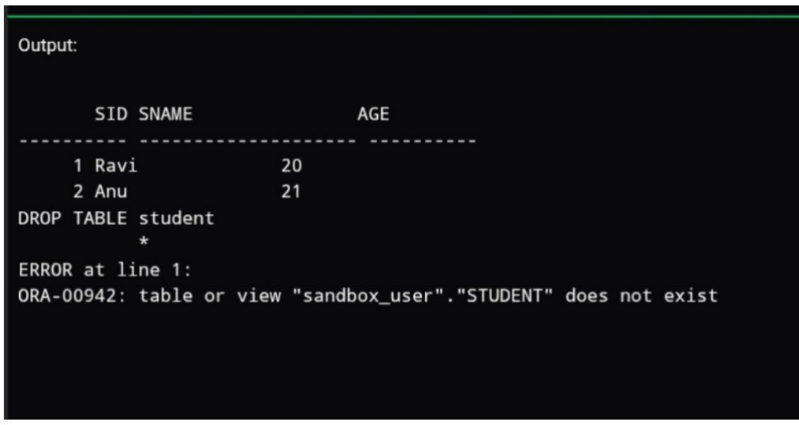


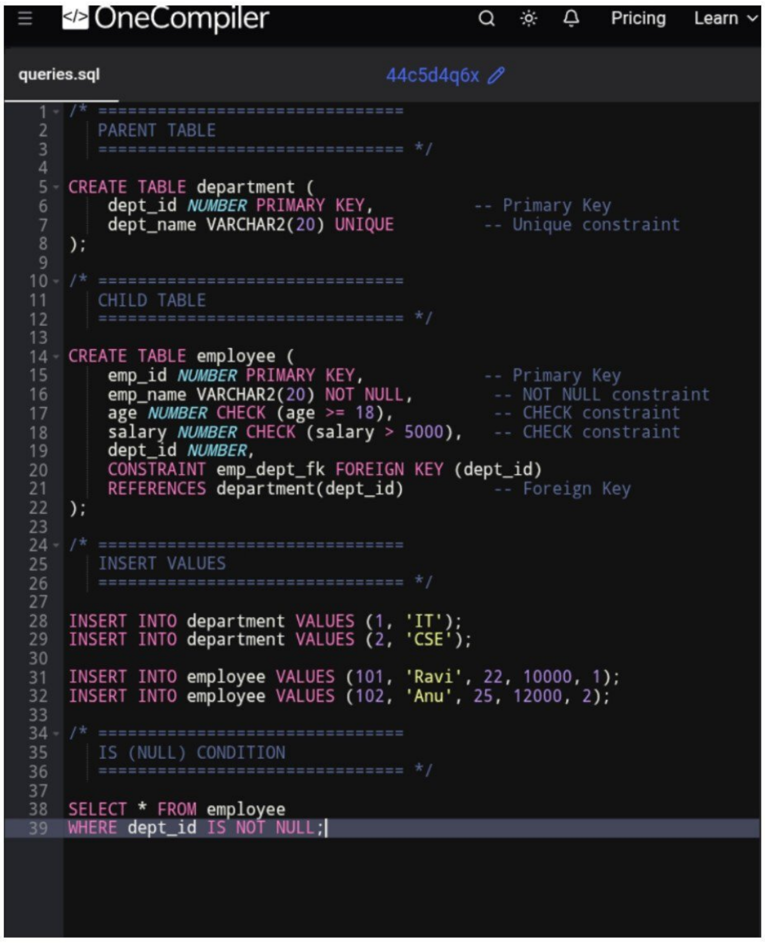
queries.sql

```sql
/* ================================
   DDL STATEMENTS - SINGLE PROGRAM
   ================================ */

/* CREATE TABLE */
CREATE TABLE student (
    sid NUMBER,
    sname VARCHAR2(20)
);

/* ALTER TABLE - Add a new column */
ALTER TABLE student
ADD age NUMBER;

/* INSERT RECORDS */
INSERT INTO student VALUES (1, 'Ravi', 20);
INSERT INTO student VALUES (2, 'Anu', 21);

/* DISPLAY RECORDS */
SELECT * FROM student;

/* TRUNCATE TABLE - Remove all records */
TRUNCATE TABLE student;

/* DROP TABLE - Delete table completely */
DROP TABLE student;
```

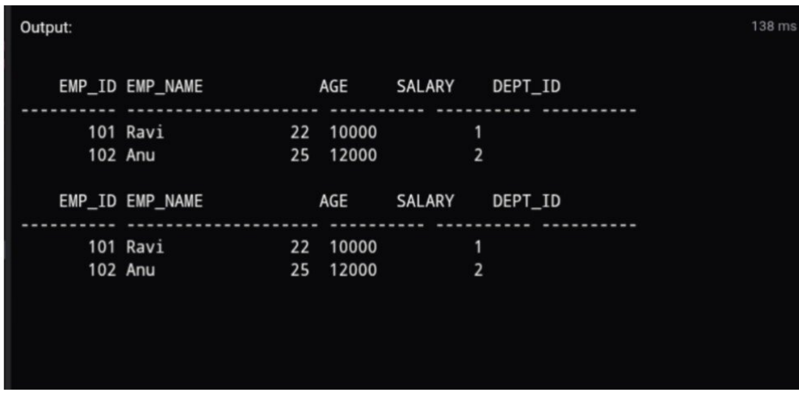


Output:

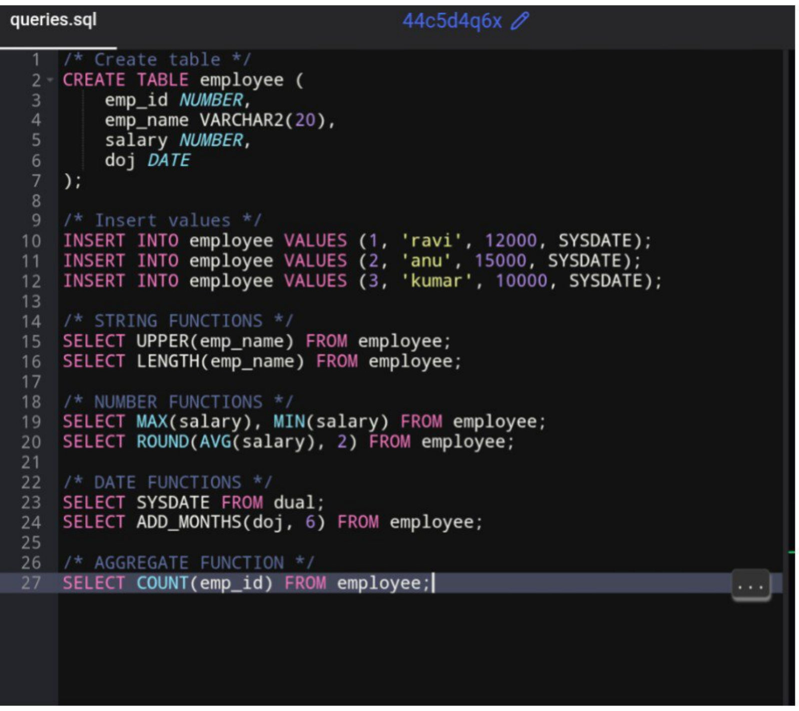
```
       SID SNAME                       AGE
---------- -------------------- ----------
         1 Ravi                         20
         2 Anu                          21
DROP TABLE student
           *
ERROR at line 1:
ORA-00942: table or view "sandbox_user"."STUDENT" does not exist
```



```sql
/* ==============================
   PARENT TABLE
