

Subconsultas

Introducción a las subconsultas

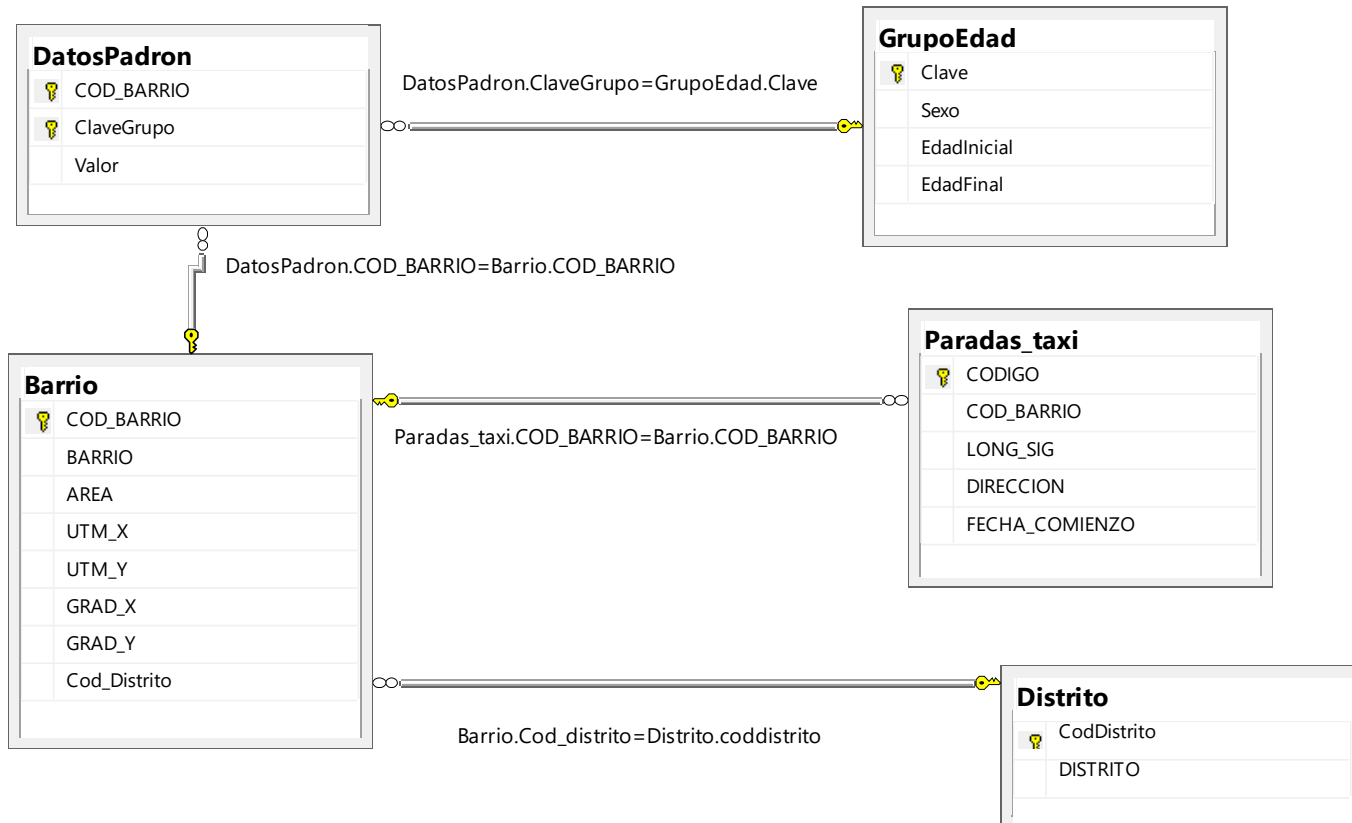
Qué es:

Select colocado entre paréntesis

Por qué utilizar subconsultas

- Para dividir una consulta compleja en varios pasos lógicos
- Para responder una consulta que depende de los resultados de otra consulta

DatosSCTFE



1.- Uso de una subconsulta como una tabla derivada

- Es un conjunto de registros dentro de una consulta que funciona como una tabla
- Ocupa el lugar de la tabla en la cláusula FROM
- Cuando queremos partir de una consulta compleja ya resuelta.
- Primero se resuelve la subconsulta y se trabaja con ella como si su resultado fuera una tabla.
- Es fundamental darle nombre para poder referirnos a ella o a sus datos.

