

# Lenguaje SQL III

**Alex Di Genova**

**04/05/2022**

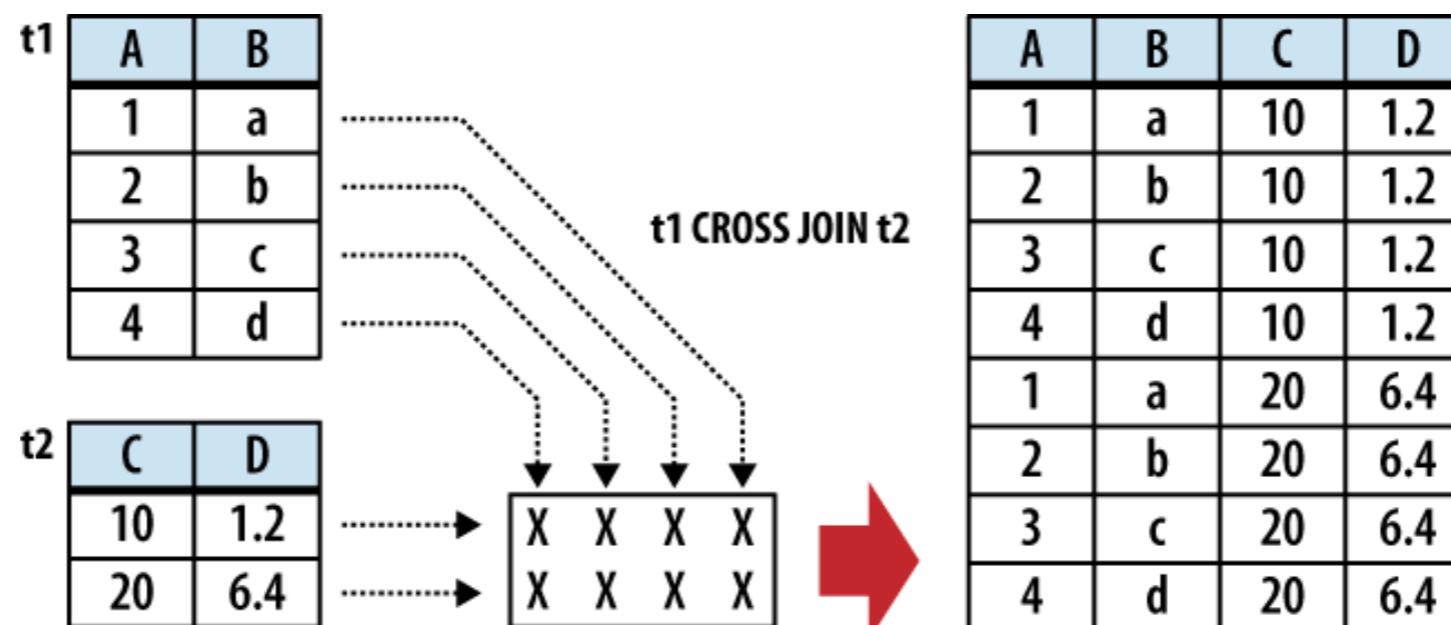
# Contenidos

- JOINs
- subconsultas
- Tecnicas avanzadas

# SQL

## CROSS JOIN

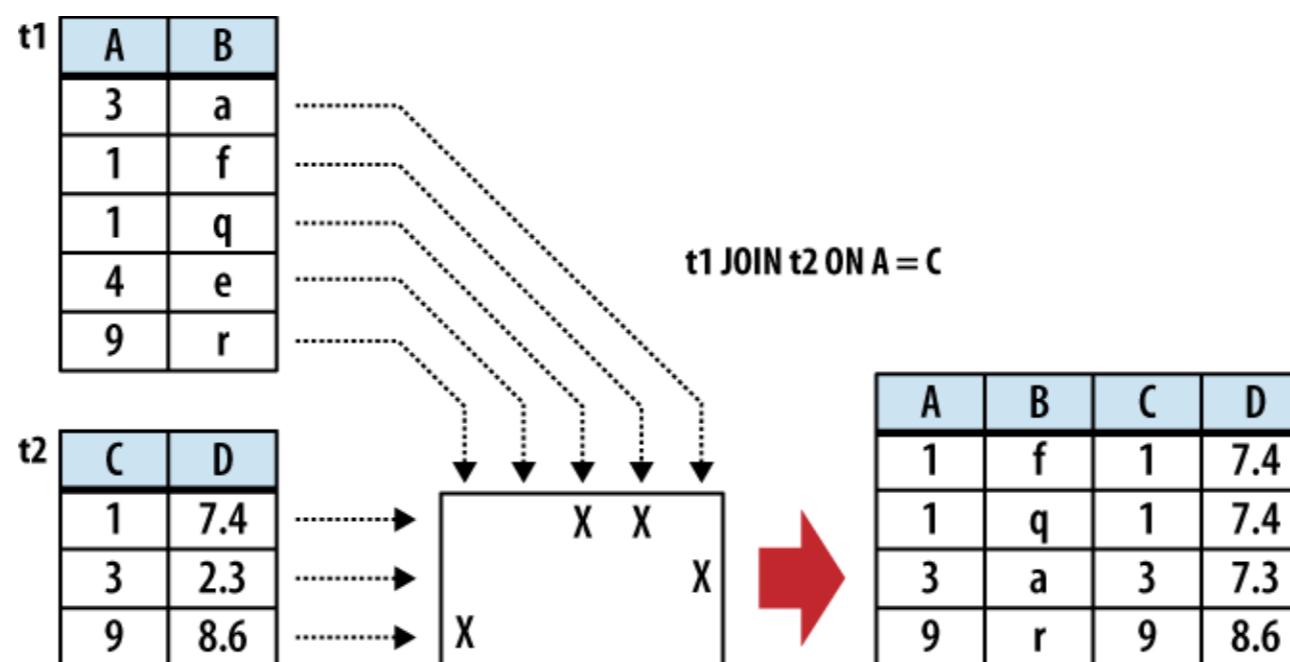
- **SELECT ... FROM t1 CROSS JOIN t2 ...**
- Hace coincidir cada fila de la primera tabla con cada fila de la segunda tabla.
- Si las tablas de entrada tienen x e y columnas, la tabla resultante tendrá x+y columnas.
- Si las tablas de entrada tienen n y m filas, la tabla resultante tendrá n x m filas (producto cartesiano)