   ============================== */

CREATE TABLE department (
    dept_id NUMBER PRIMARY KEY,          -- Primary Key
    dept_name VARCHAR2(20) UNIQUE         -- Unique constraint
);

/* ==============================
   CHILD TABLE
   ============================== */

CREATE TABLE employee (
    emp_id NUMBER PRIMARY KEY,                -- Primary Key
    emp_name VARCHAR2(20) NOT NULL,           -- NOT NULL constraint
    age NUMBER CHECK (age >= 18),             -- CHECK constraint
    salary NUMBER CHECK (salary > 5000),      -- CHECK constraint
    dept_id NUMBER,
    CONSTRAINT emp_dept_fk FOREIGN KEY (dept_id)
    REFERENCES department(dept_id)            -- Foreign Key
);

/* ==============================
   INSERT VALUES
   ============================== */

INSERT INTO department VALUES (1, 'IT');
INSERT INTO department VALUES (2, 'CSE');

INSERT INTO employee VALUES (101, 'Ravi', 22, 10000, 1);
INSERT INTO employee VALUES (102, 'Anu', 25, 12000, 2);

/* ==============================
   IS (NULL) CONDITION
   ============================== */

SELECT * FROM employee
WHERE dept_id IS NOT NULL;
```

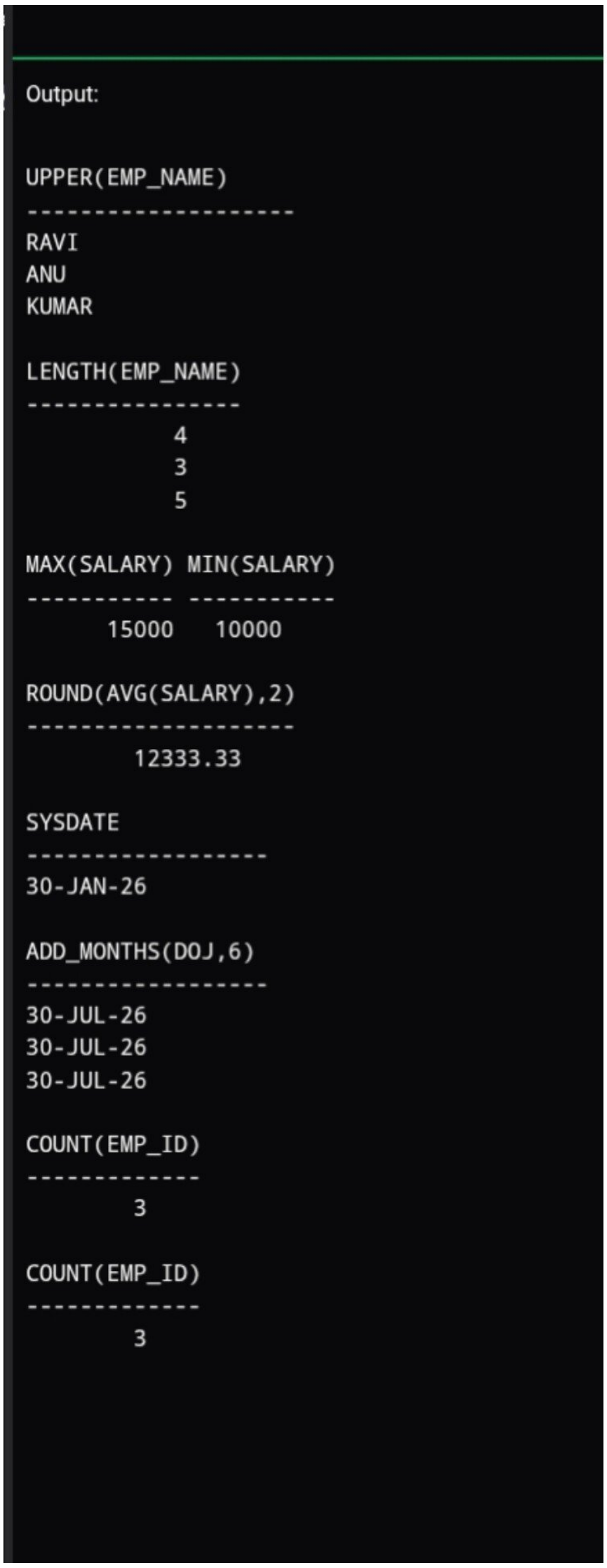


```
    EMP_ID EMP_NAME                    AGE     SALARY    DEPT_ID
---------- -------------------- ---------- ---------- ----------
       101 Ravi                         22      10000          1
       102 Anu                          25      12000          2

    EMP_ID EMP_NAME                    AGE     SALARY    DEPT_ID
---------- -------------------- ---------- ---------- ----------
       101 Ravi                         22      10000          1
       102 Anu                          25      12000          2
```

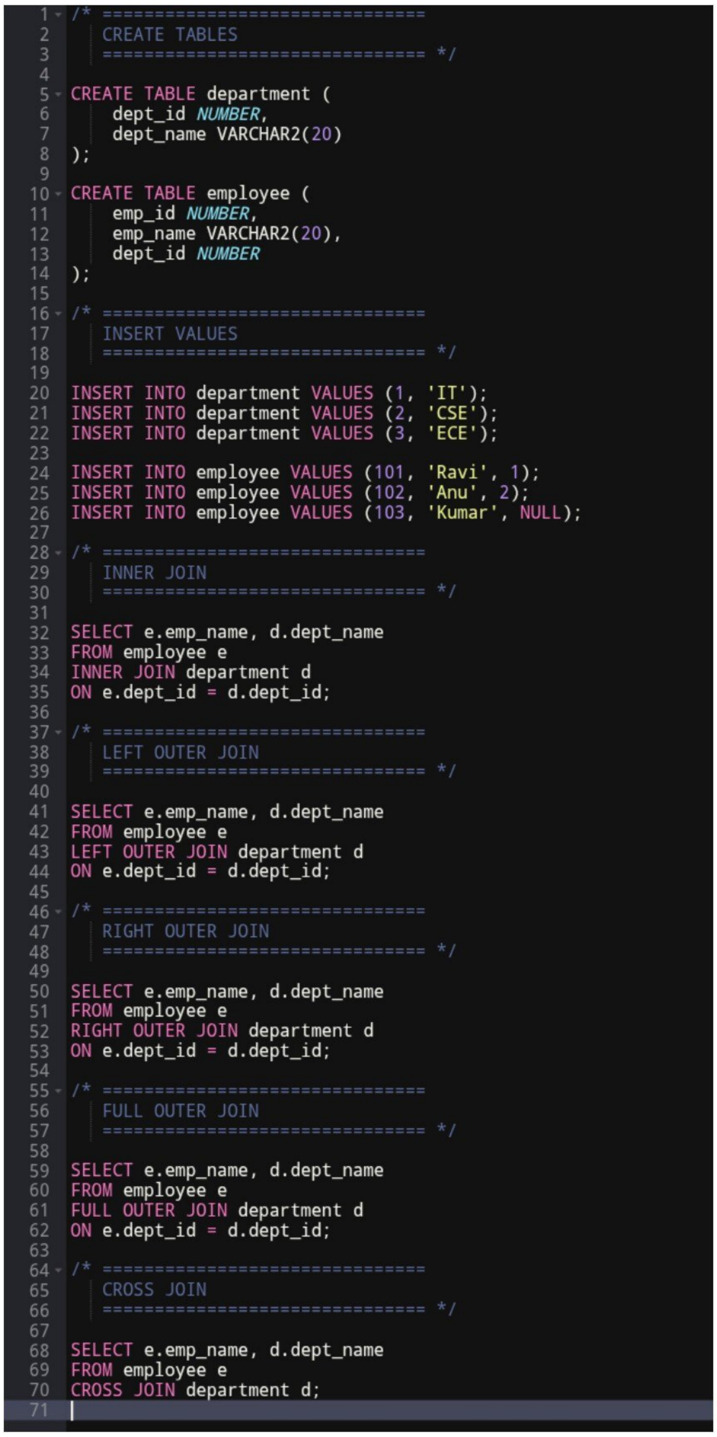


queries.sql 44c5d4q6x

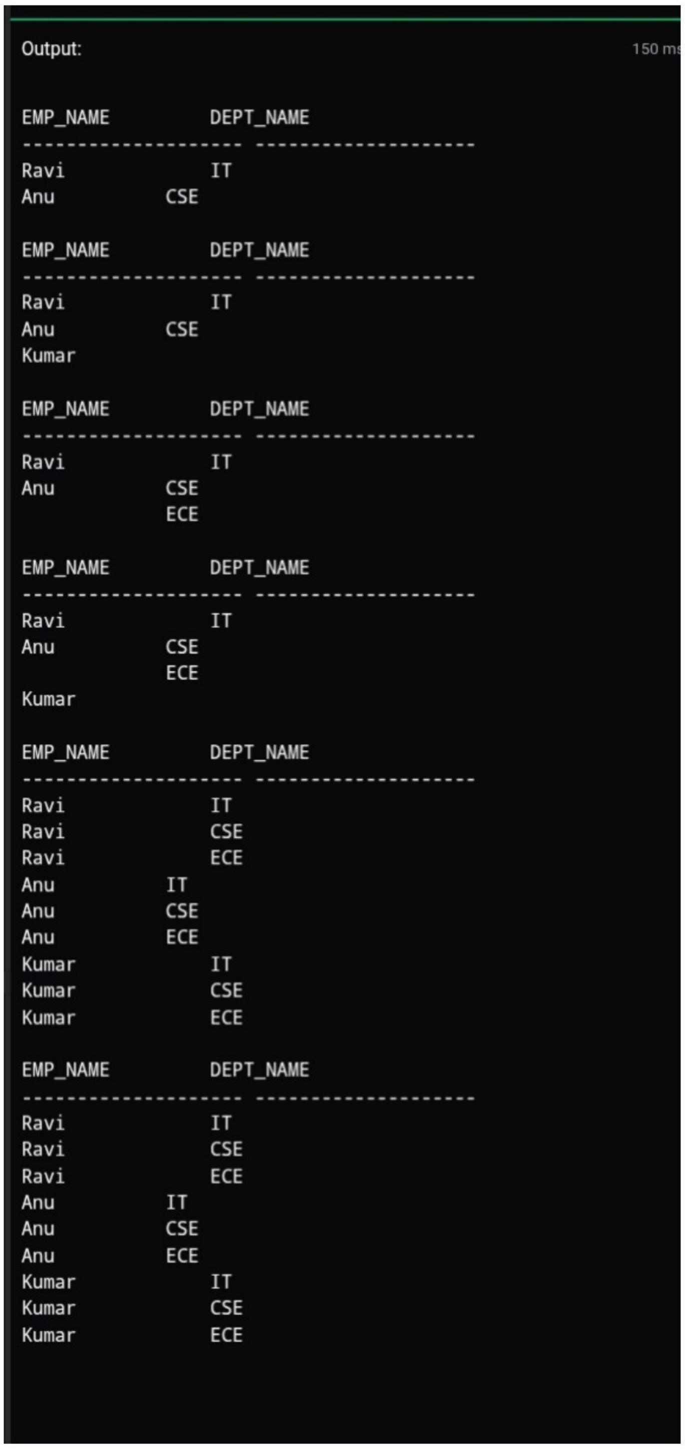
```sql
/* Create table */
CREATE TABLE employee (
    emp_id NUMBER,
    emp_name VARCHAR2(20),
    salary NUMBER,
    doj DATE
);

/* Insert values */
INSERT INTO employee VALUES (1, 'ravi', 12000, SYSDATE);
INSERT INTO employee VALUES (2, 'anu', 15000, SYSDATE);
INSERT INTO employee VALUES (3, 'kumar', 10000, SYSDATE);

/* STRING FUNCTIONS */
SELECT UPPER(emp_name) FROM employee;
SELECT LENGTH(emp_name) FROM employee;

/* NUMBER FUNCTIONS */