1.- Uso de una subconsulta como una tabla derivada

```
use DatosSCTFE;
go
select barrio, padron, npar
from barrio
inner join (select COD_BARRIO, count(*) as npar
            from Paradas_taxi
            group by COD_BARRIO) as spar
on spar.COD_BARRIO=barrio.COD_BARRIO
inner join (select COD_BARRIO, sum(valor) as padron
            from DatosPadron
            group by COD_BARRIO) as spad
on barrio.COD_BARRIO=spad.COD_BARRIO;
```

2.- Uso de una subconsulta como una expresión

- Se evalúa y trata como una expresión
- Se ejecuta una vez para la instrucción entera
- Sólo puede devolver un único valor

```
use DatosSCTFE;  
go  
select sum(valor) as padron,  
       (select count(*) as npar from Paradas_taxi) as totpar  
from DatosPadron;
```

3.- Uso de una subconsulta en la condición WHERE

- Podemos usar una subconsulta en la cláusula **where**
- Según los resultados de la subconsulta tendremos diferentes posibilidades
 - **Devuelven único valor**
 - Comparar con operador **>, <, <=, >=, <>, =**
 - **Devuelven una lista de valores**
 - Comparar con **IN** que indica si el valor está en la lista definida por la subconsulta
 - Comparar con operador **>, <, <=, >=, <>, =** mediante el modificador **ANY/SOME ó ALL**, que indica si se verifica para alguno o todos los valores.

3.- Uso de una subconsulta en la condición WHERE

Comparar con operador >, <, <=, >=, <>
Para subconsultas que devuelven un único valor.

```
use DatosSCTFE;
go
select barrio, sum(valor)
from barrio
inner join DatosPadron
on barrio.COD_BARRIO=DatosPadron.COD_BARRIO
group by barrio
having sum(valor) > (select sum(valor) as padron
                    from DatosPadron)*0.05;
```


3.- Uso de una subconsulta en la condición WHERE

Comparar con **IN** que indica si el valor está en la lista definida por la subconsulta.

Para subconsultas que devuelven una lista de valores.

```
use DatosSCTFE;
go
select barrio, sum(valor) as padron
from barrio
inner join DatosPadron
on barrio.COD_BARRIO=DatosPadron.COD_BARRIO
where barrio.COD_BARRIO in (select COD_BARRIO as npar
                             from Paradas_taxi
                             group by COD_BARRIO
                             having count(*)>2)
group by BARRIO;
```

3.- Uso de una subconsulta en la condición WHERE

Comparar con operador $>$, $<$, \leq , \geq , \neq

Mediante el modificador ANY/SOME ó ALL

Valida si se verifica para alguno o todos los valores.

Para subconsultas que devuelven una lista de valores.

3.- Uso de una subconsulta en la condición WHERE

```
use DatosSCTFE;
go
select barrio, sum(valor) as padron
from barrio as b
inner join DatosPadron
on b.COD_BARRIO=DatosPadron.COD_BARRIO
group by barrio
having sum(valor) < all (select sum(valor)
                        from DatosPadron
                        inner join Paradas_taxi
                        on DatosPadron.COD_BARRIO=Paradas_taxi.COD_BARRIO
                        group by DatosPadron.COD_BARRIO);
```

3.- Uso de una subconsulta en la condición WHERE

>, <, <=, >=, <> ANY o SOME

Condición válida para alguno de los valores devueltos por la subconsulta.

>, <, <=, >=, <> ALL

Condición válida para todos los valores devueltos por la subconsulta.

Uso de una subconsulta para correlacionar datos

- Enlazar los datos de una consulta con los de una subconsulta
- En la subconsulta se pueden usar campos de los registros de la consulta
- La subconsulta se recalcula para cada registro de la consulta principal.