# SQL

## INNER JOIN

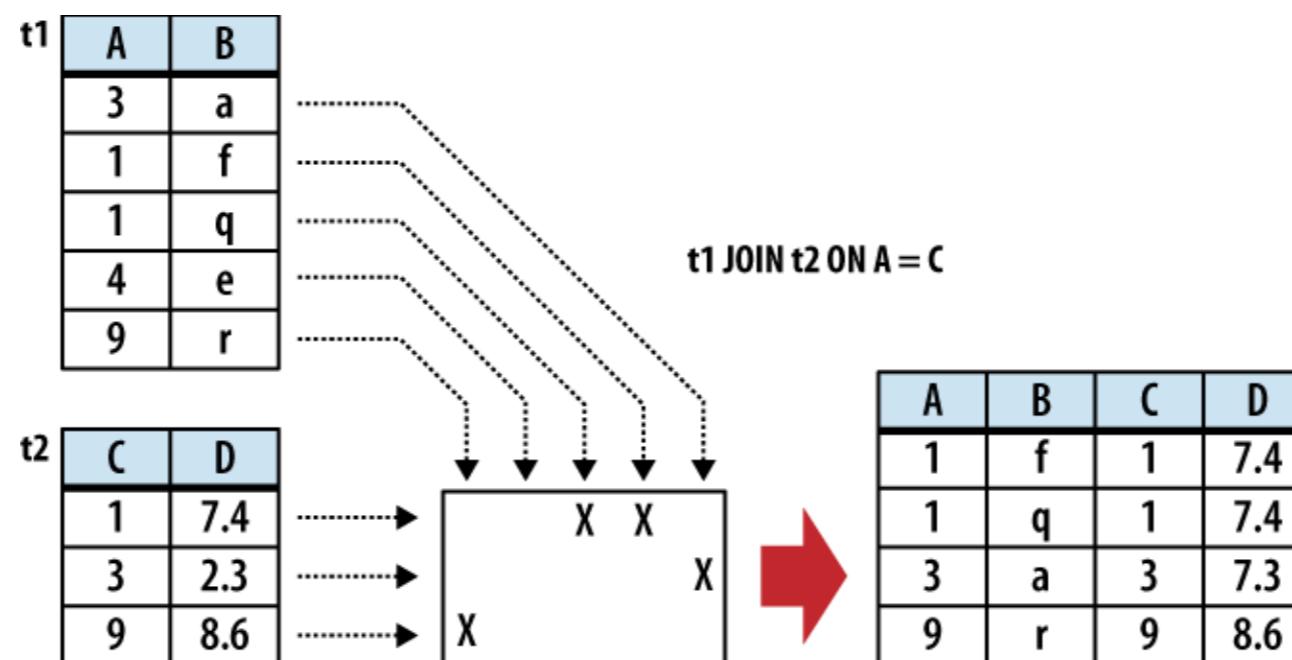
- **SELECT ... FROM t1 JOIN t2 ON expresion\_de\_condicion...**
- La expresión condición es usualmente utilizada para comparar dos columnas de dos tablas diferentes.
- Si las tablas de entrada tienen x e y columnas, la tabla resultante tendrá no más x+y columnas.
- Si las tablas de entrada tienen n y m filas, la tabla resultante tendrá [0,n x m] filas



# SQL

## INNER JOIN

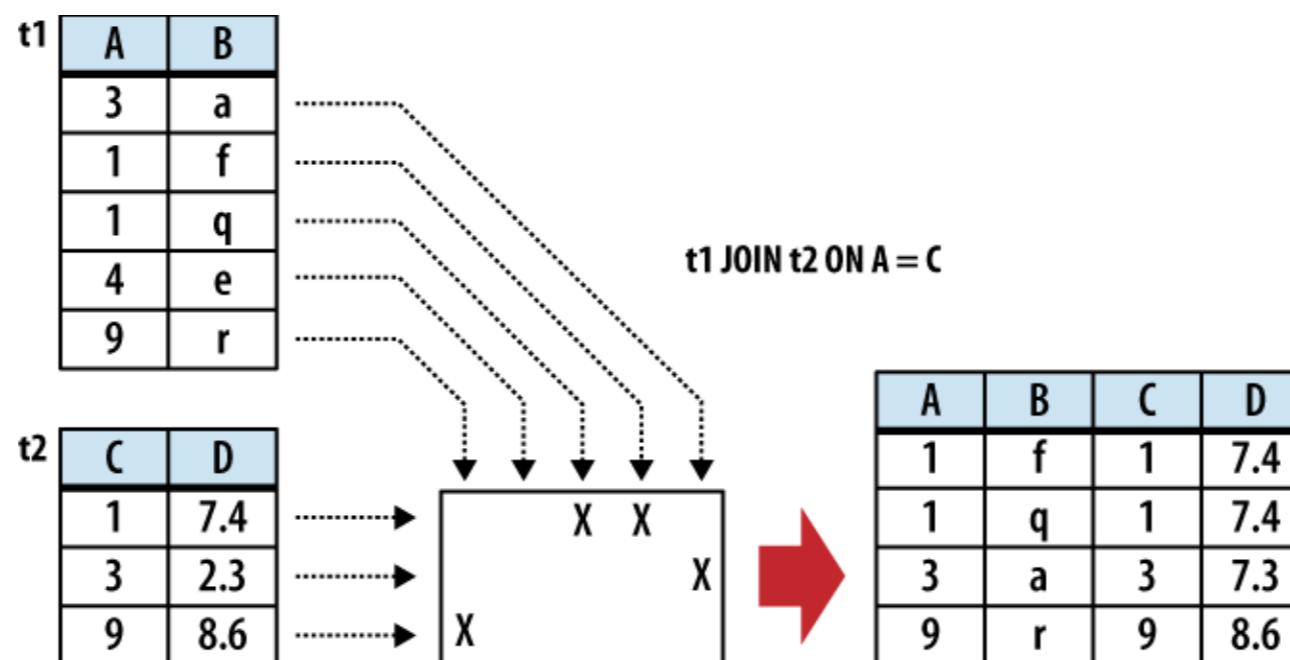
- **SELECT ... FROM t1 INNER JOIN t2 ON expresion\_de\_condicion...**
- **SELECT ... FROM t1 JOIN t2 ON expresion\_de\_condicion...**
- La expresión condición es usualmente utilizada para comparar dos columnas de dos tablas diferentes.
- Si las tablas de entrada tienen x e y columnas, la tabla resultante tendrá no más  $x+y$  columnas.
- Si las tablas de entrada tienen n y m filas, la tabla resultante tendrá  $[0,n \times m]$  filas
- **SELECT ... FROM alumno JOIN curso ON alumno.id=curso.alum\_id...**



