



# DBMS Quiz 2



COLLEGE OF ENGINEERING  
(AUTONOMOUS)

**Name:** Bongu Hari Hara Charan

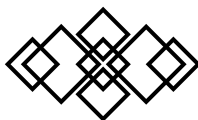
**Roll no:** 323103310032

**Branch:** CSE-1

**Semistor:** 3

**Professor:** Lakshmi Aparna Sreepada

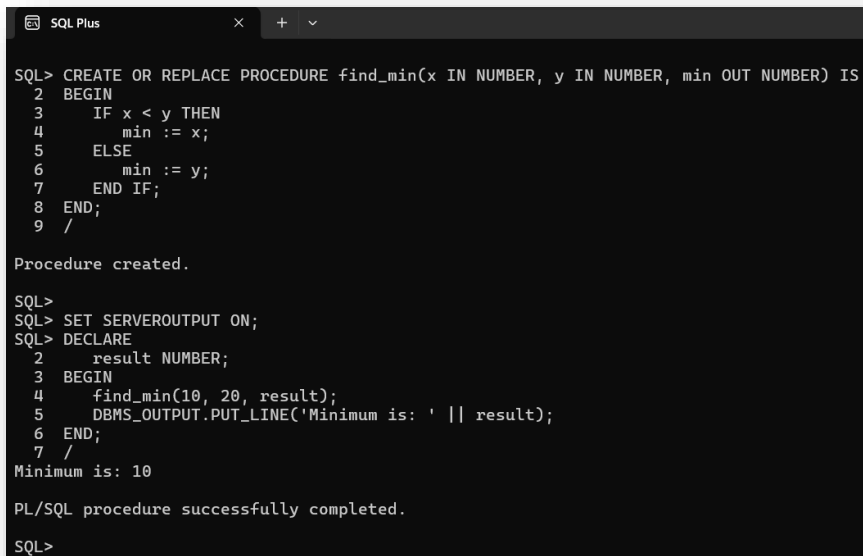
**Links:**



## 1. Find the Minimum of Two Numbers Using Procedures

Ans.

```
CREATE OR REPLACE PROCEDURE find_min(x IN NUMBER, y IN NUMBER, min OUT
NUMBER) IS
BEGIN
    IF x < y THEN
        min := x;
    ELSE
        min := y;
    END IF;
END;
/
-- Call the procedure
DECLARE
    result NUMBER;
BEGIN
    find_min(10, 20, result);
    DBMS_OUTPUT.PUT_LINE('Minimum is: ' || result);
END;
/
```



```
SQL Plus
SQL> CREATE OR REPLACE PROCEDURE find_min(x IN NUMBER, y IN NUMBER, min OUT NUMBER) IS
2 BEGIN
3 IF x < y THEN
4 min := x;
5 ELSE
6 min := y;
7 END IF;
8 END;
9 /

Procedure created.

SQL>
SQL> SET SERVEROUTPUT ON;
SQL> DECLARE
2 result NUMBER;
3 BEGIN
4 find_min(10, 20, result);
5 DBMS_OUTPUT.PUT_LINE('Minimum is: ' || result);
6 END;
7 /
Minimum is: 10

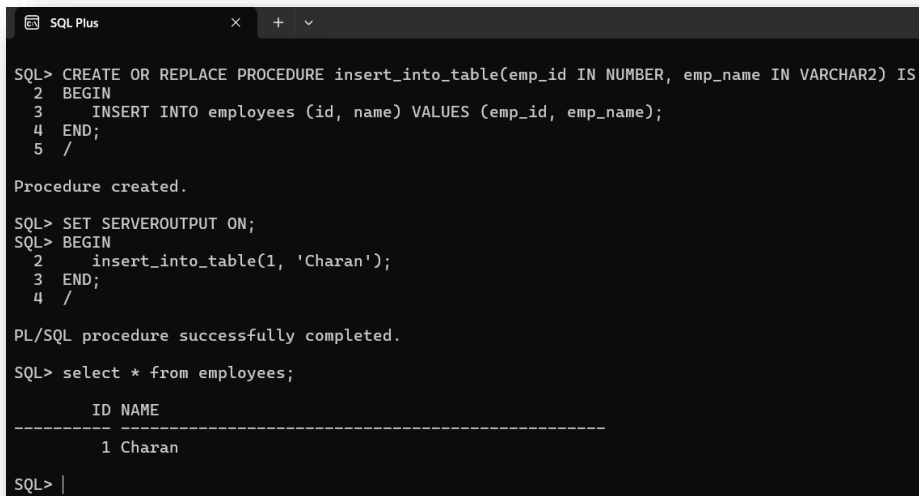
PL/SQL procedure successfully completed.

SQL>
```

