

Prácticas sobre conceptos básicos de: PL-SQL

Para todas las prácticas tenéis que presentar el programa hecho junto con el resultado en un único documento word con vuestro nombre .
Revisar el texto teórico de CURSORES . Os he puesto algunos de los ejemplos más significativos del texto. Revisar también los conceptos que se indican en el curso de OPENWEBINARS.

Tipos de cursores

- Cursores implícitos
 - Se utilizan cuando la consulta devuelve un único valor.
- Cursores explícitos
 - Se encuentran declarados y controlados por el programador.
 - Se utilizan cuando la consulta devuelve un conjunto de registros.
 - Son más rápidos.

Pasos para trabajar con cursores

1. Declarar el cursor.
2. Abrir el cursor en el servidor.
3. Recuperar cada una de sus filas (bucle).
4. Cerrar el cursor.

Atributos en cursores implícitos

- SQL%NOTFOUND
 - Devuelve “true” cuando la última sentencia “select” no devuelve ninguna fila, o cuando “insert”, “delete” o “update” no afecten a ninguna fila.
- SQL%FOUND
 - Devuelve “true” cuando la última sentencia “select” devuelve alguna fila, o cuando “insert”, “delete” o “update” afecten a alguna fila.

Atributos en cursores implícitos

- SQL%ROWCOUNT
 - Devuelve el número de fila afectadas por “insert”, “delete” o “update” o las filas devueltas por una sentencia “select”.
- SQL%ISOPEN
 - Siempre devuelve “false”, porque Oracle cierra automáticamente el cursor implícito cuando termina la ejecución de la sentencia “select”.

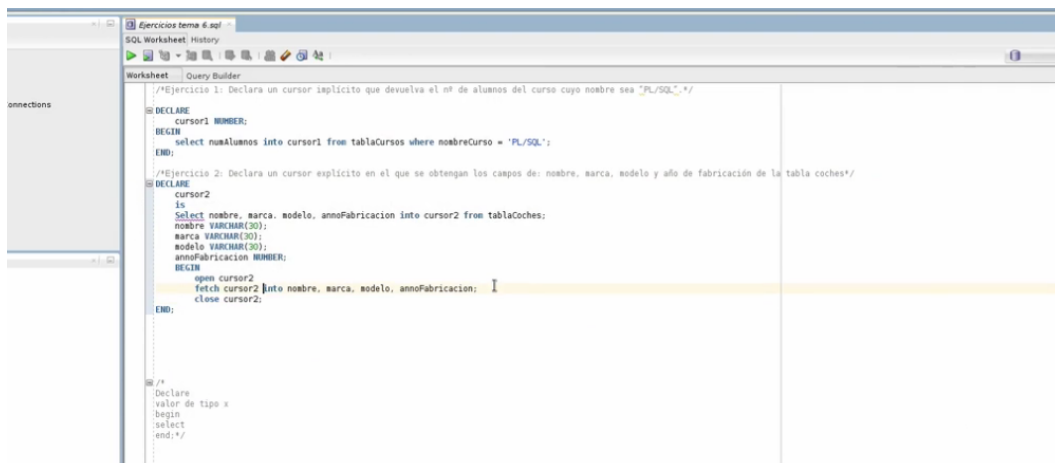
```
/*Ejemplo de cursor implícito*/
SET SERVEROUTPUT ON;
/*Declaración del cursor*/
DECLARE
  vNombre VARCHAR2(50);
  /*Selección de los datos que se van a tratar en el cursor*/
BEGIN
  SELECT nombre INTO vNombre from Alumnos WHERE codAlumno = '1';
  dbms_output.put_line('La lectura del cursor es: ' || vNombre);
END;
```

```
/*Ejemplo de cursos explícito*/
/*Declaración del cursor*/
DECLARE
  CURSOR cAlumnos
  IS
  SELECT codAlumno, nombre, apellidos
  FROM PAISES;
  codAlumno varchar(3);
  nombre varchar(50);
  apellidos varchar(25);
  /*Selección de los datos que se van a tratar en el cursor*/
BEGIN
  OPEN cAlumnos;
  FETCH cAlumnos INTO codAlumno,nombre,apellidos;
  CLOSE cAlumnos;
END;
```

```
/*Ejemplo de un recorrido de un cursor*/
/*Declaración del cursor*/
DECLARE
  CURSOR cAlumnos
  IS
  SELECT codAlumno, nombre, apellidos
  FROM PAISES;
  codAlumno varchar(3);
  nombre varchar(50);
  apellidos varchar(25);
BEGIN
  OPEN cAlumnos;
  /*Bucle para recorrer el cursor*/
  LOOP
    FETCH cAlumnos INTO codAlumno,nombre,apellidos;
    EXIT WHEN cAlumnos%NOTFOUND;
    dbms_output.put_line(nombre);
  END LOOP;
  /*Se cierra el cursor*/
  CLOSE cAlumnos;
END;
```