# SQL

## INNER JOIN

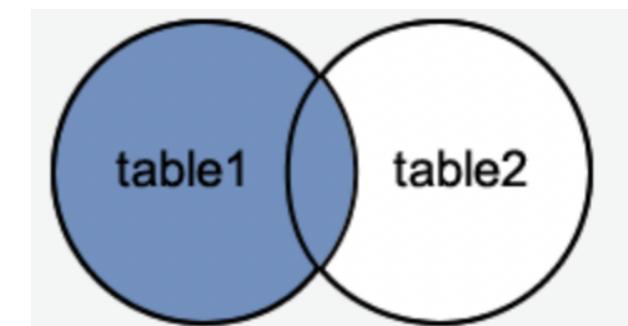
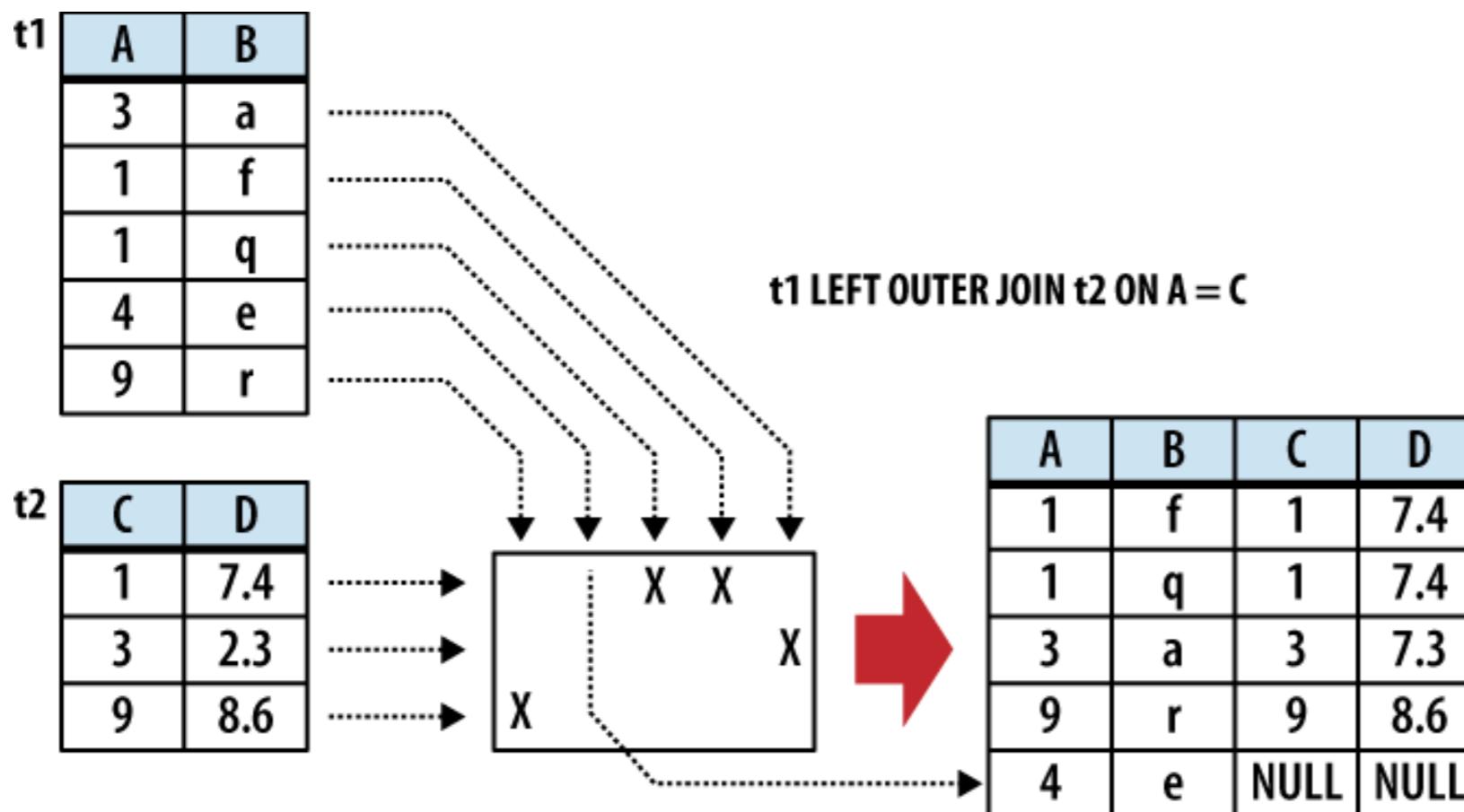
- **SELECT ... FROM alumno JOIN curso ON alumno.id=curso.alum\_id...**
- **SELECT ... FROM alumno JOIN curso USING (aid,...)**
  - El atributo debe existir en ambas tablas
- **SELECT ... FROM alumno NATURAL JOIN curso**
  - Join automatico en aid



