



BASES DE DATOS II

MANEJO DE EXCEPCIONES

MSc. Jimena Adriana Timaná Peña

Manejo de Excepciones

En PL/SQL un error en la ejecución es llamado **excepción**.

¿Qué es una excepción?

- Es un ***identificador*** en PL / SQL, generado durante la ejecución de un bloque.
- Cuando ocurre un error se lanza una excepción y se transfiere el control a la sección del bloque encargada de su manejo.

Manejo de Excepciones

- Si la excepción **se controla** en el bloque, entonces no se propaga en el ambiente o en el bloque que lo contiene. El bloque PL / SQL termina con éxito.
- Sin embargo, si la excepción se lanza y **no existe el correspondiente manejador de excepción** entonces el bloque termina con un error y la excepción se propaga al ambiente de llamado.
- Por lo tanto, lo ideal es **controlar las excepciones** para permitir que el programa siga.

Manejo de Excepciones

Tipos de excepciones

Implícitamente generadas

- Predefined Oracle Server (*Predefinidas del Servidor Oracle*)
- Non-predefined Oracle Server (*No predefinidas del Servidor Oracle*)

Explícitamente generadas

- User-defined (*Definidas por el Usuario*)

Excepción	Descripción	Instrucciones para el manejo
Predefined Oracle Server	Uno de los aproximadamente 20 errores que ocurren con mayor frecuencia en el código PL / SQL	No se deben declarar. El servidor Oracle los genera implícitamente.
Non-predefined Oracle Server	Cualquier otro error estándar del servidor Oracle	Declárelos dentro de la sección declarativa y permita que Oracle Server los genere implícitamente.
User-defined	Una condición que el desarrollador determina es anormal.	Declárelos dentro de la sección declarativa y génere los explícitamente.

Manejo de Excepciones

Implícitamente generadas - *Predefined Oracle Server* (Predefinidas del Servidor Oracle)

- cuando un error ocurre, Oracle **genera automáticamente** una excepción asociada a dicho error.
- Por ejemplo, el error **ORA-01403** ocurre cuando no se recuperan filas de la base de datos en una instrucción SELECT. Así, PL/SQL genera implícitamente la excepción *NO_DATA_FOUND*.

Manejo de Excepciones

Ejemplo 1: Primero, a través de un bloque anónimo consultemos el nombre de un empleado que sepamos que exista, basado en su identificación.

```
SET VERIFY OFF
```

```
SET ECHO OFF
```

```
SET SERVEROUTPUT ON
```

```
declare
```

```
    v_name emp.ename%TYPE;
```

```
begin
```

```
    select ename
```

```
    into v_name
```

```
    from emp
```

```
    where empno = 7839;
```

```
    DBMS_OUTPUT.PUT_LINE('El nombre es: ' || v_name);
```

```
end;
```

```
bloque anónimo terminado
El nombre es: KING
```

Manejo de Excepciones

Ejemplo 1: Ahora con un empleado que **no exista**:

```
SET VERIFY OFF  
SET ECHO OFF  
SET SERVEROUTPUT ON
```

```
declare  
    v_name emp.ename%TYPE;  
begin  
    select ename  
    into v_name  
    from emp  
    where empno = 7777;  
    DBMS_OUTPUT.PUT_LINE('El nombre es: ' || v_name);  
end;
```

Oracle **genera automáticamente**
una excepción asociada a dicho
error



Informe de error:
ORA-01403: no data found

Manejo de Excepciones

Sintaxis para la captura de excepciones:

```
EXCEPTION
  WHEN exception1 [OR exception2 . . .] THEN
    statement1;
    statement2;
    . . .
  [WHEN exception3 [OR exception4 . . .] THEN
    statement1;
    statement2;
    . . .]
  [WHEN OTHERS THEN
    statement1;
    statement2;
    . . .]
```