Ejercicios

1. Declara un cursor implícito que devuelva el nº de alumnos del curso cuyo nombre sea "PL/SQL".
2. Declara un cursor explícito en el que se obtengan los campos de: nombre, marca, modelo y año de fabricación de la tabla coches.



Trabajar con los ejemplos que vienen en OPENWEBINARS. Crearos la tabla de alumnos y de coches y sobre ellas hacer los ejemplos. Además analizar los ejemplos que os he hecho y realizar las prácticas que os indico al final del texto.

Ejemplo1

En el siguiente ejemplo se utilizan ejemplos de cursor explícito para visualizar el código del empleado y su apellido de la tabla de empleados. Las operaciones permitidas de un cursor son: declarar, abrir, recoger información y cerrar el cursor.

Después de cada sentencia FETCH se recupera una fila de la tabla y se pregunta por alguno de los atributos del cursor:

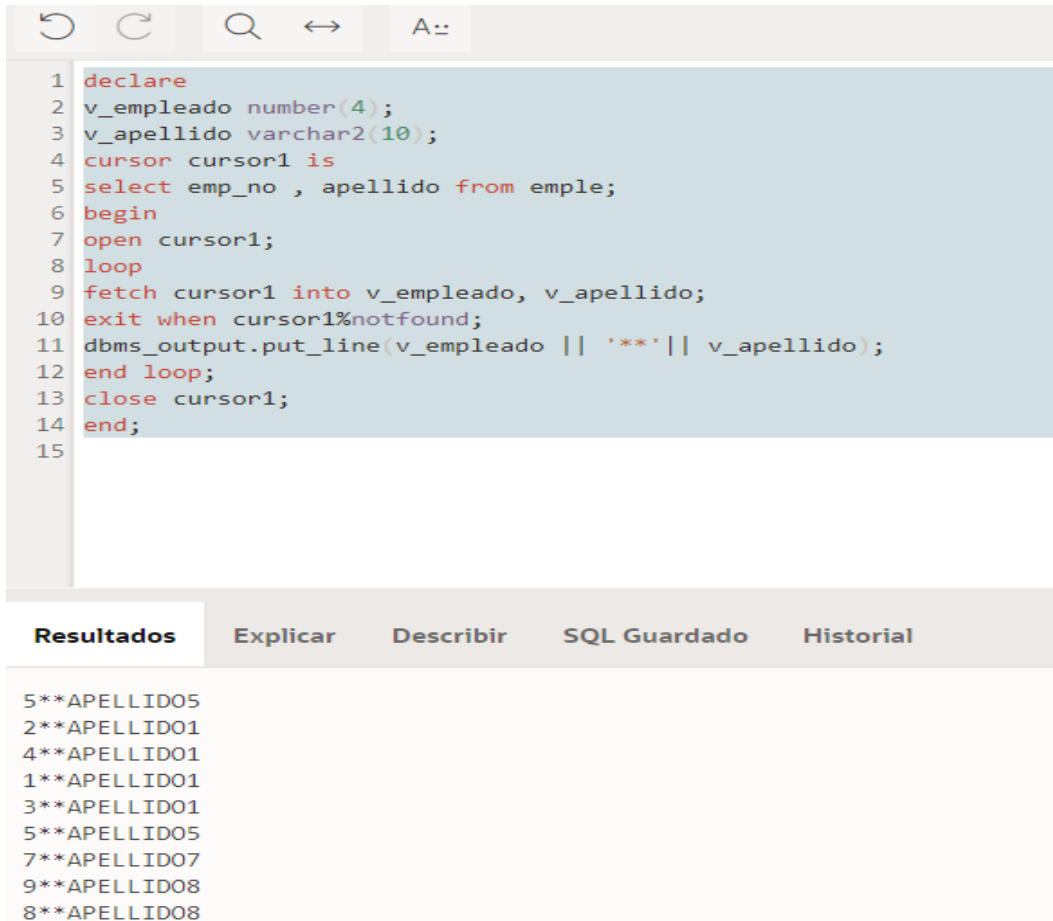
%FOUND : Se controla que el último FETCH recupera valores o no. Se suele utilizar como condición de continuación en el bucle.