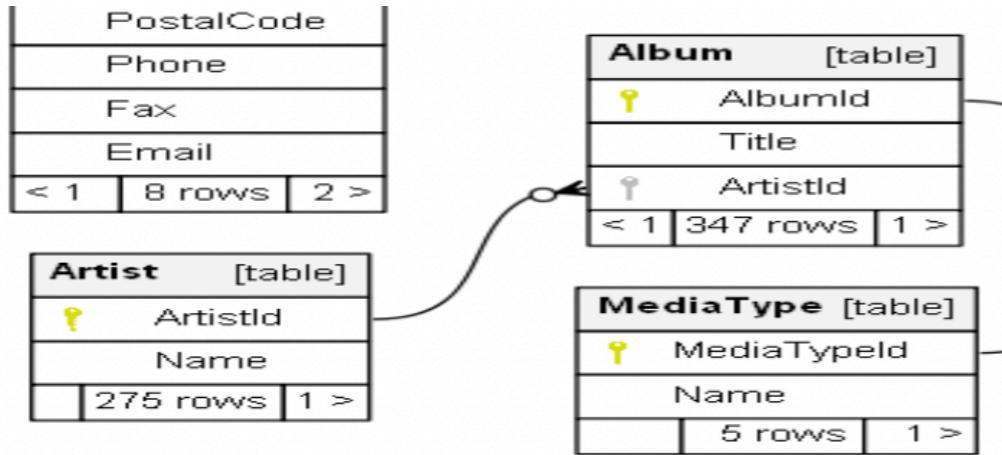
# SQL

## OUTER JOIN

- **SELECT ... FROM alumno LEFT OUTER JOIN curso ON alumno.id=curso.alum\_id...**
- Extension del INNER JOIN
- OUTER JOINS = LEFT, RIGHT, FULL (SQLite2 solo LEFT)
- Condiciones con ON, USING, NATURAL



# SQL JOIN Examples



```
select * from album CROSS JOIN Artist
limit 5;
```

```
select * from album JOIN Artist
ON album.artistId=Artist.artistId limit
5;
```

```
select * from album JOIN Artist USING
(artistId) limit 5;
```

```
select * from album NATURAL JOIN Artist
limit 5;
```

```

1 # CROSS Join example
2 %%sql
3 select * from album CROSS JOIN Artist limit 5;

```

\* sqlite:///content/chinook-database/ChinookDatabase/DataSources/  
Done.

AlbumId	Title	ArtistId	ArtistId_1	Name
1	For Those About To Rock We Salute You 1	1		AC/DC
1	For Those About To Rock We Salute You 1	2		Accept
1	For Those About To Rock We Salute You 1	3		Aerosmith
1	For Those About To Rock We Salute You 1	4		Alanis Morissette
1	For Those About To Rock We Salute You 1	5		Alice In Chains

```
1 #INNER Join
```

```
2 %%sql
3 select * from album JOIN Artist ON album.artistId=Artist.artistId
```

\* sqlite:///content/chinook-database/ChinookDatabase/DataSources/  
Done.

AlbumId	Title	ArtistId	ArtistId_1	Name
1	For Those About To Rock We Salute You 1	1		AC/DC
2	Balls to the Wall	2	2	Accept
3	Restless and Wild	2	2	Accept
4	Let There Be Rock	1	1	AC/DC
5	Big Ones	3	3	Aerosmith

```
1 #SELECT ... FROM alumno JOIN curso USING (aid,...)
```

```
2 %%sql
3 select * from album JOIN Artist USING (artistId) limit 5;
```

\* sqlite:///content/chinook-database/ChinookDatabase/DataSources/  
Done.

AlbumId	Title	ArtistId	Name
1	For Those About To Rock We Salute You 1		AC/DC
2	Balls to the Wall	2	Accept
3	Restless and Wild	2	Accept
4	Let There Be Rock	1	AC/DC
5	Big Ones	3	Aerosmith

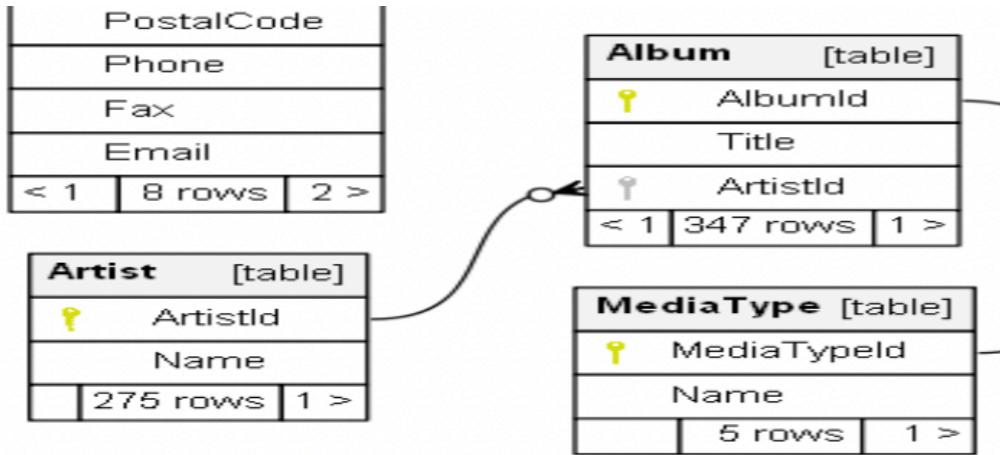
```
1 #SELECT ... FROM alumno NATURAL JOIN curso (artistId)
```

```
2 %%sql
3 select * from album NATURAL JOIN Artist limit 5;
```

\* sqlite:///content/chinook-database/ChinookDatabase/DataSources/  
Done.