Uso de una subconsulta para correlacionar datos

```
use DatosSCTFE;
go
select Barrio, count(*) as npar,
       (select sum(valor) as padron
        from DatosPadron as dp
        where dp.COD_BARRIO=barrio.COD_BARRIO)
from Paradas_taxi
inner join BARRIO
on Paradas_taxi.COD_BARRIO=BARRIO.COD_BARRIO
group by Barrio,barrio.COD_BARRIO;
```

Uso de las cláusulas EXISTS y NOT EXISTS

Validaremos si devuelve resultados (EXISTS) o no (NOT EXISTS).

La subconsulta debe estar correlacionada para que cambie su resultado al ir recorriendo la consulta.

```
use DatosSCTFE;
go
select barrio, sum(valor) as padron
from barrio as b
inner join DatosPadron
on b.COD_BARRIO=DatosPadron.COD_BARRIO
where exists (select pt.COD_BARRIO
              from Paradas_taxi as pt
              where pt.COD_BARRIO=b.COD_BARRIO)
group by barrio;
```

Evaluación de una subconsulta correlacionada

1 La consulta externa pasa un valor de columna a la consulta interna

```
USE northwind
SELECT orderid, customerid
FROM orders AS or1
WHERE 20 < (SELECT quantity
            FROM [order details] AS od
            WHERE or1.orderid = od.orderid
            AND  od.productid = 23)
```

GO

3 La consulta interna devuelve un valor a la consulta externa

2 La consulta interna utiliza los valores que pasa la consulta externa

4 Este proceso se repite para la fila siguiente de la consulta externa

Volver al paso 1

Subconsultas en las instrucciones UPDATE, DELETE e INSERT

Insert

```
insert into TABLAENQUESEINGRESA  
(CAMPOSTABLA1)  
select CAMPOSTABLACONSULTADA  
from TABLACONSULTADA;
```

Subconsultas en las instrucciones UPDATE, DELETE e INSERT

Usando las condiciones where de subconsultas

Por ejemplo

Update

```
update TABLA set  
CAMPO=NUEVOVALOR  
where CAMPO=  
(SUBCONSULTA);
```

Delete

```
delete from TABLA  
where CAMPO in  
(SUBCONSULTA);
```

Resumen

(select ...) as nombre

Subconsulta es un select entre paréntesis.

Según el número de resultados tendremos distintas posibilidades de uso:

- Valor único
- Tabla
- Columna con valores

Campos correlacionados: campos de la select externa que se pasan a la subconsulta que llevan a recalcular la subconsulta para cada registro de la consulta externa.

Resumen

Valor único

Podemos usarlo donde se usaban campos o valores.

Por ejemplo como contenido del select o como parte del where.

Tabla

Podemos usarlo donde se usaban Tablas, dentro del From fundamentalmente.

También vale para el exists.

Columna

Podemos usarlo en el Where o en el having mediante expresiones que vemos después.

Resumen

Campo	IN	(subconsulta de columna)
Valor	NOT IN	

Será cierta cuando el valor o el contenido del campo sea uno de los valores que devuelve la subconsulta

Resumen

Campo	>		
Valor	>=	ALL	(subconsulta de columna)
	<		
	<=		
	<>		Será cierta cuando el valor o el contenido del campo cumpla con la condición para TODOS
	=		los valores que devuelva la subconsulta.

Resumen

Campo	>	ANY	(subconsulta de columna)
Valor	>=	SOME	
	<		
	<=		
	<>		
	=		
			Será cierta cuando el valor o el contenido del campo cumpla con la condición para ALGUNO de los valores que devuelva la subconsulta.

