

## EXCEPCIONS

Definició	Format																					
<p><i>Les excepcions existeixen a la majoria de llenguatges de programació, serveixen per tractar errors que es produeixen quan s'executa un programa.</i></p> <p>En Oracle, serveixen per definir què s'ha de fer quan alguna de les sentències del codi que s'executa dona error. Quan es produeix un error PL/ SQL provoca una excepció i passa el control a la secció de gestió d'excepcions corresponent a aquell bloc PL/SQL. <i>Un bloc és el contingut que hi ha entre un begin i un end.</i></p> <table border="1"><thead><tr><th>Nombre de la excepció</th><th>Número de error Oracle</th><th>Descripció</th></tr></thead><tbody><tr><td>NO_DATA_FOUND</td><td>ORA-01403</td><td>La sentencia SELECT no devolvió datos.</td></tr><tr><td>TOO_MANY_ROWS</td><td>ORA-01422</td><td>La sentencia SELECT devolvió más de una fila.</td></tr><tr><td>INVALID_CURSOR</td><td>ORA-01001</td><td>Se produjo una operación de cursor ilegal.</td></tr><tr><td>ZERO_DIVIDE</td><td>ORA-01476</td><td>Se intentó dividir entre cero.</td></tr><tr><td>DUP_VAL_ON_INDEX</td><td>ORA-00001</td><td>Se intentó insertar un valor duplicado.</td></tr><tr><td>INVALID_NUMBER</td><td>ORA-01722</td><td>Falla la conversión de una cadena de caracteres a números.</td></tr></tbody></table>	Nombre de la excepció	Número de error Oracle	Descripció	NO_DATA_FOUND	ORA-01403	La sentencia SELECT no devolvió datos.	TOO_MANY_ROWS	ORA-01422	La sentencia SELECT devolvió más de una fila.	INVALID_CURSOR	ORA-01001	Se produjo una operación de cursor ilegal.	ZERO_DIVIDE	ORA-01476	Se intentó dividir entre cero.	DUP_VAL_ON_INDEX	ORA-00001	Se intentó insertar un valor duplicado.	INVALID_NUMBER	ORA-01722	Falla la conversión de una cadena de caracteres a números.	<pre>BEGIN      .....      .....      .....  EXCEPTION      WHEN &lt;nom_excepció&gt; THEN          &lt;instruccions&gt;;          .....      [WHEN OTHERS THEN          &lt;instruccions&gt;;          .....]</pre> <pre>END;</pre>
Nombre de la excepció	Número de error Oracle	Descripció																				
NO_DATA_FOUND	ORA-01403	La sentencia SELECT no devolvió datos.																				
TOO_MANY_ROWS	ORA-01422	La sentencia SELECT devolvió más de una fila.																				
INVALID_CURSOR	ORA-01001	Se produjo una operación de cursor ilegal.																				
ZERO_DIVIDE	ORA-01476	Se intentó dividir entre cero.																				
DUP_VAL_ON_INDEX	ORA-00001	Se intentó insertar un valor duplicado.																				
INVALID_NUMBER	ORA-01722	Falla la conversión de una cadena de caracteres a números.																				