AlbumId	Title	ArtistId	Name
1	For Those About To Rock We Salute You 1		AC/DC
2	Balls to the Wall	2	Accept
3	Restless and Wild	2	Accept
4	Let There Be Rock	1	AC/DC
5	Big Ones	3	Aerosmith

# SQL JOIN Examples



```
select * from album LEFT OUTER JOIN
Artist limit 5;
```

```
select * from album RIGHT OUTER JOIN
Artist limit 5;
```

```
select * from Artist LEFT OUTER JOIN
album limit 5;
```

```

1 %%sql
2 select * from album LEFT OUTER JOIN Artist USING (artistId) limit 10;

* sqlite:///content/chinook-database/ChinookDatabase/DataSources/Chinook
Done.

```

AlbumId	Title	ArtistId	Name
1	For Those About To Rock We Salute You	1	AC/DC
2	Balls to the Wall	2	Accept
3	Restless and Wild	2	Accept
4	Let There Be Rock	1	AC/DC
5	Big Ones	3	Aerosmith
6	Jagged Little Pill	4	Alanis Morissette
7	Facelift	5	Alice In Chains
8	Warner 25 Anos	6	Antônio Carlos Jobim
9	Plays Metallica By Four Cellos	7	Apocalyptica
10	Audioslave	8	Audioslave

```

1 %%sql
2 select * from album RIGHT OUTER JOIN Artist limit 10;

* sqlite:///content/chinook-database/ChinookDatabase/DataSources/Chinook
(sqlite3.OperationalError) RIGHT and FULL OUTER JOINS are not currently su
[SQL: select * from album RIGHT OUTER JOIN Artist limit 10;]
(Background on this error at: https://sqlalche.me/e/14/e3q8)

```

```

1 %%sql
2 select * from Artist LEFT OUTER JOIN album USING (artistId) limit 10;

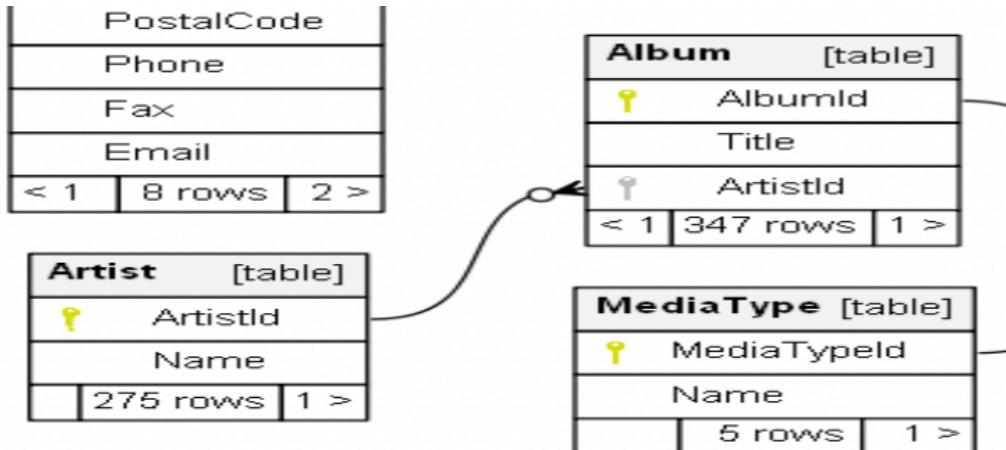
* sqlite:///content/chinook-database/ChinookDatabase/DataSources/Chinook
Done.

```

ArtistId	Name	AlbumId	Title
1	AC/DC	1	For Those About To Rock We Salute You
1	AC/DC	4	Let There Be Rock
2	Accept	2	Balls to the Wall
2	Accept	3	Restless and Wild
3	Aerosmith	5	Big Ones
4	Alanis Morissette	6	Jagged Little Pill
5	Alice In Chains	7	Facelift
6	Antônio Carlos Jobim	8	Warner 25 Anos
6	Antônio Carlos Jobim	34	Chill: Brazil (Disc 2)
7	Apocalyptica	9	Plays Metallica By Four Cellos

# SQL

## Notación JOINs



```

select * from album JOIN Artist
ON album.artistId=Artist.artistId limit
5;
  
```

```

select * from album JOIN Artist USING
(artistId) limit 5;
  
```

```

select * from album NATURAL JOIN Artist
limit 5;
  
```

```

select * from album, Artist WHERE
album.artistId=Artist.artistId limit 5;
  
```

```

3 select * from album JOIN Artist ON album.artistId=Artist.artistId limit 5;

* sqlite:///content/chinook-database/ChinookDatabase/DataSources/Chinook_Sqlite.sqlite
Done.

```

AlbumId	Title	ArtistId	ArtistId_1	Name
1	For Those About To Rock We Salute You	1		AC/DC
2	Balls to the Wall	2	2	Accept
3	Restless and Wild	2	2	Accept
4	Let There Be Rock	1	1	AC/DC
5	Big Ones	3	3	Aerosmith

```

1 #SELECT ... FROM alumno JOIN curso USING (aid,...)
2 %%sql
3 select * from album JOIN Artist USING (artistId) limit 5;
  
```

```

* sqlite:///content/chinook-database/ChinookDatabase/DataSources/Chinook_Sqlite.sqlite
Done.

```

AlbumId	Title	ArtistId	Name
1	For Those About To Rock We Salute You	1	AC/DC
2	Balls to the Wall	2	Accept
3	Restless and Wild	2	Accept
4	Let There Be Rock	1	AC/DC
5	Big Ones	3	Aerosmith

```

1 #SELECT ... FROM alumno NATURAL JOIN curso (artistId)
2 %%sql
3 select * from album NATURAL JOIN Artist limit 5;
  
```

```

* sqlite:///content/chinook-database/ChinookDatabase/DataSources/Chinook_Sqlite.sqlite
Done.

