

DML

JOINS



Producto Cartesiano

Para obtener reportes de 2 o más tablas es necesario generar un producto cartesiano.

Ejemplo: Usando la base de datos: pixup

1. Seleccionar los datos de la tabla: estado y municipio

```
mysql> select * from estado;
```

clave	nombre
01	Aguascalientes
02	Baja California
03	Baja California Sur
04	Campeche
07	Chiapas
08	Chihuahua
09	Ciudad de México
05	Coahuila
06	Colima
10	Durango
11	Estado de México

```
mysql> select * from municipio limit 20;
```

clave	nombre	clave_estado
01001	Aguascalientes	01
01002	Asientos	01
01003	Calvillo	01
01004	Cosío	01
01005	Jesús María	01
01006	Pabellón de Arteaga	01
01007	Rincón de Romos	01
01008	San José de Gracia	01
01009	Tepezalá	01
01010	El Llano	01
01011	San Francisco de los Romo	01
02001	Ensenada	02
02002	Mexicali	02
02003	Tecate	02
02004	Tijuana	02
02005	Playas de Rosarito	02
03001	Comondú	03
03002	Mulegé	03
03003	La Paz	03
03008	Los Cabos	03

20 rows in set (0.00 sec)

Producto x

2. Seleccionar los datos de ambas tablas juntas

```
mysql> select * from municipio,estado limit 20;
```

clave	nombre	clave_estado	clave	nombre
01001	Aguascalientes	01	01	Aguascalientes
01001	Aguascalientes	01	02	Baja California
01001	Aguascalientes	01	03	Baja California Sur
01001	Aguascalientes	01	04	Campeche
01001	Aguascalientes	01	07	Chiapas
01001	Aguascalientes	01	08	Chihuahua
01001	Aguascalientes	01	09	Ciudad de México
01001	Aguascalientes	01	05	Coahuila
01001	Aguascalientes	01	06	Colima
01001	Aguascalientes	01	10	Durango
01001	Aguascalientes	01	11	Estado de México
01001	Aguascalientes	01	12	Guanajuato
01001	Aguascalientes	01	13	Guerrero
01001	Aguascalientes	01	14	Hidalgo
01001	Aguascalientes	01	15	Jalisco
01001	Aguascalientes	01	16	Michoacán
01001	Aguascalientes	01	17	Morelos
01001	Aguascalientes	01	18	Nayarit
01001	Aguascalientes	01	19	Nuevo León
01001	Aguascalientes	01	20	Oaxaca

20 rows in set (0.02 sec)

Producto x

```
select * from municipio, estado
where clave_estado = estado.clave;
```

```
mysql> select * from municipio, estado where clave_estado = estado.clave limit 20;
```

clave	nombre	clave_estado	clave	nombre
01001	Aguascalientes	01	01	Aguascalientes
01002	Asientos	01	01	Aguascalientes
01003	Calvillo	01	01	Aguascalientes
01004	Cosío	01	01	Aguascalientes
01005	Jesús María	01	01	Aguascalientes
01006	Pabellón de Arteaga	01	01	Aguascalientes
01007	Rincón de Romos	01	01	Aguascalientes
01008	San José de Gracia	01	01	Aguascalientes
01009	Tepezalá	01	01	Aguascalientes
01010	El Llano	01	01	Aguascalientes
01011	San Francisco de los Romo	01	01	Aguascalientes
02001	Ensenada	02	02	Baja California
02002	Mexicali	02	02	Baja California
02003	Tecate	02	02	Baja California
02004	Tijuana	02	02	Baja California
02005	Playas de Rosarito	02	02	Baja California
03001	Comondú	03	03	Baja California Sur
03002	Mulegé	03	03	Baja California Sur
03003	La Paz	03	03	Baja California Sur
03008	Los Cabos	03	03	Baja California Sur

```
20 rows in set (0.02 sec)
```

INNER JOIN (JOIN)

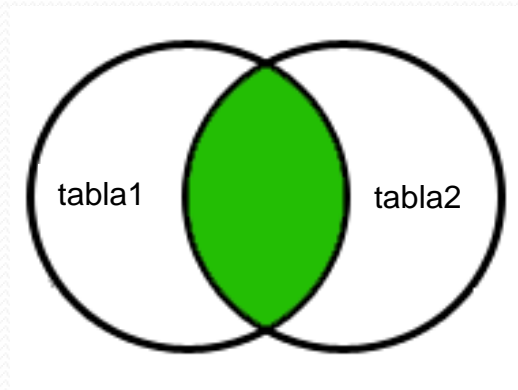
Correspondencia uno a uno

INNER JOIN

Es una sentencia SQL que permite consultar datos de 2 o más tablas. Dichas tablas deben estar relacionadas por algún atributo. En álgebra relacional es la operación *Reunión natural* ⋈

1. INNER JOIN o JOIN: Correspondencia uno a uno

Consiste en combinar cada fila de una tabla con cada fila de la otra tabla, seleccionando las filas que cumplan con una determinada condición si se usa where.