Excepcions predefinides				Exemples
<p>Són aquelles que es disparen <u>automàticament</u> al produir-se certs errors.</p> <p>Les més habituals són: <a href="#">sqlcode function</a></p>				<pre> DECLARE   l_employee_last_name      employees.last_name%TYPE;   l_employee_full_name      VARCHAR2(52);  BEGIN    l_employee_last_name := 'King';    BEGIN      SELECT last_name    ', '    first_name     INTO l_employee_full_name     FROM employees     WHERE last_name = l_employee_last_name;    EXCEPTION      WHEN NO_DATA_FOUND THEN       l_employee_full_name := '(not on file)';     WHEN TOO_MANY_ROWS THEN       l_employee_full_name := '(duplicate name)';    END;    DBMS_OUTPUT.PUT_LINE (l_employee_full_name);  END;</pre> <p>En aquest exemple podem veure que hi ha un bloc de pl/sql dins d'un altre</p>
ERROR ORACLE	SQLCODE	EXCEPTION	RAÓ	
ORA-01043	100	NO_DATA_FOUND	Una sentencia SELECT...INTO no devuelve ninguna fila.	
ORA-01422	-1422	TOO_MANY_ROWS	Una sentencia SELECT...INTO no devuelve más de una fila.	
ORA-00001	-1	DUP_VAL_ON_INDEX	Intentar almacenar un valor que estaba duplicado en el campo clave o en una columna con restricción UNIQUE.	
ORA-01722	-1722	INVALID_NUMBER	Fallo al intentar convertir una cadena a un valor numérico.	
ORA-06502	-6502	VALUE_ERROR	Un error de tipo aritmético, de conversión, de truncamiento, etc.	
ORA-06511	-6511	CURSOR_ALREADY_OPEN	Intentar abrir un cursor que ya está abierto.	
ORA-01001	-1001	INVALID_CURSOR	Intentar realizar una operación no permitida sobre un cursor, como puede ser cerrar un cursor que no está abierto.	
ORA-01476	-1476	ZERO_DIVIDE	Se intenta la división entre cero	
ORA-01017	-1017	LOGIN_DENIED	Intentar conectar a Oracle con un usuario o clave incorrecta	
ORA-01012	-1012	NOT_LOGGED_ON	Intentar acceder a la base de datos sin estar conectado a Oracle	
ORA-06504	-6504	ROWTYPE_MISMATCH	La variable del cursor del HOST y la variable del cursor PL/SQL pertenecen a tipos incompatibles	
ORA-06533	-6533	SUBSCRIPT_OUTSIDE_LIMIT	Intentar acceder a una tabla anidada o a un array con un valor de índice ilegal	
ORA-06500	-6500	STORAGE_ERROR	El bloque PL/SQL se ejecuta fuera de memoria	
ORA-00051	-51	TIMEOUT_ON_RESOURCE	Se excede el tiempo de espera para un recurso	
<p>N'hi ha més, però aquestes són les més utilitzades, cal tenir present que aquestes excepcions no s'han de declarar a la secció DECLARE.</p>				

Hi ha altres errors interns d'Oracle que no tenen assignada una excepció predefinida, tenen un codi d'error i un missatge que descriu l'error. Per obtenir el codi i el missatge es fan servir les funcions: SQLCODE i SQLERRM.

Quan es produeix un d'aquests errors es transfereix control a la secció EXCEPTION on es captura l'error a l'apartat WHEN OTHERS

.....

when OTHERS then

```
DBMS_OUTPUT.PUT_LINE('Error'||SQLCODE||
SQLERRM.)
```

.....

En aquest exemple es mostra un exemple de com es pot tractar un error a l'apartat when others

### Exemples TOO\_MANY\_ROWS

```
declare
  v_order_id number;
begin
  select order_id into v_order_id
  from orders where course_id=5;
  dbms_output.put_line('Orderid is: '||v_order_id);
exception
  when too_many_rows then
    dbms_output.put_line('Too many rows!');
end;
```

```
declare
  v_order_id number;
begin
  select order_id
  into v_order_id
  from orders
  where course_id = 5;
  dbms_output.put_line('Order id is: ' || v_order_id);
exception
  when too_many_rows then
```

```
begin
  select order_id into v_order_id
  from orders
  where course_id = 5 and rownum = 1;
  dbms_output.put_line('Orderid is: ' ||
                        v_order_id);
end;
```

```
end;
```

### Exemples NO\_DATA\_FOUND

```

declare
    v_order_id number;
begin
    select order_id into v_order_id
    from orders where course_id=5;
    dbms_output.put_line('Orderid is: '||v_order_id);
exception
    when no_data_found then
        dbms_output.put_line('No order found!');
end;

```

```

declare
    v_name varchar2(100);
    v_order_id number;
begin
    select name into v_name from coruse where course_id=2;
    dbms_output.put_line('Course name: '||v_name);

    begin
        select order_id into v_order_id
        from orders where course_id=5;
        dbms_output.put_line('Orderid is: '||v_order_id);
    exception
        when no_data_found then
            dbms_output.put_line('No order found!');
    end;

exception
    when no_data_found then
        dbms_output.put_line('No course found!');
end;