```

AlbumId	Title	ArtistId	Name
1	For Those About To Rock We Salute You	1	AC/DC
2	Balls to the Wall	2	Accept
3	Restless and Wild	2	Accept
4	Let There Be Rock	1	AC/DC
5	Big Ones	3	Aerosmith

```

1 %%sql
2 select * from album, Artist WHERE album.artistId=Artist.artistId limit 5;
  
```

```

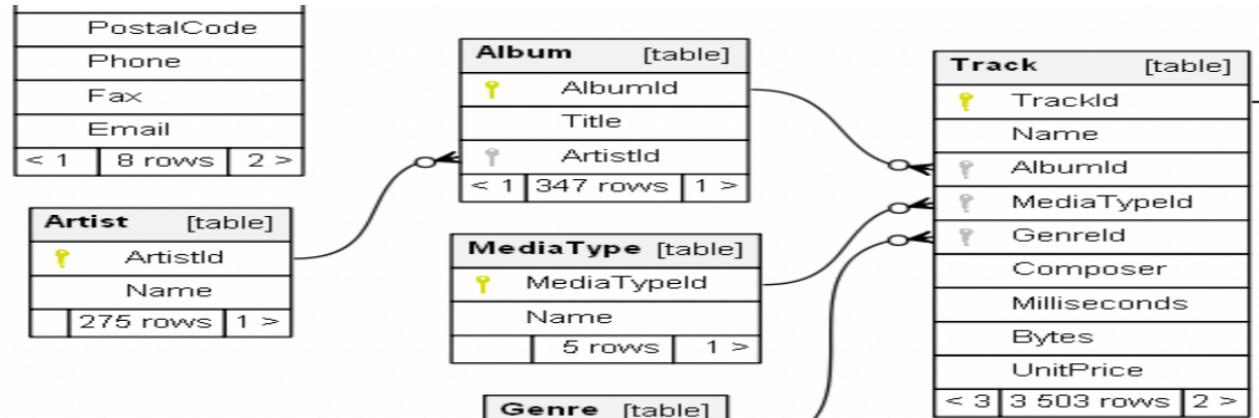
* sqlite:///content/chinook-database/ChinookDatabase/DataSources/Chinook_Sqlite.sqlite
Done.

```

AlbumId	Title	ArtistId	ArtistId_1	Name
1	For Those About To Rock We Salute You	1		AC/DC
2	Balls to the Wall	2	2	Accept

# SQL

## Multiples JOIN (JOIN & JOIN & JOIN...)



```
select * from album JOIN Artist USING (artistId) limit 5;
```

```
select * from album JOIN Artist USING (artistId) JOIN Track USING (albumid) limit 5;
```

```
[12] 1 %%sql
2 select * from album JOIN Artist USING (artistId) limit 5;

* sqlite:///content/chinook-database/ChinookDatabase/DataSources/Chinook_Sqlite.sqlite
Done.



| AlbumId | Title                                 | ArtistId | Name      |
|---------|---------------------------------------|----------|-----------|
| 1       | For Those About To Rock We Salute You | 1        | AC/DC     |
| 2       | Balls to the Wall                     | 2        | Accept    |
| 3       | Restless and Wild                     | 2        | Accept    |
| 4       | Let There Be Rock                     | 1        | AC/DC     |
| 5       | Big Ones                              | 3        | Aerosmith |


```

```
1 %%sql
2 select * from album JOIN Artist USING (artistId) JOIN Track USING (albumid) limit 5;

* sqlite:///content/chinook-database/ChinookDatabase/DataSources/Chinook_Sqlite.sqlite
Done.



| AlbumId | Title                                 | ArtistId | Name   | TrackId | Name_1                                  | MediaTypeId | GenreId | Composer                                                              | Milliseconds |   |
|---------|---------------------------------------|----------|--------|---------|-----------------------------------------|-------------|---------|-----------------------------------------------------------------------|--------------|---|
| 1       | For Those About To Rock We Salute You | 1        | AC/DC  | 1       | For Those About To Rock (We Salute You) | 1           | 1       | Angus Young, Malcolm Young, Brian Johnson                             | 343719       | 1 |
| 2       | Balls to the Wall                     | 2        | Accept | 2       | Balls to the Wall                       | 2           | 1       | None                                                                  | 342562       | 5 |
| 3       | Restless and Wild                     | 2        | Accept | 3       | Fast As a Shark                         | 2           | 1       | F. Baltes, S. Kaufman, U. Dirksneider & W. Hoffman                    | 230619       | 3 |
| 3       | Restless and Wild                     | 2        | Accept | 4       | Restless and Wild                       | 2           | 1       | F. Baltes, R.A. Smith-Diesel, S. Kaufman, U. Dirksneider & W. Hoffman | 252051       | 4 |
| 3       | Restless and Wild                     | 2        | Accept | 5       | Princess of the Dawn                    | 2           | 1       | Deaffy & R.A. Smith-Diesel                                            | 375418       | 6 |


```

# Subconsultas en SQL

## SELECT (SELECT)

- Consultas conteniendo subconsultas
  - Subconsultas pueden ser usadas casi en cualquier lugar de las consultas.
  - Select .. from where x IN ( select ... )

```
select name , AlbumId
from track
where AlbumId in (
    select AlbumId
    from album
    where title = "Californication")
limit 10;
```

```
1 #Obtener los nombres de todas las canciones del álbum "Californication".
2 %%sql
3 select name , AlbumId
4 from track
5 where AlbumId in (
6     select AlbumId
7     from album
8     where title = "Californication")
9 limit 5;
```

\* sqlite:///content/chinook-database/ChinookDatabase/DataSources/Chinook\_S  
Done.