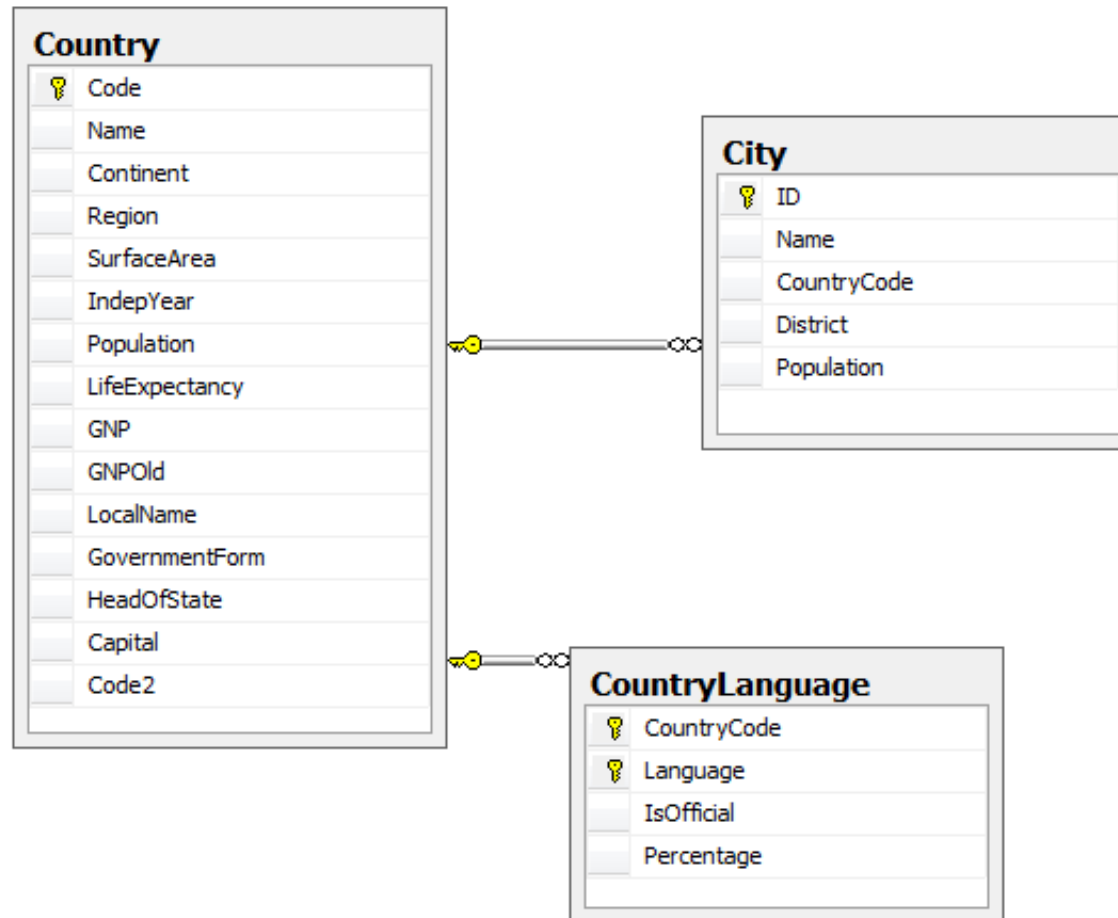
Resumen

EXISTS
NOT EXISTS **(subconsulta de tabla)**

Será cierta cuando la subconsulta devuelva algún valor.
Suele usarse con campos correlacionados.

Subconsultas mediante ejemplos

Mundo



Subconsulta como tabla

Nº de ciudades y nº de idiomas de cada país.

```

--Como tabla
--n° de ciudades y n° de idiomas de un país.
select Name, nleng,ncity
from country as c
inner join (select countrycode, count(*) as Nleng
            from countrylanguage
            group by countrycode
            ) as nl
    on nl.countrycode=c.code
inner join (select countrycode,count(*) as Ncity
            from city
            group by CountryCode
            ) as nc
    on nc.CountryCode=c.Code

```

Resultados		Mensajes	
	Name	nleng	ncity
1	Aruba	4	1
2	Afghanistan	5	4
3	Angola	9	5
4	Anguilla	1	2
5	Albania	3	1
6	Andorra	4	1
7	Netherlands Antilles	3	1
8	United Arab Emirates	2	5

Subconsulta como campos.