```

Exemples OTHERS	
<p>L'excepció <b>OTHERS</b> simbolitza qualsevol condició d'excepció que no ha estat declarada.</p> <p>S'utilitza habitualment per controlar qualsevol tipus d'error que no estat previst. En aquest cas, és comú observar la sentència <b>ROLLBACK</b> al grup de sentències de l'excepció o alguna de les funcions <b>SQLCODE</b> – <b>SQLERRM</b>.</p> <ul style="list-style-type: none"> <li>• <b>SQLCODE</b> torna el nombre de l'error d'Oracle i un 0 (zero) en cas d'èxit en executar una sentència SQL.</li> <li>• <b>SQLERRM</b> retorna la descripció del corresponent missatge d'error. També és possible lliurar-lo a la funció SQLERRM un nombre negatiu que representi un error de Oracle i aquesta tornarà el missatge associat.</li> </ul>	<pre> declare     err_num number;     err_msg varchar2(255);     result number;     msg      varchar2(255); begin     msg := sqlerrm(-1403);     dbms_output.put_line(msg);     select 1 / 0 into result from dual; exception     when others then         err_num := sqlcode;         err_msg := sqlerrm;         dbms_output.put_line('Error: '    to_char(err_num));         dbms_output.put_line(err_msg); end;</pre>
Altres exemples	
<pre> declare     v_order_id number; begin     select order_id     into v_order_id     from orders     where course_id = 1234567;     dbms_output.put_line('Order id is: '    v_order_id); exception     when others then         dbms_output.put_line('When others exception error                                 message: '    sqlerrm); end;</pre>	<pre> declare     v_dsp number; begin     select description into v_dsp     from course     where name = 'SQL 2';     dbms_output.put_line('Description is: '    v_dsp); exception     when value_error then         dbms_output.put_line('VALUE_ERROR: '    sqlerrm);         dbms_output.put_line('Change data type v_dsp in                                 varchar2(2000)'); end;</pre>

## EXCEPCIONS DEFINIDES PER L'USUARI

Definició	Exemple
<p>Són aquelles que crea l'usuari. Per això es requereixen tres passos:</p> <ol style="list-style-type: none"><li>1. Definició: es realitza a la zona DECLARE el tipus es EXCEPTION, <i>semblant a la definició d'una variable</i>.</li><li>2. Provocar o disparar l'excepció mitjançant l'ordre : RAISE</li><li>3. Tractar l'excepció a l'apartat: EXCEPTION: WHEN THEN</li></ol>	<pre>DECLARE ..... import_error EXCEPTION; ... BEGIN ... IF preu NOT BETWEEN minim AND maxim THEN     RAISE import_error; END IF; ... EXCEPTION     WHEN import_error THEN         DBMS_OUTPUT.PUT_LINE('Import erroni'); ... END;</pre>

Definició	Exemple
<p>Combinació de gestió d'excepcions:</p> <ol style="list-style-type: none"> <li>1. ZERO_DIVIDE, predefinida</li> <li>2. e_sinreg , definida per l'usuari</li> <li>3. OTHERS , per a la resta d'excepcions no explicitades</li> </ol>	<pre> <b>DECLARE</b>   e_sinreg <b>EXCEPTION</b>;   a <b>number</b>(10) := 25;   b <b>number</b>(10) := 0;   c <b>number</b>(10); <b>BEGIN</b>   <b>Select</b> count(*) <b>INTO</b> a <b>FROM</b> Articulos;   <b>If</b> a &lt; 10 <b>THEN</b>     <b>RAISE</b> e_sinreg;   <b>END IF</b>;   c := a / b;   DBMS_OUTPUT.PUT_LINE     (' Esto nunca llegará a mostrarse. '); <b>EXCEPTION</b>   <b>WHEN</b> ZERO_DIVIDE <b>THEN</b> DBMS_OUTPUT.PUT_LINE     ('No se puede dividir por 0');   <b>WHEN</b> e_sinreg <b>THEN</b> DBMS_OUTPUT.PUT_LINE     ('Hay menos de 10 articulos. ');   <b>WHEN</b> OTHERS <b>THEN</b> DBMS_OUTPUT.PUT_LINE     ('Se ha producido otra excepción. '); <b>END</b>; </pre>

