

## **AUTOGUIA 2**

### **MODELOS Y BASES DE DATOS**

#### **ESTUDIANTES:**

**JUAN SEBASTIAN PUENTES JULIO**

**CHRISTIAN ALFONSO ROMERO**

#### **DOCENTE:**

**MARIA IRMA ROZO**

**ESCUELA COLOMBIANA DE INGENIERÍA JULIO GARAVITO**

**07/09/2024**

## INVESTIGACIÓN

### A. NULL

#### 1. ¿Qué significa?

- NULL es un literal que representa una referencia nula que no hace referencia a ningún objeto.

#### 2. ¿Resultado de operarlo con los diferentes tipos de operadores: aritméticos, lógicos y de comparación?

- Comparación: IS NOT NULL devuelve una tabla con los registros en donde no haya un dato en blanco o no existente, mientras que, IS NULL devuelve la misma con datos desconocidos.
- Aritméticos: Cualquier operación con NULL resulta en NULL.
- Lógico: El resultado de una operación lógica es True o False o Null, ya que lo que hace es la validación de si se cumple o no.

### B. JUNTA

#### 1. ¿Cuáles son las diferencias entre junta interna y externa?

- La junta interna devuelve solo los registros que coinciden en una tabla A y una tabla B, mientras que, la junta externa devuelve todas las filas cuando hay una coincidencia en una u otra tabla.

#### 2. ¿Qué opciones se tienen para la junta interna ?

- Se tiene tres opciones:  
JOIN (INNER JOIN): Devuelve solo los registros que coinciden en una tabla A y una tabla B.  
NATURAL JOIN: Se combinan columnas que tienen el mismo nombre en ambas tablas.  
CROSS JOIN (Producto Cartesiano): Combina ambas tablas sin importar si cumplen una condición o no. Este join toma cada fila de una tabla y la une a cada fila de otra tabla.

#### 3. ¿Qué opciones se tienen para la junta externa?

- Se tienen tres opciones:

LEFT JOIN: Devuelve todas las filas de la tabla izquierda y las coincidentes de la tabla derecha.

RIGHT JOIN: Devuelve todas las filas de la tabla derecha y las coincidentes de la tabla izquierda.

FULL JOIN: Devuelve todas las filas cuando hay una coincidencia en una u otra tabla

## PRÁCTICA

### JOIN

1. *Modify it to show the matchid and player name for all goals scored by Germany. To identify German players, check for: teamid = "GER"*

```
SELECT matchid, player
FROM goal
WHERE teamid = "GER"
```

2. *Show id, stadium, team1, team2 for just game 1012*

```
SELECT id,stadium,team1,team2
FROM game
WHERE id = 1012
```

3. *Modify it to show the player, teamid, stadium and mdate for every German goal.*

```
SELECT player,teamid,stadium, mdate
FROM game JOIN goal ON (id=matchid AND teamid = "GER")
```

4. *Show the team1, team2 and player for every goal scored by a player called Mario*

```
SELECT team1,team2,player
FROM goal JOIN game ON (player LIKE "Mario%" AND matchid = id)
```

5. *Show player, teamid, coach, gtime for all goals scored in the first 10 minutes  
gtime<=10*

```
SELECT player, teamid, coach, gtime
FROM goal JOIN eteam ON (teamid = id)
WHERE gtime<=10
```

6. *List the dates of the matches and the name of the team in which 'Fernando Santos' was the team1 coach.*

```
SELECT mdate, teamname
FROM game JOIN eteam ON (team1 = eteam.id)
WHERE coach = "Fernando Santos"
```

7. *List the player for every goal scored in a game where the stadium was 'National Stadium, Warsaw'*

```
SELECT player
FROM game JOIN goal ON (matchid = id)
WHERE stadium = 'National Stadium, Warsaw'
```

8. *Instead show the name of all players who scored a goal against Germany.*

```
SELECT DISTINCT player
FROM goal JOIN game ON matchid = id
WHERE (team1 = 'GER' OR team2 = 'GER') AND (teamid != 'GER')
```

9. *Show teamname and the total number of goals scored.*

```
SELECT teamname, COUNT(gtime)
FROM eteam JOIN goal ON id=teamid
GROUP BY teamname
```

10. *Show the stadium and the number of goals scored in each stadium.*

```
SELECT stadium, COUNT(gtime)
FROM game JOIN goal ON (id = matchid)
GROUP BY stadium
```

11. *For every match involving 'POL', show the matchid, date and the number of goals scored.*

```
SELECT matchid, mdate, COUNT(gtime)
FROM game JOIN goal ON matchid = id
WHERE (team1 = 'POL' OR team2 = 'POL')
GROUP BY matchid
```