<i>exception1(2...n)</i>	Es el nombre estándar de una excepción predefinida o el nombre de una excepción definida por el usuario declarada dentro de la sección declarativa
<i>statement1(2...n)</i>	Es una o más sentencias PL / SQL o SQL.
<i>OTHERS</i>	es una cláusula opcional de manejo de excepciones que atrapa excepciones no especificadas

Algunas de las excepciones predefinidas más comunes por Oracle son:

Exception Name	Oracle Server Error Number	Description
ACCESS_INTO_NULL	ORA-06530	Attempted to assign values to the attributes of an uninitialized object
COLLECTION_IS_NULL	ORA-06531	Attempted to apply collection methods other than EXISTS to an uninitialized nested table or varray
CURSOR_ALREADY_OPEN	ORA-06511	Attempted to open an already open cursor
DUP_VAL_ON_INDEX	ORA-00001	Attempted to insert a duplicate value
INVALID_CURSOR	ORA-01001	Illegal cursor operation occurred
INVALID_NUMBER	ORA-01722	Conversion of character string to number fails
LOGIN_DENIED	ORA-01017	Logging on to Oracle with an invalid username or password
NO_DATA_FOUND	ORA-01403	Single row SELECT returned no data
NOT_LOGGED_ON	ORA-01012	PL/SQL program issues a database call without being connected to Oracle
PROGRAM_ERROR	ORA-06501	PL/SQL has an internal problem
ROWTYPE_MISMATCH	ORA-06504	Host cursor variable and PL/SQL cursor variable involved in an assignment have incompatible return types
STORAGE_ERROR	ORA-06500	PL/SQL ran out of memory or memory is corrupted
SUBSCRIPT_BEYOND_COUNT	ORA-06533	Referenced a nested table or varray element using an index number larger than the number of elements in the collection
SUBSCRIPT_OUTSIDE_LIMIT	ORA-06532	Referenced a nested table or varray element using an index number that is outside the legal range (-1 for example)
TIMEOUT_ON_RESOURCE	ORA-00051	Time-out occurred while Oracle is waiting for a resource
TOO_MANY_ROWS	ORA-01422	Single-row SELECT returned more than one row
VALUE_ERROR	ORA-06502	Arithmetic, conversion, truncation, or size-constraint error occurred
ZERO_DIVIDE	ORA-01476	Attempted to divide by zero

```
BEGIN
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        statement1;
        statement2;
    WHEN TOO_MANY_ROWS THEN
        statement1;
    WHEN OTHERS THEN
        statement1;
        statement2;
        statement3;
END;
```

Nombres de más excepciones:

https://docs.oracle.com/cd/B19306_01/appdev.102/b14261/errors.htm

Manejo de excepciones

- Atributos que pueden aparecer en una cláusula IF y a partir de las cuales podemos lanzar una excepción

Atributo	Tipo	Descripción
%ISOPEN	Booleano	TRUE si el cursor está abierto.
%NOTFOUND	Booleano	TRUE si la recuperación más reciente no devuelve ninguna fila.
%FOUND	Booleano	TRUE si la recuperación más reciente devuelve una fila.
%ROWCOUNT	Númerico	Proporciona el número total de filas devueltas hasta ese momento.

Manejo de Excepciones

Ejemplo: Retomemos el ejemplo del bloque PL/SQL que consultaba el nombre de un empleado basado en su identificación. En este caso la identificación será 777799 (este empleado NO existe). Utilicemos la sección en el bloque para el manejo de excepciones. Muestre un mensaje que diga que “El usuario no existe”.

```
SET SERVEROUTPUT ON
SET VERIFY OFF
SET ECHO OFF
DECLARE
    v_name emp.ename%TYPE;
BEGIN
    select ename
    into v_name
    from emp
    where empno = 777799;
    dbms_output.put_line('El nombre del usuario es:' || v_name);
EXCEPTION
    when NO_DATA_FOUND then
        dbms_output.put_line('El usuario no existe');
END;
```

Manejo de Excepciones

Explícitamente generadas - *User-defined (Definidas por el Usuario)*

PL/SQL permite definir nuestras propias excepciones. Las excepciones PL/SQL definidas por el usuario deben ser:

1. Declaradas en la sección de declaración DECLARE de un bloque PL / SQL. Indique un nombre para la excepción definida por usted.
2. Lanzarlas **explícitamente** con la sentencia RAISE. Use la sentencia RAISE para capturar explícitamente la excepción en la sección ejecutable del bloque.