## EXCEPCIONS «RAISE\_APPLICATION\_ERROR»

Definició	Exemple
<p>Dins el paquet DBMS_STANDARD s'inclou un procediment que s'anomena RAISE_APPLICATION_ERROR, serveix per disparar errors i associar-hi el missatge corresponent.</p> <p>El format és el següent:</p> <ul style="list-style-type: none"> <li>RAISE_APPLICATION_ERROR(numero_error,missatge_error);</li> </ul> <p>Es important tenir en compte el següent:</p> <ul style="list-style-type: none"> <li>el numero d'error ha de ser entre -20000 y -20999</li> <li>el missatge és una cadena de caràcters de fins a 512 bytes.</li> </ul> <p>Aquest procediment dispara una excepció que només pot ser tractada dins l'apartat WHEN OTHERS.</p>	<pre>CREATE or REPLACE PROCEDURE subir_horas   (emple NUMBER, horas_subir NUMBER) IS   horas_actuales NUMBER; BEGIN   Select horas into horas_actuales   from empleados   where id_empleado=emple;   if horas_actuales is NULL then     RAISE_APPLICATION_ERROR(-20010, 'No tiene horas');   else     update empleados     set horas=horas_actuales + horas_subir     where id_empleado=emple;   end if; End subir_horas;</pre>



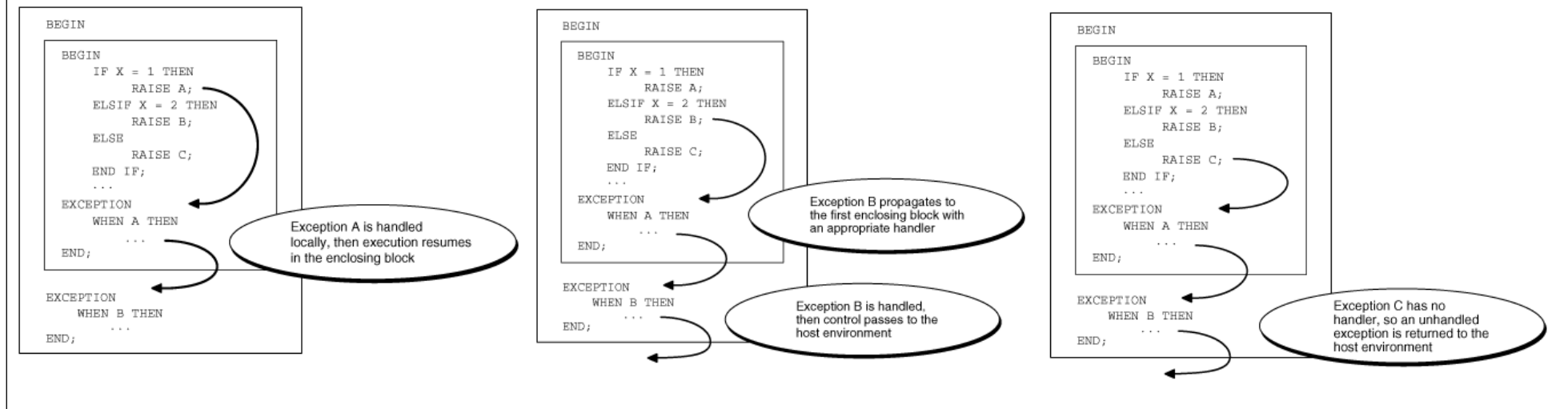
Definició	Exemple
<p>Si creem un procediment com el que es mostra a la dreta obtindrem el següent resultat a l'executar-la amb el paràmetres que es mosten a continuació.</p> <p><b>EXECUCIÓ</b></p> <pre>SET SERVEROUTPUT ON; BEGIN COMPUTE_SAL(7369); END;</pre> <p><b>RESULTAT</b></p> <p>ERROR at line 1:</p> <pre>ORA-20001: Commission is null. ORA-06512: at "SCOTT.COMPUTE_SAL", line 16 ORA-06512: at line 2</pre>	<pre>create procedure compute_sal ( i_empno in emp.empno %type) is      cursor c_emp (p_empno emp.empno%type) is     select sal, comm     from emp     where empno = p_empno;      v_sal number;     v_comm number; begin     open c_emp(i_empno);     fetch c_emp into         v_sal,         v_comm;     close c_emp;     if v_comm is null then         raise_application_error(             -20001,             'Commission is null.');</pre> <pre>    end if;     dbms_output.put_line('Total salary is: '   (v_sal + v_comm)); end compute_sal;</pre>