País con más habitantes y ciudad con más habitantes

```
--campos
--país con más habitantes y ciudad con más habitantes
select top 1 Name as paismas,
        (select top 1 name
         from city
         order by population desc) as ciudadmas
from Country
order by Population desc;
go
```

Resultados		Mensajes
	paismas	ciudadmas
1	China	Mumbai (Bombay)

Subconsulta como valor en el Where

Países con población más alta que el
50% de la mayor

```
--Where valor
--Países con población más alta que el 50% de la mayor
select name
from country
where Population >
    ((select MAX(population) from country)*0.5);
go
```

Resultados	
	name
1	China
2	India

Subconsulta en el where mediante el exists
y usando campos enlazados

Países con idioma con más del 70% de la
población.

```
--exists y campos enlazados
--países con idioma con más del 70% de la población.
select name
from country as c
where exists (select language
              from CountryLanguage as l
              where c.code=l.countrycode
                 and Percentage>70);
go
```

Resultados		Mensajes	
		name	
1		Aruba	
2		Albania	
3		Netherlands Antilles	
4		Argentina	
5		Armenia	
6		American Samoa	
7		Antigua and Barbuda	
8		Australia	

Subconsulta en el Where mediante el in

Países con alguno de los idiomas hablados
en Japón (Japan)

```
--Where in
--países con alguno de los idiomas hablados en japon
select p1.name
from country as p1
inner join countrylanguage as l1
    on l1.CountryCode=p1.code
where l1.language in (select Language
                      from country as p2
                      inner join countrylanguage as l2
                          on l2.CountryCode=p2.code
                      where p2.name='japan')
group by p1.name;
```

	name
1	American Samoa
2	Anguilla
3	Antigua and Barbuda
4	Aruba
5	Australia
6	Bahrain
7	Barbados
8	Belize

Subconsultas en el Where usando any/some

País población menor que alguna de las ciudades de España.

```
--where any/some
--País población menor que alguna de las ciudades de España.
select name
from Country
where population < any (select c.population
                        from city as c
                        inner join Country as p
                        on p.code=c.countrycode
                        where p.name='Spain');
go
```

Resultados		Mensajes
	name	
1	Aruba	
2	Anguilla	
3	Andorra	
4	Netherlands Antilles	
5	United Arab Emirates	
6	American Samoa	
7	Antarctica	
8	French Southern territories	

Subconsultas en el where mediante all

Países con población mayor que todos
los países europeos

```
--where all
--países con población mayor que todos los países europeos
select name
from country
where population > all (select p.population
                        from Country as p
                        where p.Continent='Europe');
go
```

Resultados		Mensajes	
	name		
1	Brazil		
2	China		
3	Indonesia		
4	India		
5	Pakistan		
6	United States		

Subconsulta mediante campos enlazados.