12. *For every match where 'GER' scored, show matchid, match date and the number of goals scored by 'GER'*

```
SELECT matchid, mdate, COUNT(gtime)
FROM goal JOIN game ON matchid = id
WHERE teamid = "GER"
GROUP BY matchid
```

13. *List every match with the goals scored by each team as shown. This will use "CASE WHEN" which has not been explained in any previous exercises.*

```
SELECT mdate, team1,
SUM(CASE WHEN teamid=team1 THEN 1 ELSE 0 END) score1,
team2,
SUM(CASE WHEN teamid=team2 THEN 1 ELSE 0 END) score2
FROM game LEFT JOIN goal ON matchid = id
GROUP BY mdate, team1, team2
ORDER BY mdate, matchid, team1, team2
```

## JOIN QUIZ

2	
2	
1	
2	

Netherlands	2
Poland	2
Republic of Ireland	1
Ukraine	2

Netherlands	
Poland	
Republic of Ireland	
Ukraine	

Poland	76
--------	----

Republic of Ireland	1
---------------------	---

Score the test

Your score is: 7 out of 7

## MORE JOIN

1. *List the films where the yr is 1962 [Show id, title]*

```
SELECT id, title
FROM movie
WHERE yr=1962
```

2. *Give the year of 'Citizen Kane'.*

```
SELECT yr
FROM movie
WHERE title = 'Citizen Kane'
```

3. *List all of the Star Trek movies, including the id, title and yr (all of these movies include the words Star Trek in the title). Order results by year.*

```
SELECT id, title, yr
FROM movie
WHERE title LIKE "Star Trek%"
ORDER BY yr
```

4. *What id number does the actor 'Glenn Close' have?*

```
SELECT id
FROM actor
WHERE name = 'Glenn Close'
```

5. *What is the id of the film 'Casablanca'?*

```
SELECT id
FROM movie
WHERE title = "CasaBlanca"
```

6. *Obtain the cast list for 'Casablanca'.*

```
SELECT name
FROM casting JOIN actor ON (actorid = id)
WHERE movieid = 11768
```

7. *Obtain the cast list for the film 'Alien'?*

```
SELECT name
FROM movie JOIN casting ON (id = movieid) JOIN actor ON
(casting.actorid = actor.id)
WHERE title = 'Alien'
```

8. *List the films in which 'Harrison Ford' has appeared*

```
SELECT title
FROM movie JOIN casting ON (id = movieid) JOIN actor ON casting.actorid =
actor.id
WHERE actor.name = "Harrison Ford"
```

9. *List the films where 'Harrison Ford' has appeared - but not in the starring role. [Note: the ord field of casting gives the position of the actor. If ord=1 then this actor is in the starring role]*

```
SELECT title
FROM movie JOIN casting ON (id = movieid) JOIN actor ON casting.actorid =
actor.id
WHERE actor.name = "Harrison Ford" AND NOT ord = 1
```

10. *List the films together with the leading star for all 1962 films.*

```
SELECT title, actor.name
FROM movie JOIN casting ON (id = movieid) JOIN actor ON casting.actorid =
actor.id
WHERE ord = 1 AND yr = 1962
```

11. *Which were the busiest years for 'Rock Hudson', show the year and the number of movies he made each year for any year in which he made more than 2 movies.*

```
SELECT yr,COUNT(title) FROM
movie JOIN casting ON movie.id=movieid
JOIN actor ON actorid=actor.id
WHERE name='Rock Hudson'
GROUP BY yr
HAVING COUNT(title) > 2
```

*12. List the film title and the leading actor for all of the films 'Julie Andrews' played in.*

```
SELECT title, name
FROM movie JOIN casting ON (movie.id = movieid AND ord = 1)
JOIN actor ON actorid = actor.id
WHERE movie.id IN (SELECT movieid FROM casting WHERE actorid IN
(SELECT id FROM actor WHERE name = "Julie Andrews"))
```

*13. Obtain a list, in alphabetical order, of actors who've had at least 15 starring roles.*

```
SELECT name
FROM actor JOIN casting ON (actor.id = actorid AND ord = 1)
GROUP BY name
HAVING COUNT(ord) >= 15
ORDER BY name
```

*14. List the films released in the year 1978 ordered by the number of actors in the cast, then by title.*

```
SELECT title, COUNT(title)
FROM movie JOIN casting ON id = movieid
WHERE yr = 1978
GROUP BY title
ORDER BY COUNT(title) DESC, title
```