INNER JOIN

1. Seleccionar los datos de las tablas estado y municipio donde coincidan las claves de los estados

```
mysql> select * from municipio limit 20;
```

clave	nombre	clave_estado
01001	Aguascalientes	01
01002	Asientos	01
01003	Calvillo	01
01004	Cosío	01
01005	Jesús María	01
01006	Pabellón de Arteaga	01
01007	Rincón de Romos	01
01008	San José de Gracia	01
01009	Tepezalá	01
01010	El Llano	01
01011	San Francisco de los Romo	01
02001	Ensenada	02
02002	Mexicali	02
02003	Tecate	02
02004	Tijuana	02
02005	Playas de Rosarito	02
03001	Comondú	03
03002	Mulegé	03
03003	La Paz	03
03008	Los Cabos	03

20 rows in set (0.00 sec)

```
mysql> select * from estado;
```

clave	nombre
01	Aguascalientes
02	Baja California
03	Baja California Sur
04	Campeche
07	Chiapas
08	Chihuahua
09	Ciudad de México
05	Coahuila
06	Colima
10	Durango
11	Estado de México

INNER JOIN

- Sintaxis 1: Selección de todos los atributos y uso de ON

SELECT *

FROM tabla_1

JOIN tabla_2 **ON** condicion;

Ejemplo: Todas las columnas de ambas tablas:

SELECT *

FROM municipio

JOIN estado **ON** clave_estado = estado.clave

LIMIT 20;

```
mysql> Select * from municipio join estado on clave_estado = estado.clave limit 20;
```

clave	nombre	clave_estado	clave	nombre
01001	Aguascalientes	01	01	Aguascalientes
01002	Asientos	01	01	Aguascalientes
01003	Calvillo	01	01	Aguascalientes
01004	Cosío	01	01	Aguascalientes
01005	Jesús María	01	01	Aguascalientes
01006	Pabellón de Arteaga	01	01	Aguascalientes
01007	Rincón de Romos	01	01	Aguascalientes
01008	San José de Gracia	01	01	Aguascalientes
01009	Tepezalá	01	01	Aguascalientes
01010	El Llano	01	01	Aguascalientes
01011	San Francisco de los Romo	01	01	Aguascalientes
02001	Ensenada	02	02	Baja California
02002	Mexicali	02	02	Baja California
02003	Tecate	02	02	Baja California
02004	Tijuana	02	02	Baja California
02005	Playas de Rosarito	02	02	Baja California
03001	Comondú	03	03	Baja California Sur
03002	Mulegé	03	03	Baja California Sur
03003	La Paz	03	03	Baja California Sur
03008	Los Cabos	03	03	Baja California Sur

20 rows in set (0.00 sec)

INNER JOIN

- **Sintaxis 1: Selección de algunos atributos y uso de ON**

select col1,col2,colN **FROM** tabla_1 **JOIN** tabla_2 **ON** condicion;

Cuando las columnas de ambas tablas se llaman igual, se obtiene un error de ambigüedad

ERROR 1052 (23000): Column 'nombre' in field list is ambiguous

Para evitar errores de ambigüedad, se hace referencia al nombre de la tabla que se usará o se indica un alias.

Ejemplo: estado.nombre, municipio.nombre, estado.clave

o bien con un alias

estado **as** e → **e.nombre**

INNER JOIN

- **Sintaxis 1: Selección de algunos atributos y uso de ON**

Ejemplo 1: Selección de columnas

```
select estado.nombre, municipio.clave,municipio.nombre  
from municipio  
join estado on estado.clave = clave_estado  
limit 20;
```

nombre	clave	nombre
Aguascalientes	01001	Aguascalientes
Aguascalientes	01002	Asientos
Aguascalientes	01003	Calvillo
Aguascalientes	01004	Cosío
Aguascalientes	01005	Jesús María
Aguascalientes	01006	Pabellón de Arteaga
Aguascalientes	01007	Rincón de Romos
Aguascalientes	01008	San José de Gracia
Aguascalientes	01009	Tepezalá
Aguascalientes	01010	El Llano
Aguascalientes	01011	San Francisco de los Romo
Baja California	02001	Ensenada
Baja California	02002	Mexicali
Baja California	02003	Tecate