Ciudad e idioma con más población por
país

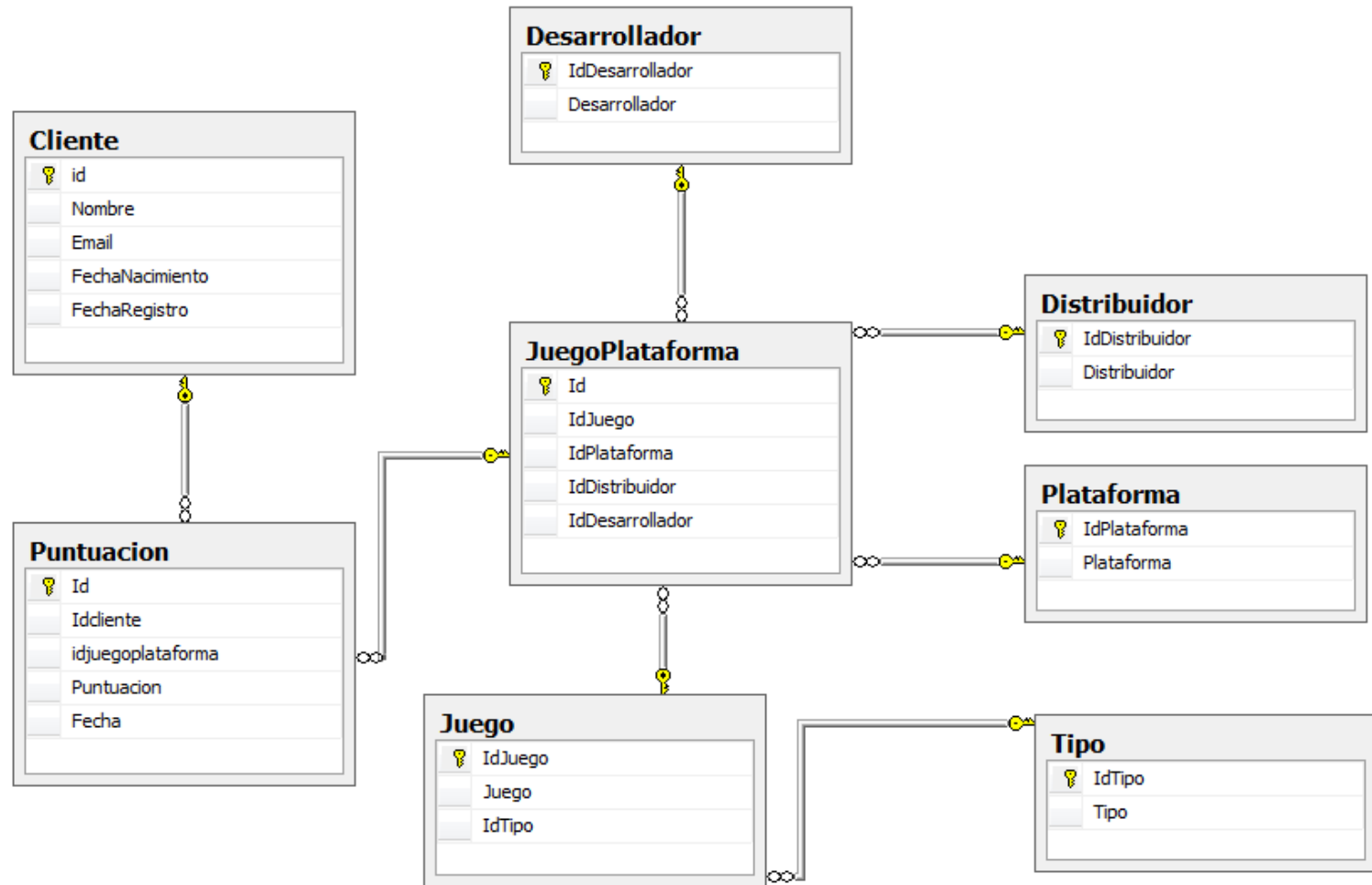
```

--campos enlazados
--ciudad e idioma con más población por país
select name,
    (select top 1 Language
     from CountryLanguage
     where country.code=CountryLanguage.CountryCode
     order by percentage desc) as IdiomaMas,
    (select top 1 Name
     from City
     where country.code=City.CountryCode
     order by population desc) as CiudadMas
from country
order by name;
go

```

	name	IdiomaMas	CiudadMas
1	Afghanistan	Pashto	Kabul
2	Albania	Albaniana	Tirana
3	Algeria	Arabic	Alger
4	American Samoa	Samoan	Tafuna
5	Andorra	Spanish	Andorra la Vella
6	Angola	Ovimbundu	Luanda
7	Anguilla	English	South Hill
8	Antarctica	NULL	NULL

Videojuegos



Subconsulta como tabla

Suma del número de puntuaciones
de cada tipo de juego

```

--Como tabla
--suma del número de puntuaciones de cada tipo de juego
select tipo, sum(npunt)
from tipo as t
inner join
    (select Juego,idtipo, COUNT(*) as npunt
     from puntuacion as p
     inner join juegoplataforma as jp
       on p.idjuegoplataforma=jp.id
     inner join juego as j
       on j.idjuego=jp.idjuego
     group by Juego,idtipo) as jcuenta
  on t.IdTipo=jcuenta.idtipo
group by tipo;
go