SELECT MAX(salary), MIN(salary) FROM employee;
SELECT ROUND(AVG(salary), 2) FROM employee;

/* DATE FUNCTIONS */
SELECT SYSDATE FROM dual;
SELECT ADD_MONTHS(doj, 6) FROM employee;

/* AGGREGATE FUNCTION */
SELECT COUNT(emp_id) FROM employee;
```



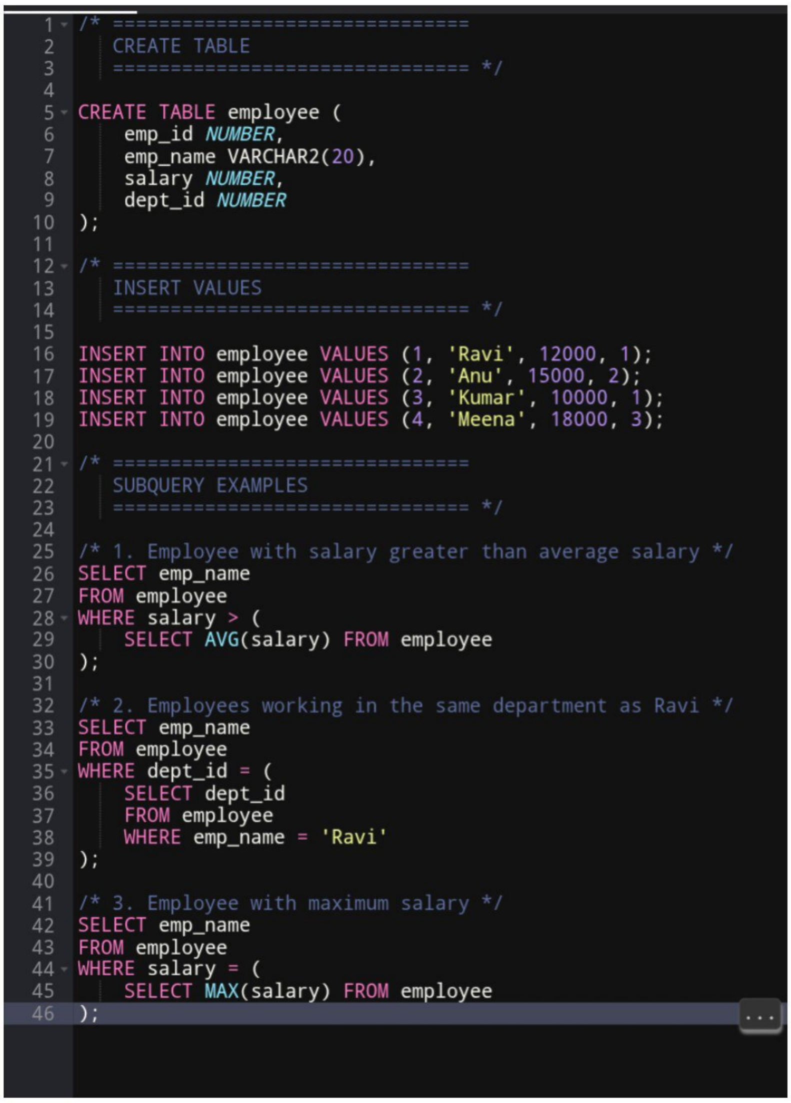
```
Output:

UPPER(EMP_NAME)
--------------------
RAVI
ANU
KUMAR

LENGTH(EMP_NAME)
----------------
               4
               3
               5

MAX(SALARY) MIN(SALARY)
----------- -----------
      15000       10000

ROUND(AVG(SALARY),2)
--------------------
            12333.33

SYSDATE
------------------
30-JAN-26

ADD_MONTHS(DOJ,6)
------------------
30-JUL-26
30-JUL-26
30-JUL-26

COUNT(EMP_ID)
-------------
            3

COUNT(EMP_ID)
-------------
            3
```



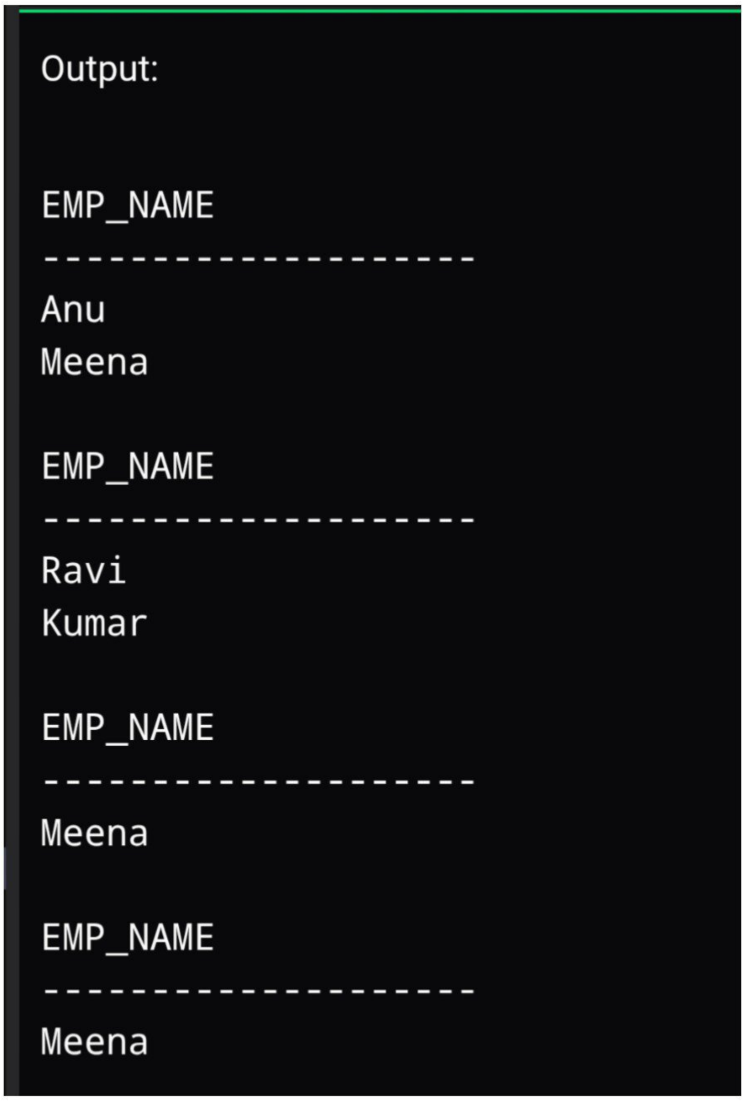
```sql
/* ==============================
   CREATE TABLES
   ============================== */

CREATE TABLE department (
    dept_id NUMBER,
    dept_name VARCHAR2(20)
);

CREATE TABLE employee (
    emp_id NUMBER,
    emp_name VARCHAR2(20),
    dept_id NUMBER
);

/* ==============================