%NOTFOUND: Se utiliza como condición de salida del bucle.

%ROWCOUNT: Devuelve el número de filas recuperadas hasta el momento

%ISOPEN: Devuelve verdadero si el cursor está abierto.

```
declare
v_empleado number(4);
v_apellido varchar2(10);
cursor cursor1 is
select emp_no , apellido from emple;
begin
open cursor1;
loop
fetch cursor1 into v_empleado, v_apellido;
exit when cursor1%notfound;
dbms_output.put_line(v_empleado || '**' || v_apellido);
end loop;
close cursor1;
end;
```



The screenshot shows a SQL IDE interface. The top toolbar contains icons for undo, redo, search, and navigation. The main editor displays a PL/SQL script with line numbers 1 through 15. The script declares two variables, opens a cursor, loops through the cursor, and prints the employee number and last name. Below the editor, there are tabs for 'Resultados', 'Explicar', 'Describir', 'SQL Guardado', and 'Historial'. The 'Resultados' tab is active, showing the output of the script: 5**APELLID05, 2**APELLID01, 4**APELLID01, 1**APELLID01, 3**APELLID01, 5**APELLID05, 7**APELLID07, 9**APELLID08, and 8**APELLID08.

```
1 declare
2 v_empleado number(4);
3 v_apellido varchar2(10);
4 cursor cursor1 is
5 select emp_no , apellido from emple;
6 begin
7 open cursor1;
8 loop
9 fetch cursor1 into v_empleado, v_apellido;
10 exit when cursor1%notfound;
11 dbms_output.put_line(v_empleado || '**' || v_apellido);
12 end loop;
13 close cursor1;
14 end;
15
```

