

Cursores en postgres

Jesús Reyes Carvajal

Un cursor es el nombre para un área memoria privada que contiene información procedente de la ejecución de una sentencia SELECT.

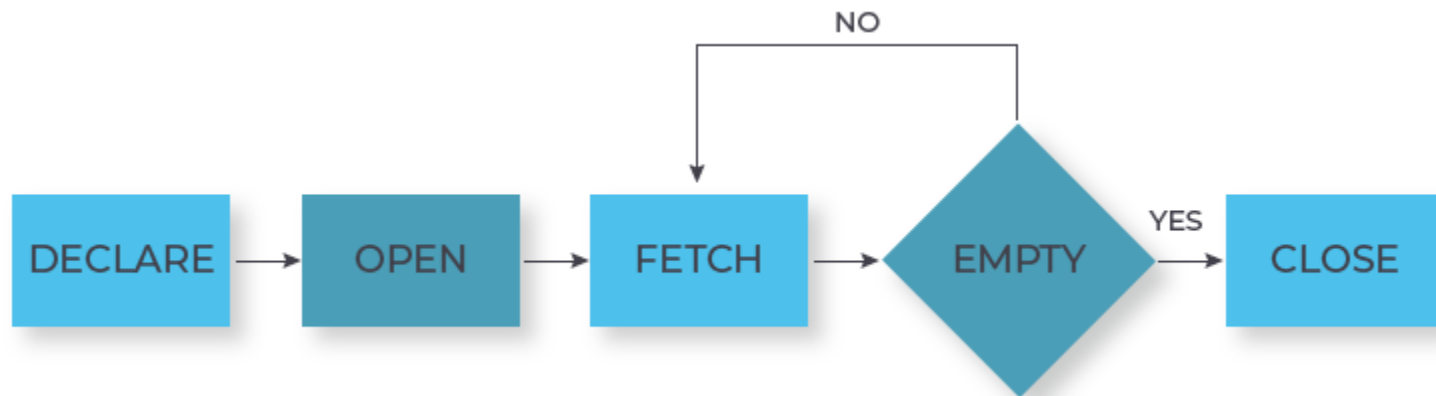
Cada cursor tiene unos atributos que nos devuelven información útil sobre el estado del cursor en la ejecución de la sentencia SQL.

Podemos distinguir dos tipos de cursores:

- **Cursores implícitos.** Este tipo de cursores se utiliza para operaciones SELECT INTO. Se usan cuando la consulta devuelve un único registro.
- **Cursores explícitos.** Son los cursores que son declarados y controlados por el programador. Se utilizan cuando la consulta devuelve un conjunto de registros. Ocasionalmente también se utilizan en consultas que devuelven un único registro por razones de eficiencia. Son más rápidos.

Para trabajar con un cursor hay que realizar los siguientes pasos:

- Declarar el cursor
- Abrir el cursor en el servidor
- Recuperar cada una de sus filas (bucle)
- Cerrar el cursor



DECLARE: permite a un usuario crear cursores, los cuales pueden ser usados para recuperar un número de filas a la vez provenientes de una consulta mas extensa. Los cursores pueden devolver datos ya sea en formato de texto o en formato binario.

Sintaxis:

DECLARE cursorname CURSOR FOR query;

cursorname: nombre del cursor a ser usado en subsecuentes operaciones FETCH

query: Consulta SQL la cual proveerá las filas a ser manipuladas.

Los cursores solo están disponibles en las transacciones y por tanto se deben tener en cuenta las siguientes sentencias SQL:

- BEGIN
- COMMIT
- ROLLBACK

Propiedades de las transacciones

Toda transacción debe cumplir las cuatro propiedades conocidas como ACID:

Atomicidad: unidad atómica, indivisible de ejecución. O se ejecuta completamente o no se ejecuta nada.

Consistencia: debe de preservar la consistencia de la base de datos.

Aislamiento: una transacción no puede verse afectada por la ejecución de otra.

Definitiva: los resultados de una transacción llevada a cabo exitosamente tienen que ser definitivos en la base de datos.

BEGIN: Inicia una transacción

CLOSE: Cierra el cursor actual

COMMIT: Termina la transacción

ROLLBACK: Aborta la transacción o interrumpe la transacción actual.

BEGIN;

DECLARE cursor1 CURSOR for select * from equipos ORDER BY nomb_equ;

FETCH 1 IN cursor1;

CLOSE cursor1;

COMMIT;

FETCH

FETCH -- retrieve rows from a query using a cursor

FETCH [direction { FROM | IN }] cursorname

direction defines the fetch direction and number of rows to fetch. It can be one of the following:

NEXT: Fetch the next row. This is the default if direction is omitted.

PRIOR: Fetch the prior row. (devuelve el cursor una posición antes de la posición actual)

FIRST: Fetch the first row of the query (same as ABSOLUTE 1).

LAST: Fetch the last row of the query (same as ABSOLUTE -1).

count: Fetch the next count rows (same as FORWARD count).

ALL: Fetch all remaining rows (same as FORWARD ALL).

Continuación.. FETCH

FORWARD: Fetch the next row (same as NEXT).

FORWARD count: Fetch the next count rows. FORWARD 0 re-fetches the current row.