INNER JOIN

- **Sintaxis 1: Selección de algunos atributos y uso de ON**

Ejemplo 2: Uso de alias para evitar ambigüedad

```
select m.clave,m.nombre,e.clave,e.nombre  
from municipio as m  
join estado as e on e.clave = clave_estado  
limit 20;
```

clave	nombre	clave	nombre
01001	Aguascalientes	01	Aguascalientes
01002	Asientos	01	Aguascalientes
01003	Calvillo	01	Aguascalientes
01004	Cosío	01	Aguascalientes
01005	Jesús María	01	Aguascalientes
01006	Pabellón de Arteaga	01	Aguascalientes
01007	Rincón de Romos	01	Aguascalientes
01008	San José de Gracia	01	Aguascalientes
01009	Tepezalá	01	Aguascalientes
01010	El Llano	01	Aguascalientes
01011	San Francisco de los Romo	01	Aguascalientes
02001	Ensenada	02	Baja California
02002	Mexicali	02	Baja California
02003	Tecate	02	Baja California
02004	Tijuana	02	Baja California
02005	Playas de Rosarito	02	Baja California
03001	Comondú	03	Baja California Sur
03002	Mulegé	03	Baja California Sur
03003	La Paz	03	Baja California Sur
03008	Los Cabos	03	Baja California Sur

INNER JOIN

- Sintaxis 1: Selección de algunos atributos , uso de ON y alias

Ejemplo 3: Cambiar el nombre de columnas con alias

```
select m.clave as cve_mun,m.nombre as municipio,e.nombre as  
estado  
from municipio as m  
join estado as e on clave_estado = e.clave  
limit 20;
```

cve_mun	municipio	estado
01001	Aguascalientes	Aguascalientes
01002	Asientos	Aguascalientes
01003	Calvillo	Aguascalientes
01004	Cosío	Aguascalientes
01005	Jesús María	Aguascalientes
01006	Pabellón de Arteaga	Aguascalientes
01007	Rincón de Romos	Aguascalientes
01008	San José de Gracia	Aguascalientes
01009	Tepezalá	Aguascalientes
01010	El Llano	Aguascalientes
01011	San Francisco de los Romo	Aguascalientes
02001	Ensenada	Baja California
02002	Mexicali	Baja California
02003	Tecate	Baja California

INNER JOIN

- **Sintaxis 2: USING**

Sólo cuando el nombre del atributo referencial es igual en ambas tablas.

Select col1,col2,colN **FROM** tabla_1 **JOIN** tabla_2 **USING** (atributoReferencial);

Ejemplo: Seleccionar discos con disquera

```
select id_disco,titulo,precio,id_disquera,nombre as disquera, id_pais
from disco
join disquera using(id_disquera)
order by 1
limit 10;
```

id_disco	titulo	precio	id_disquera	disquera	id_pais
1	Love Goes	179.00	5	AVANT! Records	12
2	Trouble	215.04	6	MALIGNA	113
3	Atlas:Intelligence	150.00	7	DFA	112
4	Hidden Figures: The Album	365.14	8	Sub Pop	112
5	So Dark the Con of Man	474.04	9	R.I.P Society	207
6	21	200.00	1	Universal Music Group	112
7	When We All Fall Asleep, Where Do We Go?	250.00	2	Spinnin Records	15
8	Future Nostalgia	230.00	3	Sony Music	113
9	Prism	150.00	4	Holodeck Records	112
10	The Thrill Of It All	180.00	5	AVANT! Records	12

Nota: En todos los casos es equivalente usar JOIN o INNER JOIN

NATURAL JOIN

- Siempre que haya un atributo de referencia bien definido se podrá usar NATURAL JOIN.

```
select id_disco,titulo as disco,precio,id_disquera,nombre as
disquera,id_pais
from disco
natural join disquera
order by 1
limit 10;
```

id_disco	disco	precio	id_disquera	disquera	id_pais
1	Love Goes	179.00	5	AVANT! Records	12
2	Trouble	215.04	6	MALIGNA	113
3	Atlas:Intelligence	150.00	7	DFA	112
4	Hidden Figures: The Album	365.14	8	Sub Pop	112
5	So Dark the Con of Man	474.04	9	R.I.P Society	207
6	21	200.00	1	Universal Music Group	112
7	When We All Fall Asleep, Where Do We Go?	250.00	2	Spinnin Records	15
8	Future Nostalgia	230.00	3	Sony Music	113
9	Prism	150.00	4	Holodeck Records	112
10	The Thrill Of It All	180.00	5	AVANT! Records	12

INNER JOIN

- Ejemplo varios join

```
select id_disco,titulo,precio,id_disquera,d.nombre as
disquera,id_pais,p.nombre as pais
from disco
join disquera as d using(id_disquera)
join pais as p using(id_pais)
order by 1
limit 10;
```

id_disco	titulo	precio	id_disquera	disquera	id_pais	pais
1	Love Goes	179.00	5	AVANT! Records	12	Italia
2	Trouble	215.04	6	MALIGNA	113	Mexico
3	Atlas:Intelligence	150.00	7	DFA	112	Estados Unidos De America
4	Hidden Figures: The Album	365.14	8	Sub Pop	112	Estados Unidos De America
5	So Dark the Con of Man	474.04	9	R.I.P Society	207	Australia
6	21	200.00	1	Universal Music Group	112	Estados Unidos De America
7	When We All Fall Asleep, Where Do We Go?	250.00	2	Spinnin Records	15	Países Bajos
8	Future Nostalgia	230.00	3	Sony Music	113	Mexico
9	Prism	150.00	4	Holodeck Records	112	Estados Unidos De America
10	The Thrill Of It All	180.00	5	AVANT! Records	12	Italia

