

Demo Day 11 😊

SET SERVEROUTPUT ON;

----- Records -----

-- Store multi value inside one variable

-- Record, Collection

DECLARE

emp EMPLOYEE%ROWTYPE; -- Multi-values -- record -- employee

--emp_id NUMBER:=&employee_id; -- input from the user --100

BEGIN

SELECT *

INTO emp

FROM employee

WHERE FULL_NAME = 'Dana Kanaan'; -- id = 100;

/*

emp {

full_name: 'Aseel Al-Serhan' ,

email: 'muadi@gmail.com'

}

*/

/*

emp.full_name := 'Raghad';

emp.email := 'raghad@gmail.com';

emp.salary := 2000;

*/

dbms_output.Put_line ('Employee Name: '

|| emp.full_name);

```

dbms_output.Put_line ('Employee Email: '
                      || emp.email);
EXCEPTION
    WHEN no_data_found THEN
        dbms_output.Put_line ('There is no employee with the ID ');
END;

```

----- A cursor is a pointer to this context area. -----

----- 1. Implicit Cursor -----

-- The uses of implicit cursor without using any attribute.

```

DECLARE

```

```

    emp_full_name VARCHAR2(35);
    emp_id NUMBER:=&employee_id;

```

```

BEGIN

```

```

    SELECT full_name
    INTO  emp_full_name
    FROM  employee
    WHERE id = emp_id;

```

```

    dbms_output.Put_line ('Employee Name: '
                      || emp_full_name);

```

```

EXCEPTION

```

```

    WHEN no_data_found THEN
        dbms_output.Put_line ('There is no employee with the ID ' || to_char(emp_id));

```

```

END;

```

-- The uses of SQL%FOUND and SQL%ROWCOUNT attribute

-- to determine if a UPDATE statement affected any rows.

```

BEGIN

    UPDATE EMPLOYEE

    SET SALARY = 1200

    WHERE FULL_NAME = 'Basheer';

    IF(SQL%FOUND) THEN

        dbms_output.Put_line ('This query affected ' || SQL%ROWCOUNT || ' Rows');

    END IF;

    IF(SQL%NOTFOUND) THEN

        dbms_output.Put_line ('No matching rows');

    END IF;

END;

```

----- 2. Explicit Cursor -----

```

/*

    1- Declare the cursor.

    2- Open the cursor.

    LOOP

        3- Fetch the cursor for retrieving the data.

    END LOOP;

    4- Close the cursor.

*/

```

```

DECLARE

    emp_name EMPLOYEE.FULL_NAME%TYPE;

    emp_dept EMPLOYEE.DEPT_ID%TYPE;

```

```

emp_salary EMPLOYEE.SALARY%TYPE;

-- 1- Declare the cursor.

CURSOR c_emp IS SELECT FULL_NAME, DEPT_ID ,SALARY FROM EMPLOYEE;

BEGIN

-- 2- Open the cursor.

IF NOT c_emp%ISOPEN THEN

    OPEN c_emp;

END IF;

-- 3- Fetch the cursor for retrieving the data.

LOOP

    FETCH c_emp INTO emp_name, emp_dept, emp_salary;

    EXIT WHEN c_emp%NOTFOUND;

    dbms_output.Put_line ('Employee Name ' || emp_name ||

        ' Employee Dept. ' || emp_dept || ' Employee Salary ' || emp_salary);

END LOOP;

dbms_output.Put_line (c_emp%ROWCOUNT);

-- 4- Close the cursor.

CLOSE c_emp;

END;

```

```

CREATE OR REPLACE PROCEDURE MAX_NUM(num1 IN NUMBER, num2 IN NUMBER)

IS

BEGIN

-- PROCEDURE BODY

IF num1 > num2 THEN

    DBMS_OUTPUT.PUT_LINE(num1);

ELSIF num1 = num2 THEN

    DBMS_OUTPUT.PUT_LINE('num1 equal num2');

ELSE

```

```
        DBMS_OUTPUT.PUT_LINE(num2);  
    END IF;  
END;
```

```
DROP PROCEDURE MAX_NUM;
```

```
EXEC MAX_NUM(1,5);
```