Manejo de Excepciones

Ejemplo: Analicemos lo que esta ocurriendo en el siguiente bloque:

```
DECLARE
    e_invalid_product  EXCEPTION;
BEGIN
    UPDATE      product
    SET         descrip = '&product_description'
    WHERE       prodid = &product_number;
    IF SQL%NOTFOUND THEN
        RAISE e_invalid_product;
    END IF;
    COMMIT;
EXCEPTION
    WHEN e_invalid_product THEN
        DBMS_OUTPUT.PUT_LINE('Invalid product number.');
```

END;

Manejo de Excepciones

Ejemplo: Analicemos lo que está ocurriendo en el siguiente bloque:

Sintaxis:

nombreExcepción EXCEPTION

Se declara la excepción definida por el usuario dentro de la sección declarativa del bloque

```
DECLARE
  e_invalid_product  EXCEPTION;  1
BEGIN
  UPDATE      product
  SET         descrip = '&product_description'
  WHERE       prodid = &product_number;
  IF SQL%NOTFOUND THEN
    RAISE e_invalid_product;
  END IF;
  COMMIT;
EXCEPTION
  WHEN e_invalid_product THEN
    DBMS_OUTPUT.PUT_LINE('Invalid product number.');
```

END;

Manejo de Excepciones

Ejemplo: Analicemos lo que está ocurriendo en el siguiente bloque:

```
DECLARE
    e_invalid_product  EXCEPTION;
BEGIN
    UPDATE      product
    SET         descrip = '&product_description'
    WHERE       prodid = &product_number;
    IF SQL%NOTFOUND THEN
        RAISE e_invalid_product;
    END IF;
    COMMIT;
EXCEPTION
    WHEN e_invalid_product THEN
        DBMS_OUTPUT.PUT_LINE('Invalid product number.');
```

2 Use la sentencia **RAISE** para lanzar la excepción explícitamente dentro de la sección ejecutable del bloque

```
END;
```


Manejo de Excepciones

Ejemplo: Analicemos lo que está ocurriendo en el siguiente bloque:

```
DECLARE
    e_invalid_product  EXCEPTION;
BEGIN
    UPDATE      product
    SET         descrip = '&product_description'
    WHERE       prodid = &product_number;
    IF SQL%NOTFOUND THEN
        RAISE e_invalid_product;
    END IF;
    COMMIT;
EXCEPTION
    WHEN e_invalid_product THEN
        DBMS_OUTPUT.PUT_LINE('Invalid product number.');
```

END;

Haga referencia a la excepción declarada dentro de la sección en el bloque para el manejo de excepciones **EXCEPTION**

Manejo de Excepciones

Existe un procedimiento que también permite emitir mensajes de error definidos por el usuario.

- En el paquete DBMS_STANDARD se incluye un procedimiento llamado ***RAISE_APPLICATION_ERROR*** que permite *predefinir* excepciones y el cual, devuelve un código de error y el mensaje de error no estándar.
- Con RAISE_APPLICATION_ERROR, puede reportar errores a su aplicación y evitar devolver excepciones no manejadas.