INNER JOIN

Ejemplo 2: Varios Join

Cuando los atributos coincidentes provienen de la misma tabla se pueden hacer joins a varias tablas de manera conjunta

```
select id_ticket, fecha, titulo, precio, cantidad
from detalle_ticket
join(ticket, disco) using(id_ticket, id_disco);
```

id_ticket	fecha	titulo	precio	cantidad
1	2019-01-23 00:00:00	Simulation Theory(Super Deluxe)	290.00	1
1	2019-01-23 00:00:00	After Laughter	200.00	2
2	2020-02-22 00:00:00	Abbey Road	192.00	3
3	2018-03-28 00:00:00	Último Acto	385.00	2
4	2020-11-20 00:00:00	21	200.00	1
4	2020-11-20 00:00:00	Carry On	350.00	1
4	2020-11-20 00:00:00	Evolve	210.00	2
4	2020-11-20 00:00:00	YHLQMDLG	220.00	2
5	2020-12-13 00:00:00	Tributo a José José	225.90	1
6	2019-05-03 00:00:00	Spring ep	205.00	1

INNER JOIN

Consultas equivalentes con varios join

1)

```
select id_ticket, fecha, titulo, precio, cantidad
from detalle_ticket
join(ticket, disco) using(id_ticket, id_disco);
```

2)

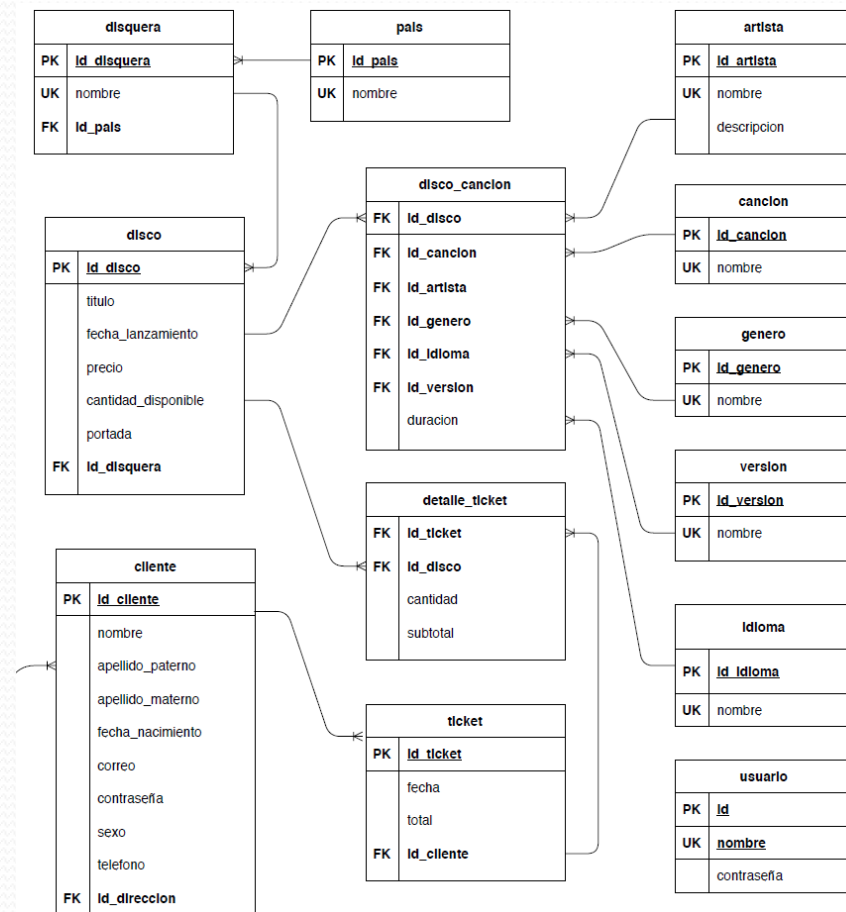
```
select id_ticket, fecha, titulo, precio, cantidad
from detalle_ticket
natural join(ticket, disco);
```

3)

```
select id_ticket, fecha, titulo, precio, cantidad
from disco
join detalle_ticket using(id_disco)
join ticket using(id_ticket);
```

4)

```
select id_ticket, fecha, titulo, precio, cantidad
from disco
natural join detalle_ticket
natural join ticket;
```



¿Preguntas?