## 2. Insert Values into a Table Using Procedures

**Ans.**

```
CREATE OR REPLACE PROCEDURE insert_into_table(emp_id IN NUMBER, emp_name IN
VARCHAR2) IS
BEGIN
    INSERT INTO employees (id, name) VALUES (emp_id, emp_name);
END;
/
SET SERVEROUTPUT ON;
BEGIN
    insert_into_table(1, 'Charan');
END;
/
```



```
SQL Plus
SQL> CREATE OR REPLACE PROCEDURE insert_into_table(emp_id IN NUMBER, emp_name IN VARCHAR2) IS
2  BEGIN
3      INSERT INTO employees (id, name) VALUES (emp_id, emp_name);
4  END;
5  /

Procedure created.

SQL> SET SERVEROUTPUT ON;
SQL> BEGIN
2      insert_into_table(1, 'Charan');
3  END;
4  /

PL/SQL procedure successfully completed.

SQL> select * from employees;

      ID NAME
-----
1 Charan

SQL> |
```

### 3. Exception Handling: Zero Error

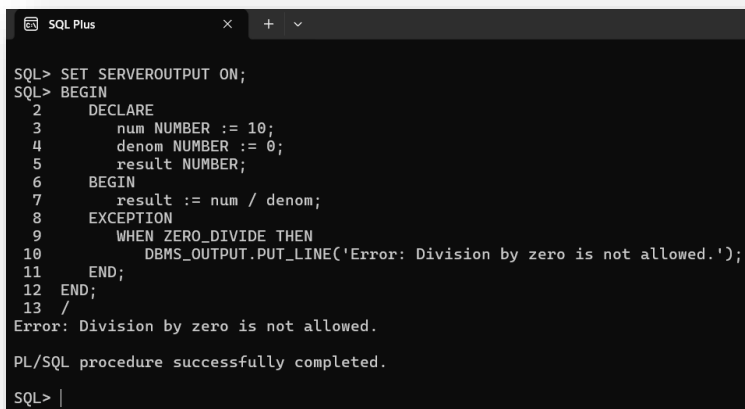
Ans.

```
BEGIN
  DECLARE
    num NUMBER := 10;
    denom NUMBER := 0;
    result NUMBER;
  BEGIN
    result := num / denom;
  EXCEPTION
    WHEN ZERO_DIVIDE THEN
      DBMS_OUTPUT.PUT_LINE('Error: Division by zero is not allowed.');
```

END;

END;

/



```
SQL Plus
SQL> SET SERVEROUTPUT ON;
SQL> BEGIN
2  DECLARE
3    num NUMBER := 10;
4    denom NUMBER := 0;
5    result NUMBER;
6  BEGIN
7    result := num / denom;
8  EXCEPTION
9    WHEN ZERO_DIVIDE THEN
10     DBMS_OUTPUT.PUT_LINE('Error: Division by zero is not allowed.');
```

Error: Division by zero is not allowed.

