

# PL/SQL Cursores

Antonio Espín Herranz

# Cursores

- Cursores implícitos.
- Cursores explícitos.
- Cursores de actualización.

# Cursores

- Un cursor es un tipo especial de Oracle.
- Representa una zona de memoria que utiliza el motor de Oracle para analizar e interpretar los comandos de SQL.
- Hay dos tipos:
  - **Implícito:** Generado **por Oracle** para cada comando de SQL.
  - **Explícito:** Gestionado **por el usuario** para tratar una instrucción select de varias líneas.

# Cursores implícitos

- Los cursores implícitos se utilizan para realizar consultas **SELECT** que devuelven un único registro.
- Deben tenerse en cuenta los siguientes puntos cuando se utilizan cursores implícitos:
- Con cada cursor implícito debe existir la palabra clave **INTO**.
  - Las variables que reciben los datos devueltos por el cursor tienen que contener el mismo tipo de dato que las columnas de la tabla.
  - Los cursores implícitos solo pueden devolver una única fila. En caso de que se devuelva más de una fila (o ninguna fila) se producirá una excepción

# Ejemplo

- **declare**  
vdescripcion **VARCHAR2**(50);  
**begin**  
**SELECT** DESCRIPCION **INTO** vdescripcion  
**from** PAISES **WHERE** CO\_PAIS = 'ESP';  
  
dbms\_output.put\_line('La lectura del cursor es: ' || vdescripcion);  
**end;**
- Cualquier consulta select que implementemos, Oracle la asociada un cursor implícito.

# Excepciones

- Se pueden producir excepciones en las siguientes situaciones:
  - **NO\_DATA\_FOUND** Se produce cuando una sentencia SELECT intenta recuperar datos pero ninguna fila satisface sus condiciones. Es decir, cuando *"no hay datos"*.
  - **TOO\_MANY\_ROWS** Dado que cada cursor implícito sólo es capaz de recuperar una fila, esta excepción detecta la existencia de más de una fila.

# Atributos

- Disponemos de 4 atributos de los cursores implícitos, a partir de los cuales podemos extraer cierta información:

**SQL%FOUND:** boolean

Si la última instrucción select ha devuelto alguna fila.

**SQL%NOTFOUND:** boolean

Si la última instrucción select no ha devuelto alguna fila.

**SQL%ISOPEN:** boolean.

En este tipo de cursores siempre será falso porque Oracle, cierra el cursor en el momento que extrae los datos.

**SQL%ROWCOUNT:** number.

El número de filas afectadas en la última consulta de acción.

# Cursores explícitos

- Los cursores se deben de declarar dentro de los bloques de código de PL/SQL.
- Se necesita indicar un **nombre** y un **comando SELECT** asociado.
- Cursor nombre\_cursor IS instrucción\_select;
- Las operaciones que realizaremos sobre este cursor serán (y por este orden):
  - Abrir el cursor; → OPEN
  - Tratar las filas; → FETCH (utilizar un bucle);
  - Cerrar el cursor; → CLOSE.



# Cursores explícitos

- **Apertura:**

- Es la siguiente instrucción después de abrirlo.
- La apertura desencadena las siguientes acciones:
  - La asignación de memoria del cursor.
  - El análisis sintáctico y semántico de la instrucción select.
  - Activación de posibles bloqueos (select ... for update);
- La apertura siempre se realiza dentro de la sección begin dentro del bloque.
- Sintaxis: **open nombre\_cursor;**

# Cursores explícitos

- **Tratamiento de filas:**

- Después de abrir el cursor, las filas se tratan una a una para lo cual necesitaremos un bucle.
- Los resultados se cargarán dentro de variables.
- Sintaxis:
  - **FETCH nombre\_cursor INTO lista\_variables;**
- **Se suele utilizar así:**
  - **fetch** c1 into person;
  - **Person se declara → person c1%rowtype;**

# Cursores explícitos

- **Cierre:**

- Después de tratar las filas, cerramos el cursor.
- Sintaxis:
  - **CLOSE nombre\_cursor;**
- Los cursores implícitos declarados por Oracle se cierran automáticamente.

# Cursores explícitos

- Disponemos de una versión del bucle for para cursores.
- Este bucle realiza las 4 operaciones (declara implícitamente la variable, abre el cursor, trata las filas y cierra el cursor).
- Sintaxis:
  - **FOR** variable **IN** cursor **LOOP**
    - Instrucciones;
  - **END LOOP;**

# Ejemplo

-- Activar la salida del buffer.

set serveroutput on;

Declare

cursor c1 is select \* from personas;

person c1%rowtype;

begin

**open** c1;

**fetch** c1 into person;

while c1%found loop

dbms\_output.put\_line(person.nombre || ' ' || person.edad);

**fetch** c1 into person;

end loop;

**close** c1;

end;

/

# Ejemplo for cursor

- - Activar la salida del buffer.

