

Database Programming with PL/SQL

7-1: Handling Exceptions

Practice Activities

Vocabulary

Identify the vocabulary word for each definition below:

EXCEPTION HANDLER	Code that defines the recovery actions to be performed when execution-time errors occur.
EXCEPTION	Occurs when an error is discovered during the execution of a program that disrupts the normal operation of the program.

Try It / Solve It

1. What happens when Oracle encounters a runtime problem while executing a PL/SQL block?

Muestra el error en la consola y el código se deja de ejecutar.

2. What do you need to add to your PL/SQL block to address these problems?

EXCEPTION HANDLER

3. List three advantages of handling exceptions within a PL/SQL block.

Se tiene un mayor control del código.

Se sabe la razón por la que el código dejó de ejecutarse.

4. Run this PL/SQL code and then answer the questions that follow.

```
DECLARE
  v_jobid employees.job_id%TYPE;
BEGIN
  SELECT job_id INTO v_jobid
  FROM employees
  WHERE department_id = 80;
END;
```

A. What happens when you run the block? In your own words, explain what you can do to fix this problem.

La consulta trata de almacenar más de un valor en una variable.

Se puede agregar una exception.

B. Modify the code to fix the problem. Use a TOO_MANY_ROWS exception handler.

```
DECLARE
v_jobid employees.job_id%TYPE;
BEGIN
SELECT job_id INTO v_jobid
FROM employees
WHERE department_id = 80;
EXCEPTION
when too_many_rows THEN
dbms_output.put_line('La consulta devuelve mas de una fila');
```

END;

C. Run your modified code. What happens this time?

Ahora el código funciona

5. Run the following PL/SQL block, which tries to insert a new row (with department_id = 50) into the departments table. What happens and why?

```
BEGIN
  INSERT INTO departments (department_id, department_name, manager_id,
                           location_id)
    VALUES (50, 'A new department', 100, 1500);
  DBMS_OUTPUT.PUT_LINE('The new department was inserted');
EXCEPTION
  WHEN OTHERS THEN
    DBMS_OUTPUT.PUT_LINE ('An exception has occurred.');
```

END;

La fila no se inserta porque ya existe un departamento con el id 50.

6. Enter the following code to create a copy of the employees table for this and the next question.

```
CREATE TABLE emp_temp AS SELECT * FROM employees;
```

In the new emp_temp table, delete all but one of the employees in department 10.

```
SELECT * FROM emp_temp WHERE department_id = 10;
```

```
DELETE FROM emp_temp WHERE employee_id = ...; (repeat as necessary)
```

Enter the following PL/SQL block, which tries to SELECT all the employees in a specific department. Run it three times, using department_ids 10, 20, and 30. What happens and why?

Marca una excepción, porque se trata de almacenar mas de un valor en las variables v_employee_id y last_name.

```

DECLARE
  v_employee_id employees.employee_id%TYPE;
  v_last_name    employees.last_name%TYPE;
BEGIN
  SELECT employee_id, last_name INTO v_employee_id, v_last_name
    FROM employees
   WHERE department_id = 10; -- run with values 10, 20, and 30
  DBMS_OUTPUT.PUT_LINE('The SELECT was successful');
EXCEPTION
  WHEN OTHERS THEN
    DBMS_OUTPUT.PUT_LINE('An exception has occurred');
END;

```

7. Modify your code from question 6 to add two more exception handlers to trap the possible exceptions individually. Use `NO_DATA_FOUND` and `TOO_MANY_ROWS`. Re-run the block three times, using 10, 20, and 30 as before. Observe the message displayed in each case.

```

DECLARE
  v_employee_id employees.employee_id%TYPE;
  v_last_name    employees.last_name%TYPE;
BEGIN
  SELECT employee_id, last_name INTO v_employee_id, v_last_name
    FROM employees
   WHERE department_id = 10; -- run with values 10, 20, and 30
  DBMS_OUTPUT.PUT_LINE('The SELECT was successful');
EXCEPTION
  WHEN NO_DATA_FOUND THEN
    DBMS_OUTPUT.PUT_LINE('No se encontraron datos');
  WHEN TOO_MANY_ROWS THEN
    DBMS_OUTPUT.PUT_LINE('Muchas filas');
  WHEN OTHERS THEN
    DBMS_OUTPUT.PUT_LINE('An exception has occurred');
END;

```

When finished, remember to delete the emp_temp table.

```

DROP TABLE emp_temp;

```

8. List three guidelines for trapping exceptions.

- **NO_DATA_FOUND**
- **TOO_MANY_ROWS**
- **OTHERS**

9. Enter and run the following PL/SQL block. Explain the output. Note: the WHEN OTHERS handler successfully handles any type of exception which occurs.

```
DECLARE
    v_number    NUMBER(2);
BEGIN
    v_number := 9999;
EXCEPTION
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('An exception has occurred');
END;
```

Marca una excepción porque el tipo de dato de la variable solo tiene como extensión 2 cifras y el valor que se le esta asignando es de 4 cifras.

10. Modify the block in question 9 to omit the exception handler, then re-run the block. Explain the output.

ORA-06502: PL/SQL: numeric or value error: number precision too large

El error no tiene nombre y por eso lo marca con números.

11. Enter and run the following code and explain the output.

```
DECLARE
  v_number  NUMBER(4); BEGIN
  v_number := 1234;
DECLARE
  v_number  NUMBER(4);
BEGIN      v_number :=
5678;
  v_number := 'A character string';
  END;
EXCEPTION
  WHEN OTHERS THEN
    DBMS_OUTPUT.PUT_LINE('An exception has occurred');
    DBMS_OUTPUT.PUT_LINE('The number is: ' || v_number);
  END;
```

Se le asignan datos que no corresponden a el tipo de dato que es la variable.