



Instituto Politécnico Nacional

Escuela Superior de Cómputo



Distributed Data Base

Reporte de laboratorio #6

Nombre: De la rosa Hernández Samuel

Grupo: 3CM5

Fecha de entrega: 25/03/2019

Índice

Marco teórico	2
Instrucciones	3
Conclusiones.....	3
Referencias	13

Marco teórico

Returning Data Using Result Sets

If you include a `SELECT` statement in the body of a stored procedure (but not a `SELECT ... INTO` or `INSERT ... SELECT`), the rows specified by the `SELECT` statement will be sent directly to the client. For large result sets the stored procedure execution will not continue to the next statement until the result set has been completely sent to the client. For small result sets the results will be spooled for return to the client and execution will continue. If multiple such `SELECT` statements are run during the execution of the stored procedure, multiple result sets will be sent to the client. This behavior also applies to nested TSQL batches, nested stored procedures and top-level TSQL batches.

Instrucciones

1) Resolver lo siguiente

a. Con base al nombre del club conocer los gerentes asociados a el.

delimiter #

create procedure p1(in cb varchar(30))

begin

select g.nombre from gerente g, club c

where g.idclub=c.idclub

and c.nombre like concat(cb,"%")

order by g.nombre;

end #

delimiter ;

```
mysql> delimiter #
mysql> create procedure p1(in cb varchar(30))
-> begin
-> select g.nombre from gerente g, club c
-> where g.idclub=c.idclub
-> and c.nombre like concat(cb,"%")
-> order by g.nombre;
-> end #
Query OK, 0 rows affected (0.26 sec)
mysql> delimiter ;
```

call p1("Pachuca");

```
mysql> call p1("Pachuca");
+-----+
| nombre |
+-----+
| FLORES MONDRAGON HUGO MICHELLE |
| HERNANDEZ LOPEZ FILEMON        |
+-----+
2 rows in set (0.13 sec)
Query OK, 0 rows affected (0.14 sec)
```

b. Con base al nombre del club mostrar los servicios que provee.

delimiter \$

create procedure p2(in x varchar(30))

begin

select s.nombre from servicio s,

servicioclub sc, club c

```

where c.idclub=sc.idclub
and sc.idservicio=s.idservicio
and c.nombre like concat(x,"%")
order by s.nombre;

end $

delimiter ;

```

```

mysql> delimiter $
mysql> create procedure p2(in x varchar(30))
-> begin
-> select s.nombre from servicio s,
-> servicioclub sc, club c
-> where c.idclub=sc.idclub
-> and sc.idservicio=s.idservicio
-> and c.nombre like concat(x,"%")
-> order by s.nombre;
-> end $
Query OK, 0 rows affected (0.22 sec)
mysql> delimiter ;

```

call p2("Durango");

```

mysql> call p2("Durango");
+-----+
| nombre |
+-----+
| Farmacia |
| Foto Centro |
| Joyeria |
| Servicio a Domicilio |
| TMA Centro Llantero |
| Optica |
+-----+
6 rows in set (0.09 sec)
Query OK, 0 rows affected (0.11 sec)

```

- c. Con base al nombre de un proveedor mostrar los productos que suministra.

```

delimiter %
create procedure p3(in p varchar(20))
begin
select pd.nombre
from proveedor pv, producto pd
where pd.idproveedor=pv.idproveedor
and pv.nombre like concat(p,"%")
order by 1;

end %

```

delimiter ;

```
mysql> delimiter %
mysql> create procedure p3(in p varchar(20))
-> begin
-> select pd.nombre
-> from proveedor pv, producto pd
-> where pd.idproveedor=pv.idproveedor
-> and pv.nombre like concat(p,"%")
-> order by 1;
-> end %
Query OK, 0 rows affected (0.19 sec)
mysql> delimiter ;
```

call p3("Sabritas");

```
mysql> call p3("Sabritas");
+-----+
| nombre |
+-----+
| Sabritas Cheetos |
| Sabritas Churumais |
| Sabritas Crujitos |
| Sabritas Doritos Nacho |
| Sabritas Fritos |
| Sabritas Poffets |
| Sabritas Rancheritos |
| Sabritas Rufles |
| Sabritas Sabritones |
| Sabritas Sun Chips |
| Sabritas Tacos |
| Sabritas Tacos |
| Sabritas Tradicionales |
+-----+
13 rows in set (0.05 sec)
Query OK, 0 rows affected (0.07 sec)
```

- d. Con base a un estado, mostrar el nombre y el teléfono de los clubes existentes en dicho estado.