Name	AlbumId
Around The World	195
Parallel Universe	195
Scar Tissue	195
Otherside	195
Get On Top	195

```
1 %%sql
2 select name , AlbumId
3 from track
4 where AlbumId not in (
5     select AlbumId
6     from album
7     where title = "Californication")
8 limit 5;
```

\* sqlite:///content/chinook-database/ChinookDatabase/DataSources/Chinook\_S  
Done.

Name	AlbumId
For Those About To Rock (We Salute You)	1
Balls to the Wall	2
Fast As a Shark	3
Restless and Wild	3
Princess of the Dawn	3

# SQL

## Técnicas avanzadas

- Subconsultas (IN)
  - **SELECT \* from X NATURAL JOIN Y;**
  - **SELECT \* from X NATURAL JOIN (SELECT \* FROM Y);**

```
select * from album NATURAL JOIN
Artist limit 5;
```

```
select * from album NATURAL JOIN
(select * from Artist) limit 5;
```

col [not] IN ( test1, test2, test3 )

col [not] IN ( SELECT c FROM t )

col [not] IN temp.in\_test

```
1 %%sql
2 select * from album  NATURAL JOIN Artist  limit 5;

* sqlite:///content/chinook-database/ChinookDatabase/DataSources/C
Done.
```

AlbumId	Title	ArtistId	Name
1	For Those About To Rock We Salute You	1	AC/DC
2	Balls to the Wall	2	Accept
3	Restless and Wild	2	Accept
4	Let There Be Rock	1	AC/DC
5	Big Ones	3	Aerosmith

```
1 %%sql
2 select * from album NATURAL JOIN (select * from Artist) limit 5;

* sqlite:///content/chinook-database/ChinookDatabase/DataSources/C
Done.
```

AlbumId	Title	ArtistId	Name
1	For Those About To Rock We Salute You	1	AC/DC
2	Balls to the Wall	2	Accept
3	Restless and Wild	2	Accept
4	Let There Be Rock	1	AC/DC
5	Big Ones	3	Aerosmith

# SQL

## Técnicas avanzadas

Combinación de múltiples select

SELECT .. FROM .. WHERE ...  
GROUP BY ... HAVING ...

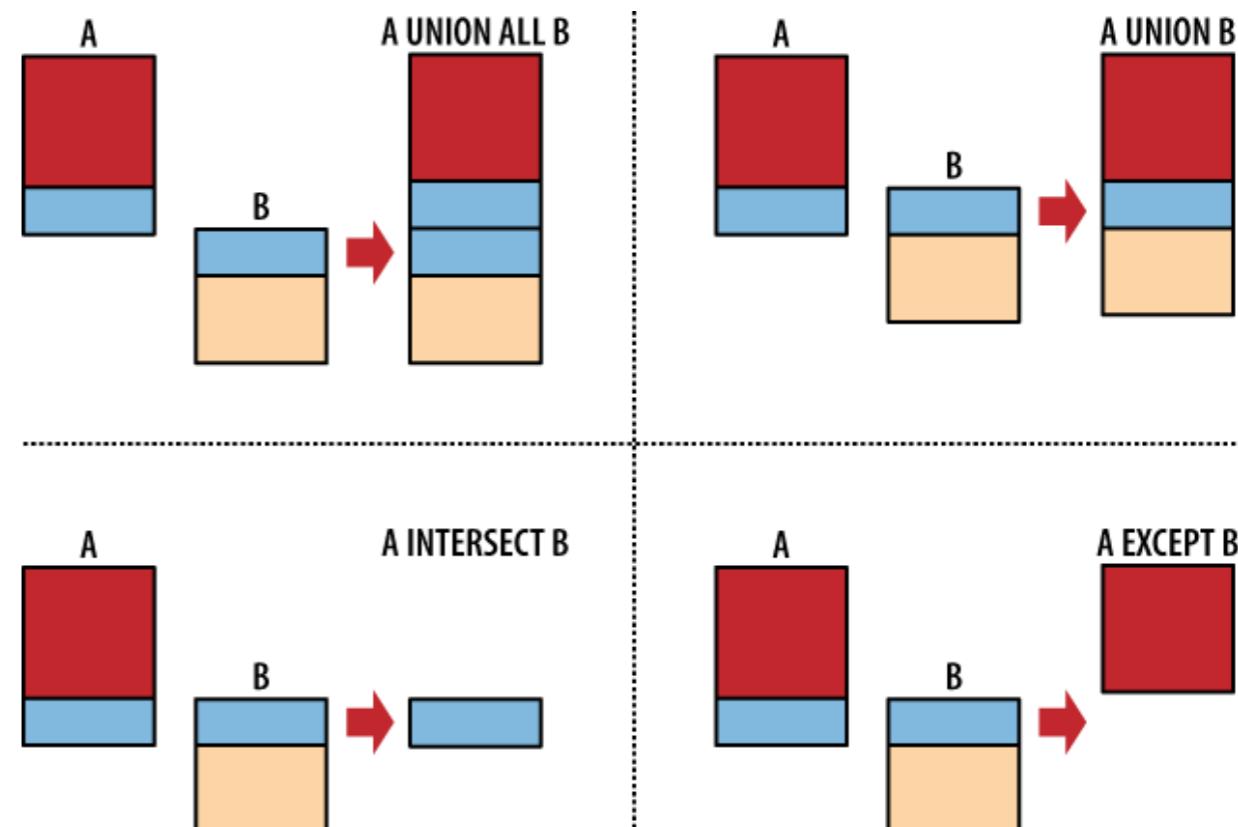
[Operador de mezcla]

SELECT .. FROM .. WHERE ...  
GROUP BY ... HAVING ...

[...]

ORDER BY ... LIMIT ....