```

Resultados		Mensajes
	tipo	(Sin nombre d
1	Acción	25
2	Aventura	4
3	Conducción	2
4	Deportes	2
5	Estrategia	3
6	Otros	8
7	Rol	10
8	Simulación	2

Subconsulta en campos

Distribuidor y desarrollador con más juegos.

```

--campos
--distribuidor y desarrollador con más juegos.
select
    (select top 1 Desarrollador as desarr
     from Desarrollador as d
     inner join juegoplataforma as jp
         on d.IdDesarrollador=jp.IdDesarrollador
     group by d.Desarrollador
     order by count(distinct idjuego) desc) as desa,
    (select top 1 Distribuidor as distrib
     from Distribuidor as d
     inner join juegoplataforma as jp
         on d.IdDistribuidor=jp.IdDistribuidor
     group by d.Distribuidor
     order by count(distinct idjuego) desc) as distri

```

	Resultados	Mensajes
	desa	distri
1	Robot Entertainment	Por determinar

Subconsulta en Where como valor

Puntuaciones, dando juego, plataforma y nombre del cliente con puntuación mayor que la media de las puntuaciones


```
--Where valor  
--puntuaciones, dando juego, plataforma y nombre del cliente con puntuación  
--mayor que la media de las puntuaciones
```

```
select Juego, Plataforma,nombre,puntuacion  
from cliente as c  
inner join puntuacion as p  
    on c.id=p.idcliente  
inner join juegoplataforma as jp  
    on p.idjuegoplataforma=jp.Id  
inner join juego as j  
    on j.idjuego=jp.idjuego  
inner join Plataforma as pl  
    on pl.IdPlataforma=jp.idplataforma  
where Puntuacion >  
    (select avg(puntuacion) from puntuacion)
```

Resultados		Mensajes		
	Juego	Plataforma	nombre	puntuacion
1	BioShock Infinite	PS3	Jack English	10
2	The Legend of Zelda	Wii	Neville Chang	9
3	Los Sims	PC	Hayden Chavez	7
4	Call of Duty: Black Ops	PS3	Amal Harrington	9
5	League of Legends	PC	Nehru Atkins	8

Subconsulta en el where mediante exists y con enlace de campos

Juegos/plataforma cuyo distribuidor no exista en otro juego

```

|--exists con enlace de campos
--juegos/plataforma cuyo distribuidor no exista en otro juego
select distribuidor, juego, plataforma
from distribuidor as d
inner join juegoplataforma as jp
    on d.iddistribuidor=jp.iddistribuidor
inner join juego as j
    on j.idjuego=jp.IdJuego
inner join Plataforma as p
    on jp.IdPlataforma=p.idplataforma
where not exists
(
select juego
from distribuidor as d2
inner join juegoplataforma as jp2
    on d2.iddistribuidor=jp2.iddistribuidor
inner join juego as j2
    on j2.idjuego=jp2.idjuego
where j2.idjuego <> j.idjuego
    and d2.IdDistribuidor=d.iddistribuidor);
go

```

	distribuidor	juego	plataforma
1	Brandon Brizzi	1000 Amps	Mac
2	Brandon Brizzi	1000 Amps	PC
3	UIG Entertai...	1953 - KGB Un...	PC
4	3D Retro Ga...	3D Ground Zero	PC
5	Playrix Enter...	4 Elements	PC

Subconsulta en el Where usando el in

Juegos con tipo que tengan más de 1000
juegos

```
--Where in
--Juegos con tipo que tengan más de 1000 juegos
select Juego
from Juego
where idtipo in
    (select t.idtipo
     from tipo as t
     inner join Juego as j
       on j.IdTipo=t.IdTipo
     group by t.idtipo
     having count(*) >1000)|
```

Resultados		Mensajes
Juego		
1	0 Day Attack on Earth	
2	007 Legends	
3	0D Beat Drop	
4	0x10c	
5	1 vs 100	
6	1... 2... 3... KICK IT!	
7	10 Minutes Solution	
8	100 juegos de mesa	
-	