delimiter #

create procedure p4(in s varchar(20))

begin

select c.nombre, c.tel

from club c, estado e

where c.idedo=e.idedo

and e.nombre like concat(s,"%")

order by c.nombre;

end #

delimiter ;

```
mysql> delimiter #
mysql> create procedure p4(in s varchar(20))
-> begin
-> select c.nombre, c.tel
-> from club c, estado e
-> where c.idedo=e.idedo
-> and e.nombre like concat(s,"%")
-> order by c.nombre;
-> end #
Query OK, 0 rows affected (0.11 sec)
mysql> delimiter ;
```

call p4("Durango");

```
mysql> call p4("Durango")
-> ;
+-----+-----+
| nombre | tel |
+-----+-----+
| Durango | Tel.: (618) 813-9070 |
| Villa Jardin | Tel.: 5532-78-90 |
+-----+-----+
2 rows in set (0.04 sec)
Query OK, 0 rows affected (0.05 sec)
```

- e. Con base a un código postal, mostrar el nombre del club y el estado donde se ubica:

```
delimiter $
create procedure p5(in cp int)
begin
select c.nombre, e.nombre
from club c, estado e
where c.idedo=e.idedo
and c.direccion like concat("%",cp,"%")
order by 1;
end $
delimiter ;
```

```
mysql> delimiter $
mysql> create procedure p5(in cp int)
-> begin
-> select c.nombre, e.nombre
-> from club c, estado e
-> where c.idedo=e.idedo
-> and c.direccion like concat("%",cp,"%")
-> order by 1;
-> end $
Query OK, 0 rows affected (0.09 sec)
mysql> delimiter ;
```

call p5("77710");

```
mysql> call p5("77710");
+-----+-----+
| nombre | nombre |
+-----+-----+
| Playa del Carmen | Quintana Roo |
+-----+-----+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.01 sec)
```

f. Actualizar el teléfono de un socio determinado.

delimiter #

create procedure c6(in id int, in t varchar(15))

begin

select idSocio, nombre, tel

from socio

where idsocio=id;

update socio set tel=t

where idsocio=id;

select idSocio, nombre, tel

from socio

where idsocio=id;

end #

delimiter ;

```
mysql> delimiter #
mysql> create procedure c6(in id int, in t varchar(15))
-> begin
-> select idSocio, nombre, tel
-> from socio
-> where idsocio=id;
->
-> update socio set tel=t
-> where idsocio=id;
->
-> select idSocio, nombre, tel
-> from socio
-> where idsocio=id;
-> end #
Query OK, 0 rows affected (0.11 sec)

mysql> delimiter ;
```

call c6(20,"55-55-55-55-99");


```
mysql> call c6(20,"55-55-55-55-99");
```

idSocio	nombre	tel
20	PIMENTEL MARTINEZ WALDO	554-234-5555

```
1 row in set (0.01 sec)
```

idSocio	nombre	tel
20	PIMENTEL MARTINEZ WALDO	55-55-55-55-99

```
1 row in set (0.23 sec)
```

```
Query OK, 0 rows affected (0.24 sec)
```

g. Dar de alta a un proveedor.

```
delimiter #
```

```
create procedure c7(in id int, in n varchar(45), in t varchar(25))
```

```
begin
```

```
insert into proveedor
```

```
values(id,n,t);
```

```
select * from proveedor
```

```
where idproveedor=id;
```

```
end #
```

```
delimiter ;
```

```
mysql> delimiter #
mysql> create procedure c7(in id int, in n varchar(45), in t varchar(25))
-> begin
-> insert into proveedor
-> values(id,n,t);
-> select * from proveedor
-> where idproveedor=id;
-> end #
Query OK, 0 rows affected (0.18 sec)
mysql> delimiter ;
```

```
call c7(999,"sam","55-98-87");
```

```
mysql> call c7(999,"Sam","55-98-87");
```

idProveedor	nombre	tel
999	Sam	55-98-87

```
1 row in set (0.07 sec)
```

```
Query OK, 0 rows affected (0.07 sec)
```

h. Reasignar a un gerente a un club determinado.

```

delimiter #
create procedure c8(in idg int, in nc varchar(20))
begin

select g.nombre, c.nombre
from gerente g, club c
where g.idclub=c.idclub
and g.idgerente=idg;

update gerente
set g.idclub=(select idclub from club where nombre like concat(nc,"%"))
where idgerente=idg;

select g.nombre, c.nombre
from gerente g, club c
where g.idclub=c.idclub
and g.idgerente=idg;

end #

delimiter ;