   INSERT VALUES
   ============================== */

INSERT INTO department VALUES (1, 'IT');
INSERT INTO department VALUES (2, 'CSE');
INSERT INTO department VALUES (3, 'ECE');

INSERT INTO employee VALUES (101, 'Ravi', 1);
INSERT INTO employee VALUES (102, 'Anu', 2);
INSERT INTO employee VALUES (103, 'Kumar', NULL);

/* ==============================
   INNER JOIN
   ============================== */

SELECT e.emp_name, d.dept_name
FROM employee e
INNER JOIN department d
ON e.dept_id = d.dept_id;

/* ==============================
   LEFT OUTER JOIN
   ============================== */

SELECT e.emp_name, d.dept_name
FROM employee e
LEFT OUTER JOIN department d
ON e.dept_id = d.dept_id;

/* ==============================
   RIGHT OUTER JOIN
   ============================== */

SELECT e.emp_name, d.dept_name
FROM employee e
RIGHT OUTER JOIN department d
ON e.dept_id = d.dept_id;

/* ==============================
   FULL OUTER JOIN
   ============================== */

SELECT e.emp_name, d.dept_name
FROM employee e
FULL OUTER JOIN department d
ON e.dept_id = d.dept_id;

/* ==============================
   CROSS JOIN
   ============================== */

SELECT e.emp_name, d.dept_name
FROM employee e
CROSS JOIN department d;
```



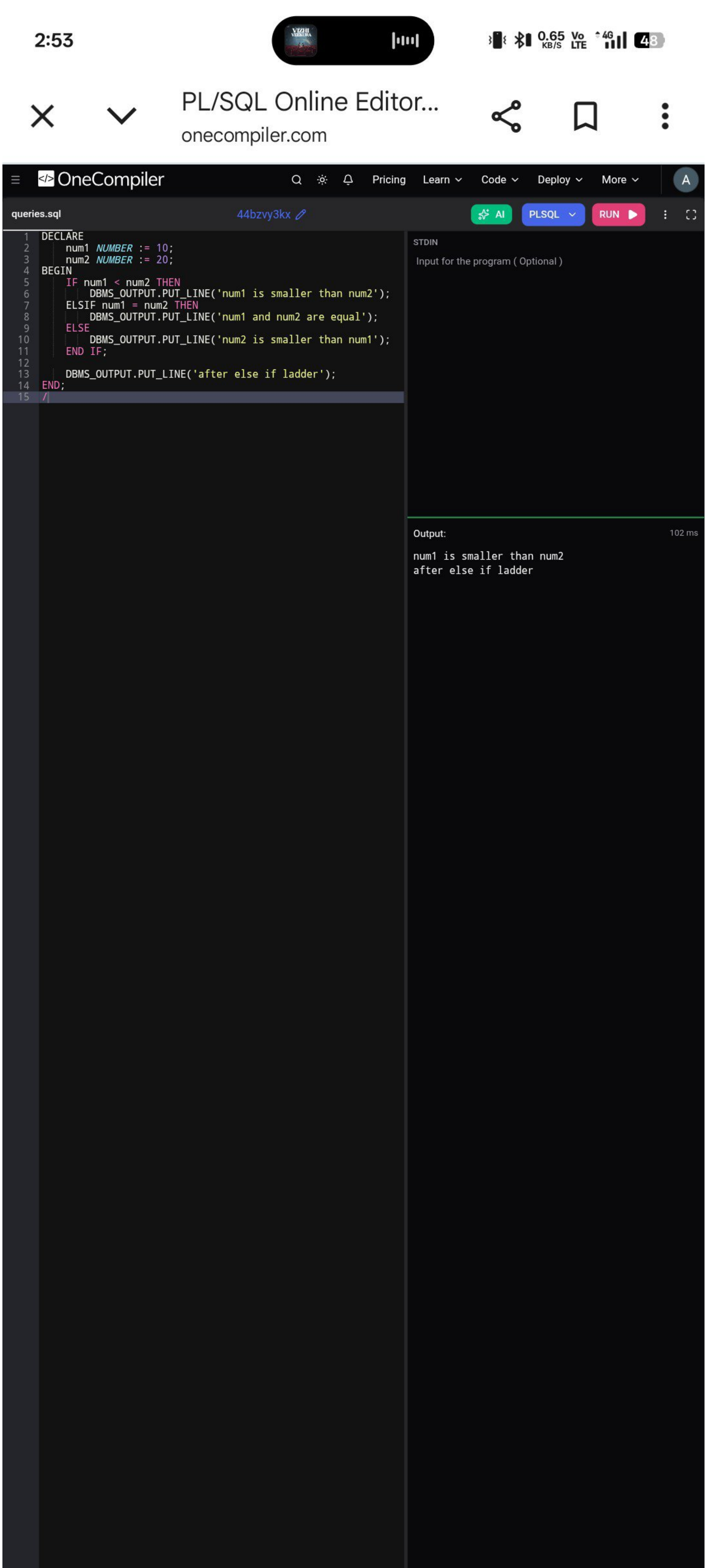
```
Output:                                                          150 m

EMP_NAME             DEPT_NAME
-------------------- --------------------
Ravi                 IT
Anu             CSE

EMP_NAME             DEPT_NAME
-------------------- --------------------
Ravi                 IT
Anu             CSE
Kumar

EMP_NAME             DEPT_NAME
-------------------- --------------------
Ravi                 IT
Anu             CSE
                ECE

EMP_NAME             DEPT_NAME
-------------------- --------------------
Ravi                 IT
Anu             CSE
                ECE
Kumar

EMP_NAME             DEPT_NAME
-------------------- --------------------
Ravi                 IT
Ravi                 CSE
Ravi                 ECE
Anu             IT
Anu             CSE
Anu             ECE
Kumar                IT
Kumar                CSE
Kumar                ECE

EMP_NAME             DEPT_NAME
-------------------- --------------------