Resultados Explicar Describir SQL Guardado Historial

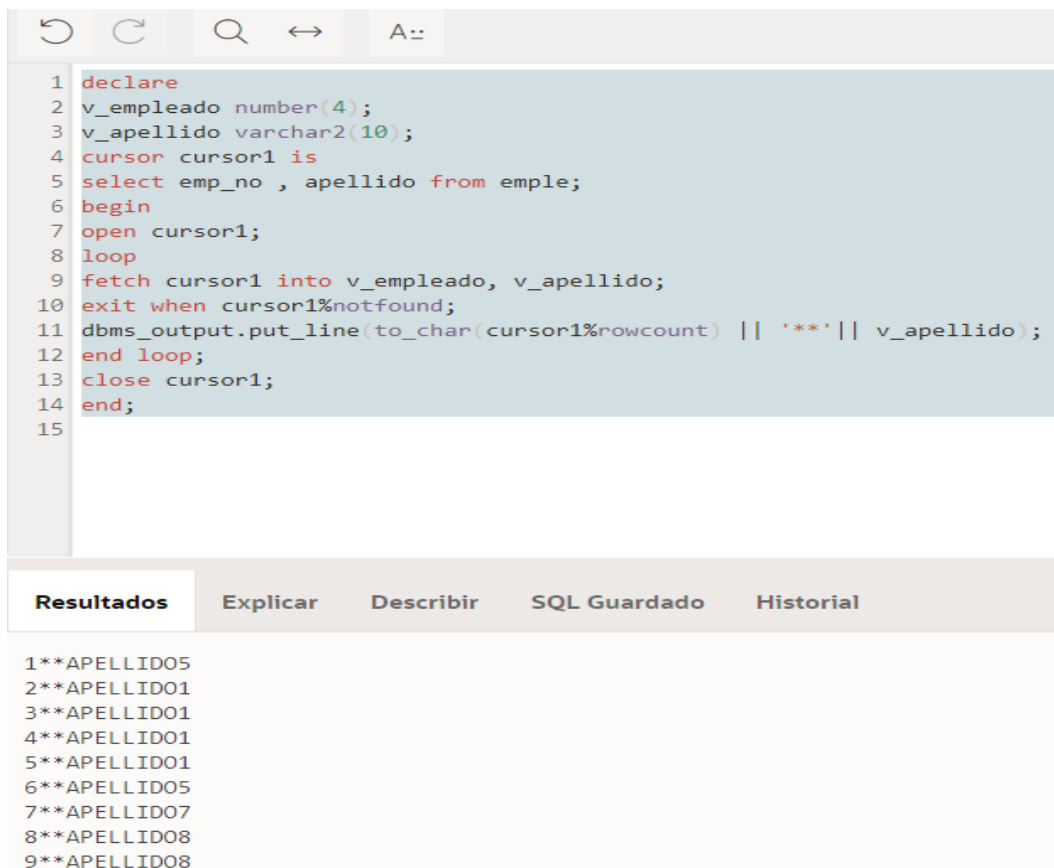
5**APELLID05
2**APELLID01
4**APELLID01
1**APELLID01
3**APELLID01
5**APELLID05
7**APELLID07
9**APELLID08
8**APELLID08

Ejemplo2

Uso de %ROWCOUNT

```
declare
v_empleado number(4);
v_apellido varchar2(10);
cursor cursor1 is
select emp_no , apellido from emple;
begin
open cursor1;
loop
fetch cursor1 into v_empleado, v_apellido;
exit when cursor1%notfound;
dbms_output.put_line(to_char(cursor1%rowcount) || '**' || v_apellido);
end loop;
close cursor1;
```

end;



The screenshot shows a SQL IDE interface. The top toolbar contains icons for undo, redo, search, and a dropdown menu labeled 'A:'. The main editor area displays a PL/SQL script with the following code:

```
1 declare
2 v_empleado number(4);
3 v_apellido varchar2(10);
4 cursor cursor1 is
5 select emp_no , apellido from emple;
6 begin
7 open cursor1;
8 loop
9 fetch cursor1 into v_empleado, v_apellido;
10 exit when cursor1%notfound;
11 dbms_output.put_line(to_char(cursor1%rowcount) || '**' || v_apellido);
12 end loop;
13 close cursor1;
14 end;
```

Below the editor, there is a tabbed interface with four tabs: 'Resultados' (selected), 'Explicar', 'Describir', 'SQL Guardado', and 'Historial'. The 'Resultados' tab displays the output of the script:

```
1**APELLID05
2**APELLID01
3**APELLID01
4**APELLID01
5**APELLID01
6**APELLID05
7**APELLID07
8**APELLID08
9**APELLID08
```

Ejemplo3

Cursores FOR ..LOOP

Se declara el cursor en la sección declarativa (como cualquier otro cursor) y se procesa el cursor utilizando el siguiente formato:

FOR <nombrevariable registro> IN <nombrecursor> LOOP

END LOOP;

Al entrar en el bucle se abre el cursor de manera automática

<nombrevariableregistro> es el nombre de la variable de registro que creará el bucle para recoger los datos del cursor y se ejecuta un FETCH implícito.