```

```

mysql> delimiter #
mysql> create procedure c8(in idg int, in nc varchar(20))
-> begin
-> select g.nombre, c.nombre
-> from gerente g, club c
-> where g.idclub=c.idclub
-> and g.idgerente=idg;
->
-> update gerente
-> set g.idclub=(select idclub from club where nombre like concat(nc,"%"))
-> where idgerente=idg;
->
-> select g.nombre, c.nombre
-> from gerente g, club c
-> where g.idclub=c.idclub
-> and g.idgerente=idg;
-> end #
Query OK, 0 rows affected (0.11 sec)
mysql> delimiter ;

```

call c8(195,"Durango");

```

mysql> call c8(195,"Durango");
+-----+-----+
| nombre | nombre |
+-----+-----+
| SALAZAR OLGUIN LUCIA ALEJANDRA | Villahermosa |
+-----+-----+
1 row in set (0.00 sec)

```

- i. Eliminar a los gerentes que tienen un apellido paterno determinado.

```

delimiter #
create procedure c9(in ap varchar(30))
begin

select nombre
from gerente
where nombre like concat("%",ap,"%")
order by 1;

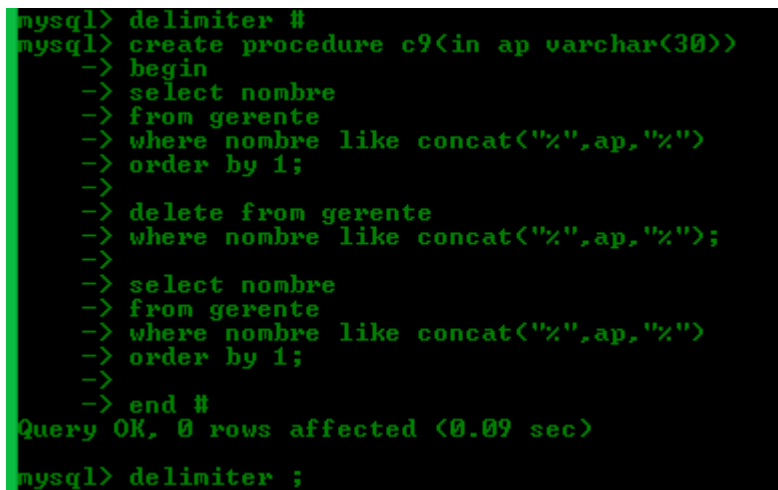
delete from gerente
where nombre like concat("%",ap,"%");

select nombre
from gerente
where nombre like concat("%",ap,"%")
order by 1;

end #

delimiter ;

```



```

mysql> delimiter #
mysql> create procedure c9<in ap varchar(30)>
-> begin
-> select nombre
-> from gerente
-> where nombre like concat("<div data-bbox="133 732 261 751" data-label="Text">


```

call c9("Lopez");

```


```

```
mysql> call c9("Lopez");
+-----+
| nombre |
+-----+
| GONZALEZ LOPEZ ALAN |
| HERNANDEZ LOPEZ FILEMON |
| LOPEZ GARCIA JAVIER |
| LOPEZ HERNANDEZ YAZMIN SARAH |
| LOPEZ ISLAS MARIANA |
| LOPEZ LOPEZ MARCIAL |
| LOPEZ MARRON RICARDO NESTOR |
| LOPEZ OBREGON ERIKA LISSETTE |
| LOPEZ PADILLA DIEGO GERARDO |
| LOPEZ SALVADOR ROBERT |
| MADRID LOPEZ HECTOR EDUARDO |
| SANCHEZ LOPEZ JESSICA JOSHELIN |
| TAPIA LOPEZ ARIANNA VANET |
+-----+
13 rows in set (0.02 sec)

Empty set (0.19 sec)

Query OK, 0 rows affected (0.21 sec)
```

Conclusiones

En esta práctica podemos ocupar DDL con PROCEDURE para poder realizar procedimientos que puedan modificar algunas tuplas.

Referencias

1. Date C. J. Introducción a los Sistemas de Bases de Datos, Séptima Edición. Pearson Educación de México, México 2001.
2. Microsoft Corporation , Access SQL: conceptos básicos, vocabulario y sintaxis, Virginia Bristow U.S. , <https://support.office.com/es-es/article/access-sql-conceptos-b%C3%A1sicos-vocabulario-y-sintaxis-444d0303-cde1-424e-9a74-e8dc3e460671>