FORWARD ALL: Fetch all remaining rows.

BACKWARD: Fetch the prior row (same as PRIOR).

BACKWARD count: Fetch the prior count rows (scanning backwards). BACKWARD 0 re-fetches the current row.

BACKWARD ALL: Fetch all prior rows (scanning backwards).

BEGIN;

DECLARE cur1 CURSOR FOR select * from productos;

FETCH 1 IN cur1;

p1		Panela		1200		12		ca1
----	--	--------	--	------	--	----	--	-----

FETCH 2 IN cur1;

p2		Azucar		1500		8		ca1
----	--	--------	--	------	--	---	--	-----

p3		Bolsa agua		800		10		ca5
----	--	------------	--	-----	--	----	--	-----

FETCH LAST IN cur1;

p8		Gaseosa 1LT		2800		20		ca5
----	--	-------------	--	------	--	----	--	-----

FETCH prior IN cur1;

p7		Gaseosa		1800		23		ca5
----	--	---------	--	------	--	----	--	-----

FETCH backward 2 IN cur1;

p5		Arroz 500g		1200		10		ca4
----	--	------------	--	------	--	----	--	-----

p6		Arroz 1000g		2000		12		ca4
----	--	-------------	--	------	--	----	--	-----

FETCH first IN cur1;

p1		Panela		1200		12		ca1
----	--	--------	--	------	--	----	--	-----

FETCH next IN cur1;

p2		Azucar		1500		8		ca1
----	--	--------	--	------	--	---	--	-----

FETCH forwark 6 IN cur1;

p3		Bolsa agua		800		10		ca5
----	--	------------	--	-----	--	----	--	-----

p4		Botella agua		1800		15		ca5
----	--	--------------	--	------	--	----	--	-----

p5		Arroz 500g		1200		10		ca4
----	--	------------	--	------	--	----	--	-----

p6		Arroz 1000g		2000		12		ca4
----	--	-------------	--	------	--	----	--	-----

p7		Gaseosa		1800		23		ca5
----	--	---------	--	------	--	----	--	-----

p8		Gaseosa 1LT		2800		20		ca5
----	--	-------------	--	------	--	----	--	-----

COMMIT;

cod_prod		nomb_prod		precio		stock		cod_cat
-----	+	-----	+	-----	+	-----	+	-----
p1		Panela		1200		12		ca1
p2		Azucar		1500		8		ca1
p3		Bolsa agua		800		10		ca5
p4		Botella agua		1800		15		ca5
p5		Arroz 500g		1200		10		ca4
p6		Arroz 1000g		2000		12		ca4
p7		Gaseosa		1800		23		ca5
p8		Gaseosa 1LT		2800		20		ca5


```
CREATE OR REPLACE FUNCTION numciclistas(cur1 refcursor)
RETURNS refcursor AS $$
BEGIN
```

```
    OPEN cur1 FOR SELECT * FROM ciclistas;
    RETURN cur1;
```

```
END
$$
LANGUAGE plpgsql;
```

```
--INICIO DE TRANSACCION
```

```
ciclismo=# BEGIN;
```

```
BEGIN
```

```
ciclismo=# SELECT numciclistas('asd');
numciclistas
```

```
-----
```

```
asd
(1 row)
```

```
ciclismo=# fetch asd;
```

```
cod_cic | nomb_cic | fechan | cod_nac
-----+-----+-----+-----
```

```
c1 | Vincenzo Nibali | 1991-02-16 | p91
(1 row)
```

```
ciclismo=# move forward 3 from asd;
```

```
MOVE 3
```

```
ciclismo=# fetch asd;
```

```
cod_cic | nomb_cic | fechan | cod_nac
-----+-----+-----+-----
```

```
c5 | Dmitriy Gruzdev | 1988-04-11 | p95
(1 row)
```

```
ciclismo=# fetch prior from asd;
```

```
cod_cic | nomb_cic | fechan | cod_nac
-----+-----+-----+-----
```

```
c4 | Andriy Grivko | 1989-12-13 | p185
(1 row)
```

```
ciclismo=# fetch first from asd;
```

```
cod_cic | nomb_cic | fechan | cod_nac
-----+-----+-----+-----
```

```
c1 | Vincenzo Nibali | 1991-02-16 | p91
(1 row)
```

```
ciclismo=#COMMIT;
```

Ejemplo de un cursor en una función

cod_cic	nomb_cic	fechan	cod_nac
c1	Vincenzo Nibali	1991-02-16	p91
c2	Lars Boom	1989-10-13	p80
c3	Jakob Fuglsang	1991-01-12	p50
c4	Andriy Grivko	1989-12-13	p185
c5	Dmitriy Gruzdev	1988-04-11	p95
c6	Tanel Kangert	1991-02-16	p61
c7	Michele Scarponi	1993-02-09	p91
c8	Rein Taaramae	1986-10-09	p61
c9	Lieuwe Westra	1986-11-14	p80
c10	Fabio Aru	1993-02-09	p91
c11	Jean-Christophe Péraud	1991-04-09	p66
c12	Jan Bakelants	1992-03-11	p14
c13	Romain Bardet	1990-01-02	p66
c14	Patrick Gretsch	1990-07-03	p3