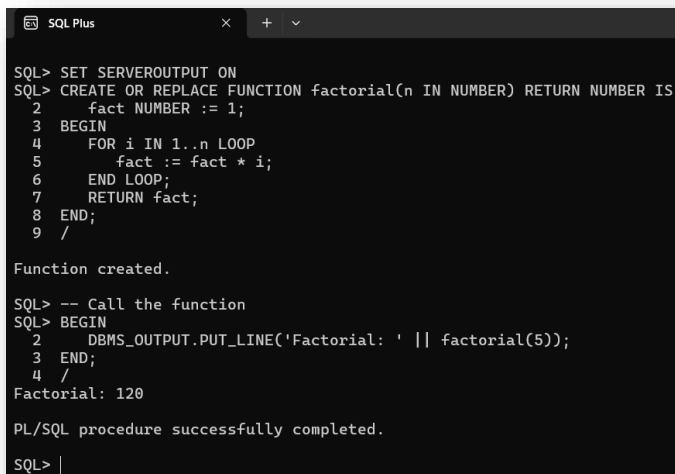
PL/SQL procedure successfully completed.

SQL> |

## 4. Find Factorial Using Functions

**Ans.**

```
CREATE OR REPLACE FUNCTION factorial(n IN NUMBER) RETURN NUMBER IS
    fact NUMBER := 1;
BEGIN
    FOR i IN 1..n LOOP
        fact := fact * i;
    END LOOP;
    RETURN fact;
END;
/
-- Call the function
BEGIN
    DBMS_OUTPUT.PUT_LINE('Factorial: ' || factorial(5));
END;
/
```



```
SQL Plus
SQL> SET SERVEROUTPUT ON
SQL> CREATE OR REPLACE FUNCTION factorial(n IN NUMBER) RETURN NUMBER IS
2   fact NUMBER := 1;
3   BEGIN
4       FOR i IN 1..n LOOP
5           fact := fact * i;
6       END LOOP;
7       RETURN fact;
8   END;
9   /

Function created.

SQL> -- Call the function
SQL> BEGIN
2   DBMS_OUTPUT.PUT_LINE('Factorial: ' || factorial(5));
3   END;
4   /
Factorial: 120

PL/SQL procedure successfully completed.

SQL> |
```

## 5. Exception Handling: No Rows

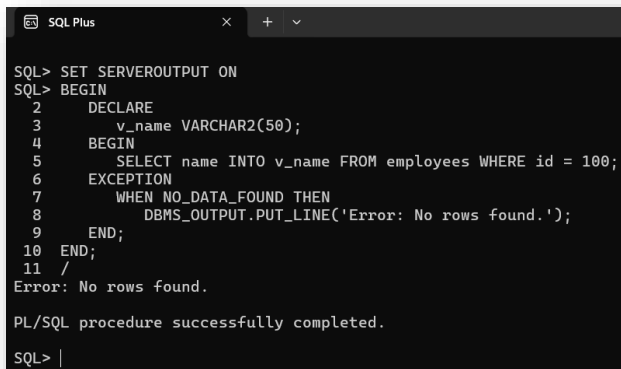
Ans.

```
BEGIN
  DECLARE
    v_name VARCHAR2(50);
  BEGIN
    SELECT name INTO v_name FROM employees WHERE id = 100;
  EXCEPTION
    WHEN NO_DATA_FOUND THEN
      DBMS_OUTPUT.PUT_LINE('Error: No rows found.');
```

END;

END;

/



```
SQL Plus
SQL> SET SERVEROUTPUT ON
SQL> BEGIN
2   DECLARE
3     v_name VARCHAR2(50);
4   BEGIN
5     SELECT name INTO v_name FROM employees WHERE id = 100;
6   EXCEPTION
7     WHEN NO_DATA_FOUND THEN
8       DBMS_OUTPUT.PUT_LINE('Error: No rows found.');
```

9 END;

10 END;

11 /

Error: No rows found.

PL/SQL procedure successfully completed.

SQL> |