A continuación un ejemplo para analizarlo.

```
declare
v_empleado number(4);
v_apellido varchar2(10);
cursor cursor1 is
select emp_no , apellido from emple
where salario > 10000;
begin
for v_registro in cursor1 loop
dbms_output.put_line(v_registro.emp_no || '**' || v_registro.apellido);
end loop;
end;
```

```
1 declare
2 v_empleado number(4);
3 v_apellido varchar2(10);
4 cursor cursor1 is
5 select emp_no , apellido from emple
6 where salario > 10000;
7 begin
8 for v_registro in cursor1 loop
9 dbms_output.put_line(v_registro.emp_no || '**' || v_registro.apellido);
10 end loop;
11 end;
12
```

Resultados	Explicar	Describir	SQL Guardado	Historial
------------	----------	-----------	--------------	-----------

```
5**APELLID05
4**APELLID01
5**APELLID05
7**APELLID07
9**APELLID08
8**APELLID08
```

Ejemplo4

Uso de cursores para actualizar filas:

Se declara el cursor en la sección declarativa (como cualquier otro cursor) y se procesa el cursor utilizando el siguiente formato:

CURSOR <nombrecursor> IS <sentencia SELECT del cursor>

FOR UPDATE;

Los cursores así definidos , permiten actualizar la última fila recuperada con FETCH mediante el comando UPDATE o DELETE incluyendo el especificador WHERE CURRENT OF <nombrecursor> en la cláusula para actualizar o borrar.

UPDATE | DELETEWHERE CURRENT OF <nombre cursor>

El ejemplo recoge datos de un empleado y se suma al salario una determinada cantidad con la forma de proceder que he indicado anteriormente.

```
declare
v_empleado number(4);
v_apellido varchar2(10);
v_incremento number;
cursor cursor1 is
select emp_no , apellido, salario from emple
where emp_no = 1
for update;
v_registro cursor1%ROWTYPE;
begin
open cursor1;
fetch cursor1 into v_registro;
while cursor1%found loop
v_incremento :=100;
update emple set salario = salario + v_incremento
where current of cursor1;
fetch cursor1 into v_registro;
end loop;
dbms_output.put_line(v_registro.emp_no || '**' || v_registro.apellido);
end;
```



```
1 declare
2 v_empleado number(4);
3 v_apellido varchar2(10);
4 v_incremento number;
5 cursor cursor1 is
6 select emp_no , apellido, salario from emple
7 where emp_no = 1
8 for update;
9 v_registro cursor1%ROWTYPE;
10 begin
11 open cursor1;
12 fetch cursor1 into v_registro;
13 while cursor1%found loop
14 v_incremento :=100;
15 update emple set salario = salario + v_incremento
16 where current of cursor1;
17 fetch cursor1 into v_registro;
18 end loop;
19 dbms_output.put_line(v_registro.emp_no || '**' || v_registro.apellido);
20 end;
21
```

Resultados Explicar Describir SQL Guardado Historial

1**APELLIDO1

Sentencia procesada.

Para todas las prácticas tenéis que presentar el programa hecho junto con el resultado en un único documento word con vuestro nombre .

Práctica1

Construir un bloque PLSQL con un cursor para ver todos los departamentos de la tabla DEPART con todos sus atributos asociados Hacedlo con LOOP. Controlar fin de tabla DEPART. Visualizar todos los campos de la tabla línea a línea.

Práctica2

Construir un bloque PLSQL con un cursor para ver todos los departamentos de la tabla DEPART con todos sus atributos asociados. Hacedlo con FOR .. LOOP. Controlar fin de tabla DEPART. Visualizar todos los campos de la tabla línea a línea.

Práctica3

Construir un bloque PLSQL con un cursor para actualizar el salario de los empleados de la tabla EMPLE que superen una determinada cantidad e incrementarla en 1000.
Hacedlo con LOOP. Utilizar FOR UPDATE.
Visualizar empleado, apellido y oficio de la tabla línea a línea.

Práctica4

Construir un bloque PLSQL con un cursor para actualizar el salario de los empleados de la tabla EMPLE que superen una determinada cantidad e incrementarla en 1000.
Hacedlo con FOR ..LOOP. Utilizar FOR UPDATE . Visualizar los campos de empleado apellido y oficio línea a línea.