## EXEMPLES

Cas	Exemple
<p><u>Encara podeu gestionar una excepció per a una instrucció i, a tot seguit, continuar amb la següent instrucció.</u> Col·loqueu la instrucció al seu propi subbloc amb els seus propis controladors d'excepcions. Si es produeix un error al subbloc, un controlador local pot detectar l'excepció. Quan el subbloc acaba, el bloc que l'envolta continua executant-se en el punt on acaba el subbloc.</p>	<pre> DECLARE     sal_calc NUMBER(8,2);  BEGIN     INSERT INTO employees_temp VALUES (303, 2500, 0);      BEGIN -- sub-block begins         SELECT salary / commission_pct INTO sal_calc         FROM employees_temp         WHERE employee_id = 301;     EXCEPTION         WHEN ZERO_DIVIDE THEN             sal_calc := 2500;     END; -- sub-block ends      INSERT INTO employees_temp VALUES (304,     sal_calc/100, .1); EXCEPTION     WHEN ZERO_DIVIDE THEN         NULL; END; </pre>

Cas	Exemple
<p><u>La sentència INSERT pot generar una excepció a causa d'un valor duplicat en una columna única.</u> En aquest cas, canviem el valor que ha de ser únic i continuem amb la següent iteració del bucle. Si l'INSERT té èxit, sortim del bucle immediatament. Amb aquesta tècnica, hauríeu d'utilitzar un bucle FOR o WHILE per limitar el nombre d'intents.</p> <p>A l'exemple de la dreta la definició de les taules de la BD és la següent:</p> <pre>CREATE TABLE results (   res_name VARCHAR(20),   res_answer VARCHAR2(3) ); CREATE UNIQUE INDEX res_name_ix ON results (res_name); INSERT INTO results VALUES ('SMYTHE', 'YES'); INSERT INTO results VALUES ('JONES', 'NO');</pre>	<pre>DECLARE   name      VARCHAR2(20) := 'SMYTHE';   answer    VARCHAR2(3)  := 'NO';   suffix    NUMBER := 1;  BEGIN   FOR i IN 1..5 LOOP  -- try 5 times      BEGIN  -- sub-block begins       SAVEPOINT start_transaction; -- mark a savepoint       /* Remove rows from a table of survey results. */       DELETE FROM results WHERE res_answer = 'NO';       /* Add a survey respondent's name and answers. */       INSERT INTO results VALUES (name, answer);       -- raises DUP_VAL_ON_INDEX if two have the same name       COMMIT;       EXIT;     EXCEPTION       WHEN DUP_VAL_ON_INDEX THEN         ROLLBACK TO start_transaction; -- undo changes         suffix := suffix + 1; -- try to fix problem         name := name    TO_CHAR(suffix);     END;  -- sub-block ends    END LOOP; END;</pre>

Cas	Exemple
<p>Quan s'aixeca una excepció, si PL/SQL no troba un controlador per a aquesta al bloc o subprograma on es produeix, l'excepció es propaga. És a dir, l'excepció es reproduïx en blocs superiors successius, que inclouen el bloc anterior, fins que es troba un controlador (handler) o no hi ha més blocs per cercar. Si no es troba cap controlador, PL/SQL retorna un error d'excepció no gestionat a l'entorn des d'on s'ha cridat el bloc de PL/SQL. Aquest entorn potser un client per interactuar directament amb la BD o un programa java o python que han cridat a un procediment o funció PL/SQL.</p>	<pre> BEGIN   DECLARE ----- sub-block begins     past_due EXCEPTION;     due_date DATE := trunc(SYSDATE) - 1;     todays_date DATE := trunc(SYSDATE);   BEGIN     IF due_date &lt; todays_date THEN       RAISE past_due;     END IF;   END; ----- sub-block ends    EXCEPTION     WHEN OTHERS THEN       ROLLBACK; END; </pre>



## Webgrafia

Enllaços web	
<a href="#">avoiding-too-many-rows-errors-in-pl-sql/</a>	Avoiding TOO_MANY_ROWS errors in PL/SQL
<a href="#">dup_val_on_index.html</a>	PL/SQL Dup_val_on_index
<a href="https://docs.oracle.com/cd/A97630_01/appdev.920/a96624/13_elems48.htm">https://docs.oracle.com/cd/A97630_01/appdev.920/a96624/13_elems48.htm</a>	Errors cursors implícits
<a href="#">.....handling errors.htm#i3359</a>	Handling PL/SQL Errors
<a href="#">...AcompleteexampleusingRAISEAPPLICATIONERROR.htm</a>	A complete example using RAISE_APPLICATION_ERROR