UNION ALL, UNION, INTERSECT,  
EXCEPT



# Algebra relacional

## UNION, INTERSECTION, DIFFERENCE, PRODUCT

```

1 %%sql
2 drop table if exists R;
3 drop table if exists S;
4 -- Relación R --
5 CREATE TABLE R (
6 a_id INTEGER,
7 b INTEGER);
8 -- Relación S --
9 CREATE TABLE S (
10 a_id INTEGER,
11 b INTEGER);
12
13 -- valores en R --
14 INSERT INTO R(a_id,b) VALUES
15 (1,10),
16 (2,20),
17 (2,31),
18 (4,32);
19 -- valores en R --
20 INSERT INTO S(a_id,b) VALUES
21 (4,32),
22 (5,50),
23 (7,51);

```

$$R \cup S$$

```

1 %%sql
2 select * from R
3 UNION ALL
4 select * from S;

```

\* sqlite:///test.db  
Done.

**a\_id** **b**

1	10
2	20
2	31
4	32
4	32

5	50
7	51

```

1 %%sql
2 select * from R
3 EXCEPT
4 select * from S;

```

\* sqlite:///test.db  
Done.

**a\_id** **b**

1	10
2	20
2	31

$$R \cap S$$

```

1 %%sql
2 select * from R
3 INTERSECT
4 select * from S;

```

\* sqlite:///test.db  
Done.

**a\_id** **b**

4	32
---	----

$$R \times S$$

```

1 %%sql
2 select * from R,S;

```

\* sqlite:///test.db  
Done.

**a\_id** **b** **a\_id\_1** **b\_1**

1	10	4	32
1	10	5	50
1	10	7	51
2	20	4	32
2	20	5	50
2	20	7	51
2	31	4	32
2	31	5	50
2	31	7	51
4	32	4	32
4	32	5	50
4	32	7	51

$$R - S$$

```

1 %%sql
2 select * from R
3 EXCEPT
4 select * from S;

```

\* sqlite:///test.db  
Done.

**a\_id** **b**

1	10
2	20
2	31

# Complex SQL queries

## Que hacen las siguientes consultas?

```
%%sql
SELECT ar.Name AS Artist, COUNT(DISTINCT(tr.TrackId)) AS TrackssSold, SUM(ii.UnitPrice) AS Revenue
FROM artist ar
INNER JOIN album al ON ar.ArtistId = al.ArtistId
INNER JOIN track tr ON al.AlbumId = tr.AlbumId
INNER JOIN invoiceline ii ON tr.TrackId = ii.TrackId
INNER JOIN invoice i ON ii.InvoiceId = i.InvoiceId
WHERE strftime('%Y', i.InvoiceDate) = '2013'
GROUP BY ar.Name
ORDER BY TrackssSold DESC
LIMIT 5;
```

Identifica los 5 artistas más populares entre los clientes de la tienda de música de Chinook en 2013.

```
%%sql
SELECT c.FirstName || ' ' || c.LastName AS CustomerName,
       (SELECT COUNT(*) FROM invoiceline WHERE InvoiceId IN
        (SELECT InvoiceId FROM invoice WHERE CustomerId = c.CustomerId)) AS TotalTracksPurchased,
       (SELECT COUNT(DISTINCT AlbumId) FROM
        (SELECT AlbumId FROM invoiceline INNER JOIN track ON invoiceline.TrackId = track.TrackId
         WHERE InvoiceId IN (SELECT InvoiceId FROM invoice WHERE CustomerId = c.CustomerId))) AS
TotalAlbumsPurchased,
       (SELECT SUM(Total) FROM invoice WHERE CustomerId = c.CustomerId) AS TotalSpent
FROM customer c;
```

# SQL

## SELECT

- **SELECT** pipeline

```
SELECT [DISTINCT] select_header
      FROM tablas
      WHERE expresion_filtrado
      GROUP BY expresion_de_agrupamiento
      HAVING expresion_filtrado
      ORDER BY expresion_orden
      LIMIT contador
      OFFSET indice
```

# Practicar en GoogleColab!!!

The screenshot shows a Google Colab notebook interface. The title bar indicates the file is 'SQL\_IV\_chinook\_db.ipynb'. The notebook contains several code cells and their outputs:

- SQL con Chinook**: A section containing a code cell with a play button and the following SQL query:

```
1 # CROSS Join example
2 %%sql
3 select * from album CROSS JOIN Artist limit 5;
```

The output shows the results of the cross join between the 'album' and 'Artist' tables.
- Joins**: A section containing a code cell with a play button and the following SQL query:

```
* sqlite:///content/chinook-database/ChinookDatabase/DataSources/Chinook_Sqlite.sqlite
Done.
AlbumId      Title      ArtistId ArtistId_1      Name
1   For Those About To Rock We Salute You 1      1      AC/DC
1   For Those About To Rock We Salute You 1      2      Accept
1   For Those About To Rock We Salute You 1      3      Aerosmith
1   For Those About To Rock We Salute You 1      4      Alanis Morissette
1   For Those About To Rock We Salute You 1      5      Alice In Chains
```
- #INNER Join**: A code cell with a play button and the following SQL query:

```
1 #INNER Join
2 %%sql
3 select * from album JOIN Artist ON album.artistId=Artist.artistId limit 5;
```

The output shows the results of an inner join between the 'album' and 'Artist' tables.
- #SELECT ... FROM alumno JOIN curso USING (aid,...)**: A code cell with a play button and the following SQL query:

```
1 #SELECT ... FROM alumno JOIN curso USING (aid,...)
2 %%sql
```

The bottom status bar shows '0s completed at 9:43 AM'.

# Practicar

[https://github.com/adigenova/uohdb/blob/main/code/SQL\\_IV\\_chinook\\_db.ipynb](https://github.com/adigenova/uohdb/blob/main/code/SQL_IV_chinook_db.ipynb)

# Consultas?

Consultas o comentarios?

Muchas gracias