Manejo de Excepciones

Sintaxis:

```
raise_application_error (error_number, message[, {TRUE | FALSE}]);
```

<i>error_number</i>	Número especificado por el usuario para la excepción. Está comprendido entre -20000 y -20999
<i>message</i>	Es un mensaje especificado por el usuario. Es una cadena de caracteres de hasta 2048 bytes de longitud.
<i>TRUE FALSE</i>	Si el parámetro opcional es puesto en TRUE, el error se pone en la pila de las excepciones lanzadas anteriormente. Es útil si se quiere ver la traza de todas las excepciones que se lanzaron. Si es puesto en FALSE (por defecto), el error reemplaza los anteriores.

Manejo de Excepciones

El procedimiento ***RAISE_APPLICATION_ERROR*** puede ser utilizado en dos lugares diferentes:

- Sección ejecutable

```
...  
DELETE FROM emp  
WHERE mgr = v_mgr;  
IF SQL%NOTFOUND THEN  
    RAISE_APPLICATION_ERROR(-20202, 'This is not a valid manager');  
END IF;  
...
```

- Sección de excepciones

```
...  
EXCEPTION  
    WHEN NO_DATA_FOUND THEN  
        RAISE_APPLICATION_ERROR (-20201,  
            'Manager is not a valid employee.');
```

```
END;
```

Manejo de Excepciones

Analice el siguiente bloque PL/SQL:

```
SET SERVEROUTPUT ON
DECLARE
v_nombre VARCHAR(50);
v_prog_id NUMBER := &v;
BEGIN
    IF v_prog_id <= 0 THEN
        RAISE_APPLICATION_ERROR(-20001, 'El id de programa es cero o negativo');
    END IF;
    SELECT nombre INTO v_nombre FROM programa
    WHERE programa_id = v_prog_id;
    DBMS_OUTPUT.PUT_LINE('Nombre programa: ' || v_nombre);
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        DBMS_OUTPUT.PUT_LINE('No existen datos para el id introducido');
        RAISE_APPLICATION_ERROR(-20002, 'No hay datos en la tabla para el id
introducido', true);
END ;
```

Manejo de excepciones

Ejercicio 1:

Cree un bloque anónimo que elimine un empleado específico. Controle cuando el empleado no exista. El tipo de excepción va a ser definida por el usuario. Utilizar RAISE.

Manejo de Excepciones

Implícitamente generadas - Non-predefined Oracle Server (No predefinidas del Servidor Oracle)

- Es posible manejar excepciones que no tienen un nombre predefinido por el servidor Oracle.
- se declaran primero o los puede usar a través del manejador OTHERS. La excepción declarada es lanzada implícitamente.
- En PL/SQL, el PRAGMA EXCEPTION_INIT le dice al compilador que asocie el nombre de una excepción a un número de error estándar del Servidor Oracle. Esto le permitirá hacer referencia a cualquier excepción a través de su nombre y escribir un manejador para ello. En otras palabras, ya puede usar dicho nombre en el bloque de excepciones después del WHEN.

Manejo de Excepciones

Implícitamente generadas - Non-predefined Oracle Server (No predefinidas del Servidor Oracle)

Nota a tener en cuenta:

- **PRAGMA** (también llamado pseudoinstrucciones) es la palabra clave que significa que la sentencia es una directiva de compilación, que no se procesa cuando se ejecuta el bloque PL/SQL. Más bien, dirige al compilador de PL / SQL a interpretar todas las apariciones del nombre de excepción dentro del bloque como el número de error asociado de Oracle Server.

Manejo de Excepciones

Ejemplo:

```
DECLARE
    e_emps_remaining    EXCEPTION;
    PRAGMA EXCEPTION_INIT (
        e_emps_remaining, -2292);
    v_deptno            dept.deptno%TYPE := &p_deptno;
BEGIN
    DELETE FROM dept
    WHERE            deptno = v_deptno;
    COMMIT;
EXCEPTION
    WHEN e_emps_remaining THEN
        DBMS_OUTPUT.PUT_LINE ('Cannot remove dept ' ||
            TO_CHAR(v_deptno) || '. Employees exist. ');
END;
```