Subconsultas en Where mediante in

Cientes que han votado a los 10
juegos con mayor puntuación media

```

--Where in
--clientes que han votado a los 10 juegos con mayor
--puntuación media
select nombre
from cliente as c
inner join puntuacion as p
    on c.id=p.idcliente
inner join JuegoPlataforma as jp
    on p.idjuegoplataforma=jp.Id
where jp.idjuego in
    (select top 10 jpl.idjuego
     from puntuacion as p1
     inner join juegoplataforma as jpl
         on p1.idjuegoplataforma=jpl.id
     group by jpl.idjuego
     order by avg(puntuacion) desc)
group by Nombre;
go

```

Resultados		Mensajes
	nombre	
1	Acton Mckay	
2	Alexander Roth	
3	Damon Morton	
4	Emmanuel Rutledge	
5	Jack English	
6	Joshua Campbell	
7	Lucian Pugh	
8	Neville Chang	
9	Raymond Velazquez	

Subconsultas en Where usando any/some

Juego con distribuidor igual a alguno de los
del juego FIFA 10: Ultimate Team


```

--where any/some
--juego con distribuidor igual a alguno de los del juego
--FIFA 10: Ultimate Team
select distribuidor, juego
from distribuidor as d
inner join juegoplataforma as jp
    on d.iddistribuidor=jp.iddistribuidor
inner join juego as j
    on j.idjuego=jp.IdJuego
where distribuidor = any
(select distribuidor
from distribuidor as d2
inner join juegoplataforma as jp2
    on d2.iddistribuidor=jp2.iddistribuidor
inner join juego as j2
    on j2.idjuego=jp2.idjuego
    where j2.juego = 'FIFA 10: Ultimate Team')
group by distribuidor,juego;
go

```

Resultados		Mensajes
	distribuidor	juego
1	EA	3 on 3 NHL Arcade
2	EA	AC/DC Live: Rock Band
3	EA	Ace Combat Zero: The Belkan War
4	EA	Alice: Madness Returns
5	EA	American McGee's Alice
6	EA	ANNO 1503: The New World

Subconsultas en el where usando all

Juego programado por el mismo
desarrollador en todas las plataformas

```
--where all
--juego programado por el mismo desarrollador en todas
--las plataformas
select desarrollador, juego
from desarrollador as d
inner join juegoplataforma as jp
    on d.iddesarrollador=jp.iddesarrollador
inner join juego as j
    on j.idjuego=jp.IdJuego
where desarrollador = all
    (select desarrollador
    from desarrollador as d2
    inner join juegoplataforma as jp2
        on d2.iddesarrollador=jp2.iddesarrollador
    inner join juego as j2
        on j2.idjuego=jp2.idjuego
    where j2.IdJuego=j.idjuego)
group by desarrollador, juego;
go
```

	desarrollador	juego
7	10tacle Studios	Urban Race
8	10th Art Studio	Audere Semper
9	11 bit studios	Anomaly 2
10	11 bit studios	Anomaly: Korea
11	11 bit studios	Anomaly: Warzone Earth
12	11 bit studios	Sleepwalker's Journey
13	14*Fast	Fallout Tactics · Brotherhood of Steel

Subconsulta con campos enlazados

Para cada juego/plataforma dar el nombre del juego, la plataforma y el número total de juegos de esa plataforma.

```

--campos enlazados
--para cada juego/plataforma dar el nombre del juego,
--la plataforma y el número total de juegos de esa plataforma.
select Juego, Plataforma,
       (select COUNT(distinct idjuego)
        from juegoplataforma as jp2
        where jp2.IdPlataforma=jp.idplataforma) as Ntotal
from juego as j
inner join JuegoPlataforma as jp
    on j.IdJuego=jp.IdJuego
inner join Plataforma as p
    on p.idplataforma=jp.idplataforma

```

Resultados		Mensajes	
	Juego	Plataforma	Ntotal
1	0 Day Attack on Earth	X360	2215
2	007 Legends	PC	3891
3	007 Legends	PS3	2116
4	007 Legends	Wii U	151
5	007 Legends	X360	2215
6	0D Beat Drop	X360	2215
7	0x10n	PC	3891