Ravi                 IT
Ravi                 CSE
Ravi                 ECE
Anu             IT
Anu             CSE
Anu             ECE
Kumar                IT
Kumar                CSE
Kumar                ECE
```



```
/* ================================
   CREATE TABLE
   ================================ */

CREATE TABLE employee (
    emp_id NUMBER,
    emp_name VARCHAR2(20),
    salary NUMBER,
    dept_id NUMBER
);

/* ================================
   INSERT VALUES
   ================================ */

INSERT INTO employee VALUES (1, 'Ravi', 12000, 1);
INSERT INTO employee VALUES (2, 'Anu', 15000, 2);
INSERT INTO employee VALUES (3, 'Kumar', 10000, 1);
INSERT INTO employee VALUES (4, 'Meena', 18000, 3);

/* ================================
   SUBQUERY EXAMPLES
   ================================ */

/* 1. Employee with salary greater than average salary */
SELECT emp_name
FROM employee
WHERE salary > (
    SELECT AVG(salary) FROM employee
);

/* 2. Employees working in the same department as Ravi */
SELECT emp_name
FROM employee
WHERE dept_id = (
    SELECT dept_id
    FROM employee
    WHERE emp_name = 'Ravi'
);

/* 3. Employee with maximum salary */
SELECT emp_name
FROM employee
WHERE salary = (
    SELECT MAX(salary) FROM employee
);
```

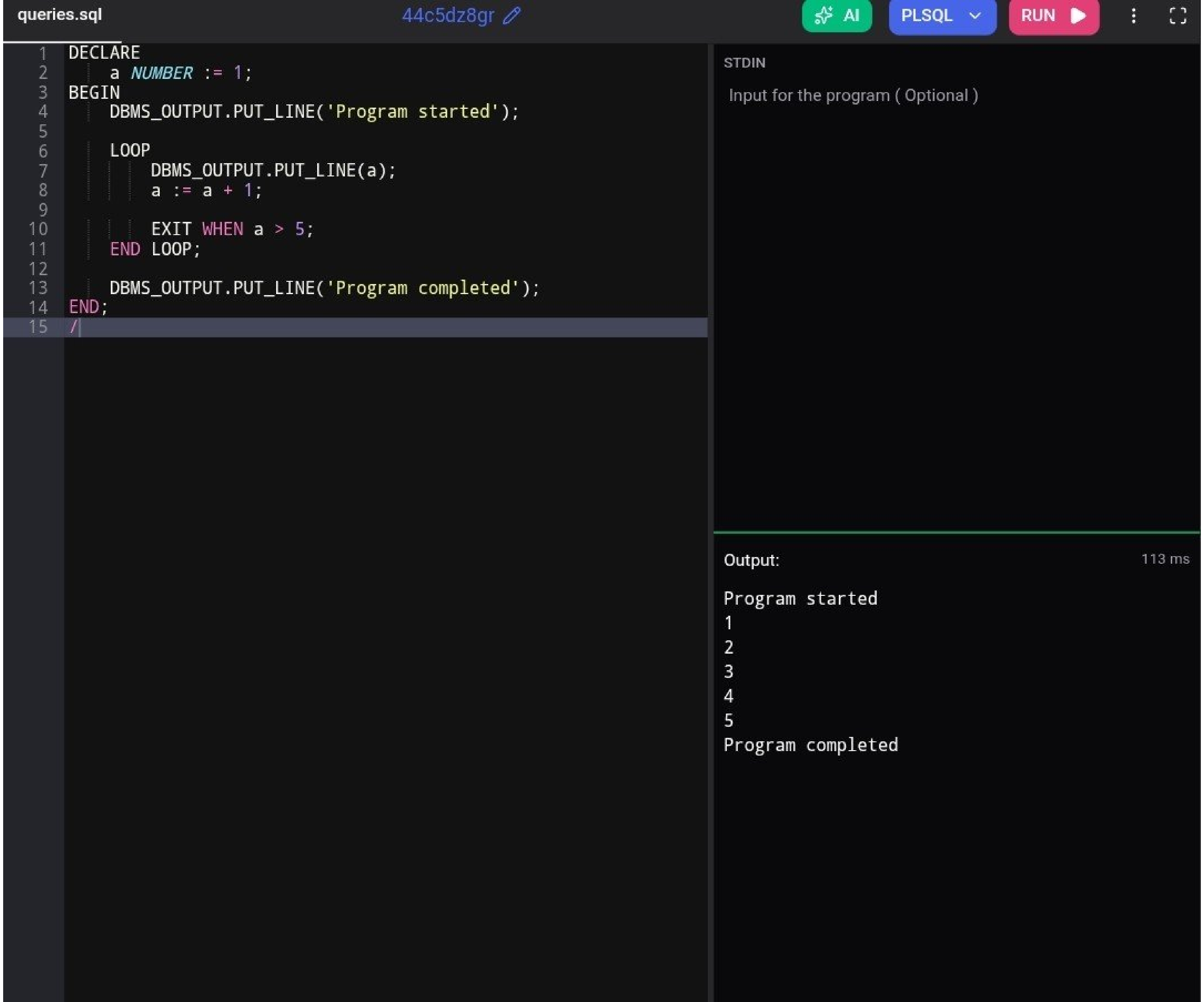


```
Output:

EMP_NAME
--------------------
Anu
Meena

EMP_NAME
--------------------
Ravi
Kumar

EMP_NAME
--------------------
Meena

EMP_NAME
--------------------
Meena
```



```
DECLARE
    num1 NUMBER := 10;
    num2 NUMBER := 20;
BEGIN
    IF num1 < num2 THEN
        DBMS_OUTPUT.PUT_LINE('num1 is smaller than num2');
    ELSIF num1 = num2 THEN
        DBMS_OUTPUT.PUT_LINE('num1 and num2 are equal');
    ELSE
        DBMS_OUTPUT.PUT_LINE('num2 is smaller than num1');
    END IF;

    DBMS_OUTPUT.PUT_LINE('after else if ladder');
END;
/
```

```
num1 is smaller than num2
after else if ladder
```



```sql
DECLARE
    a NUMBER := 1;
BEGIN
    DBMS_OUTPUT.PUT_LINE('Program started');

    LOOP
        DBMS_OUTPUT.PUT_LINE(a);
        a := a + 1;

        EXIT WHEN a > 5;
    END LOOP;

    DBMS_OUTPUT.PUT_LINE('Program completed');
END;
/
```

```
Program started
1
2
3
4
5
Program completed
```