Set serveroutput on

- - Dentro del SQL \*Plus:

SQL> **begin**

**for** cur1 **in** (**select** \* **from** EMPLOYEE) **loop**

    dbms\_output.put\_line(cur1.ID || ' is a value in EMPLOYEE');

**end** loop;

**end**;

/

- Este bucle abre, recorre y cierra el cursor.
- Se accede a los campos a través del cursor “cur1.campo”.

# Atributos de los cursores

- Los atributos nos indican el estado del cursor.
  - **%FOUND:**
    - De tipo boolean.
    - Cuando el cursor es implícito el nombre es: SQL%FOUND.
    - Si es explícito (lo hemos declarado nosotros), el nombre es nombre\_cursor%FOUND, por ejemplo: mi\_cursor%FOUND.
    - Es true si la última instrucción insert, update, delete ha procesado al menos una fila o si la última instrucción ha devuelto una y sólo una fila.
    - Es TRUE si la última instrucción FETCH ha extraído una fila.

# Atributos de los cursores

- **%NOTFOUND:**

- De tipo boolean. Es el inverso al anterior.
- Es TRUE si la última instrucción insert, update, delete no ha procesado ninguna fila.
- O si la instrucción SELECT ... INTO no ha extraído ninguna fila.
- O la última instrucción FETCH no ha extraído ninguna fila.
- Los nombres para implícito y explícito, respectivamente:
  - SQL%NOTFOUND y nombre\_cursor%NOTFOUND.



# Atributos de los cursores

- **%ISOPEN:**

- De tipo boolean.
- Para un cursor implícito es SQL%ISOPEN, siempre es falso ya que Oracle cierra los cursores después de utilizarlos.
- Para uno explícito es nombre\_cursor%ISOPEN.
- Es TRUE si el cursor está abierto.

# Atributos de los cursores

- **%ROWCOUNT:**

- De tipo numérico.
- Para el implícito el nombre es: SQL%ROWCOUNT.
- Contiene el número de filas tratadas por la última instrucción insert, delete, update.
- Devuelve 0 si SELECT...INTO no ha extraído ninguna fila, 1 si ha extraído exactamente 1 fila, y un 2 más de una fila (aunque se generaría una excepción).
- También indica el número de la última fila extraída en la instrucción FETCH.

# RowNum

- Esta variable devuelve un número que indica el orden en que se ha seleccionado la fila desde una tabla.
- Cada fila tiene un número de orden que empiezan por 1, 2, ... n.
- Nos puede servir para listar los n resultados que queramos.
  - `Select * from personas where ROWNUM <= 2;`
  - Si estamos ordenando (ORDER BY) rownum contiene el número de la fila anterior a la selección.

# Cursores de actualización

- En una operación de **modificación** o **eliminación**. La cláusula **CURRENT OF** permite acceder directamente a la que fila que extrae el comando FETCH.
- En la declaración del cursor es necesario preservar las filas mediante un bloqueo de intención: FOR UPDATE OF nom\_col.

# Sintaxis

- Los cursores de actualización se declaran igual que los cursores explícitos, añadiendo **FOR UPDATE** al final de la sentencia select.
- **CURSOR** *nombre\_cursor* **IS** *instrucción\_SELECT* **FOR UPDATE** [*nombre\_col*];
- Para actualizar los datos del cursor hay que ejecutar una sentencia **UPDATE** especificando la clausula **WHERE CURRENT OF** *<cursor\_name>*.

# Ejemplo

- Declare
  - - Declara el cursor con una consulta a la tabla de artículos, indicando un bloqueo de intención. Con esto se evitan modificaciones de otros usuarios entre la lectura y la actualización.
  - CURSOR C1 is
    - Select codigo, precio from articulos  
FOR UPDATE OF precio;
  - - Declara una variable asociada a una fila del cursor, también se puede hacer contra una tabla de la BD.
  - ART c1%ROWTYPE.
- Begin
  - Open C1; - - Abre el cursor.
  - Fetch C1 into ART; - - Lee una fila y la carga en la variable ART.
  - While C1%FOUND loop - - Mientras que haya datos.
    - Update articulos set precio = art.precio \* 2
    - Where **current of** C1; - - Accede a la misma fila extraída del comando FETCH.
    - Fetch C1 into ART; - - Procesa otra línea.
  - End Loop;
  - Commit; - - Hacer efectiva la transacción.
  - Close C1; - - Cierra el cursor.
- End;
- /

# Paso de parámetros en el cursor

- Los cursores se pueden declarar con parámetros.
  - Tenemos que indicar el nombre y el tipo, de cada parámetro.
- Sintaxis:
  - `Cursor nombre_cursor(nombre_parametro tipo,...) is consulta_select;`
- La **asignación** de valor a los parámetros se realiza **al abrir el cursor**.

# Ejemplo

Declare

```
Cursor c1(numero number) is
    Select numped, fechaped from pedidos
    Where numcli = numero;
Vc1 c1%rowtype;
```

Begin

```
-- Al abrir el cursor indicamos el valor del parámetro.
Open c1(1);
-- Procesar los datos ...
Fetch c1 into vc1;
-- Cerrar el cursor.
Close c1;
```

End;

/