**15. List all the people who have worked with 'Art Garfunkel'.**

```
SELECT name
FROM actor JOIN casting ON (actorid = id) JOIN movie ON (movie.id = movieid)
WHERE name != "Art Garfunkel" AND (movieid IN (
SELECT movieid
FROM casting JOIN actor ON actorid = id
WHERE name = 'Art Garfunkel'))
```

## JOIN QUIZ 2

A Bronx Tale	3
Bang the Drum Slowly	3
Limitless	3

Table-D

A Bronx Tale
Bang the Drum Slowly
Limitless

Table-E

A Bronx Tale	Robert De Niro	1993
Bang the Drum Slowly	Robert De Niro	1973
Limitless	Robert De Niro	2011

Score the test

Your score is: 7 out of 7

## NULL

**1. List the teachers who have NULL for their department.**

```
SELECT name
FROM teacher
WHERE dept IS NULL
```

**2. Note the INNER JOIN misses the teachers with no department and the departments with no teacher.**

```
SELECT teacher.name, dept.name
FROM teacher INNER JOIN dept ON (teacher.dept=dept.id)
```

3. *Use a different JOIN so that all teachers are listed*

```
SELECT teacher.name, dept.name  
FROM teacher LEFT JOIN dept ON dept = dept.id
```

4. *Use a different JOIN so that all departments are listed.*

```
SELECT teacher.name, dept.name  
FROM teacher RIGHT JOIN dept ON dept = dept.id
```

5. *Use COALESCE to print the mobile number. Use the number '07986 444 2266' if there is no number given. Show teacher name and mobile number or '07986 444 2266'*

```
SELECT name, COALESCE(mobile, '07986 444 2266')  
FROM teacher
```

6. *Use the COALESCE function and a LEFT JOIN to print the teacher name and department name. Use the string 'None' where there is no department.*

```
SELECT teacher.name, COALESCE(dept.name, "None")  
FROM teacher LEFT JOIN dept ON ( dept = dept.id)
```

7. *Use COUNT to show the number of teachers and the number of mobile phones.*

```
SELECT COUNT(name), COUNT(mobile)  
FROM teacher
```

8. Use **COUNT** and **GROUP BY dept.name** to show each department and the number of staff. Use a **RIGHT JOIN** to ensure that the Engineering department is listed.

```
SELECT dept.name, COUNT(teacher.name)
FROM teacher RIGHT JOIN dept ON (dept.id = dept)
GROUP BY dept.name
```

9. Use **CASE** to show the name of each teacher followed by 'Sci' if the teacher is in dept 1 or 2 and 'Art' otherwise.

```
SELECT name, CASE WHEN dept = 1 OR dept = 2 THEN "Sci" ELSE "Art"
END
FROM teacher
```

10. Use **CASE** to show the name of each teacher followed by 'Sci' if the teacher is in dept 1 or 2, show 'Art' if the teacher's dept is 3 and 'None' otherwise.

```
SELECT name, CASE
WHEN dept = 1 OR dept = 2 THEN "Sci"
WHEN dept = 3 THEN "Art"
ELSE "None" END
FROM teacher
```

## NULL QUIZ

Throd	1
Splint	1
Spiregrain	0
Cutflower	0
Deadyawn	0

Score the test

Your score is: 5 out of 6

Category: Quizzes

## SELF JOIN

1. *How many stops are in the database.*

```
SELECT COUNT(name)
FROM stops
```

2. *Find the id value for the stop 'Craiglockhart'*

```
SELECT id
FROM stops
WHERE name = 'Craiglockhart'
```

3. *Give the id and the name for the stops on the '4' 'LRT' service.*

```
SELECT id, name
FROM stops JOIN route ON id = stop
WHERE num = '4'
```

4. *The query shown gives the number of routes that visit either London Road (149) or Craiglockhart (53). Run the query and notice the two services that link these stops have a count of 2. Add a HAVING clause to restrict the output to these two routes.*

```
SELECT company, num, COUNT(*)
FROM route WHERE stop=149 OR stop=53
GROUP BY company, num
HAVING COUNT(*) = 2
```

5. *Execute the self join shown and observe that b.stop gives all the places you can get to from Craiglockhart, without changing routes. Change the query so that it shows the services from Craiglockhart to London Road.*

```
SELECT a.company, a.num, a.stop, b.stop
FROM route a JOIN route b ON
(a.company=b.company AND a.num=b.num)
WHERE a.stop=53 AND b.stop = 149
```

6. *The query shown is similar to the previous one, however by joining two copies of the stops table we can refer to stops by name rather than by number. Change the query so that the services between 'Craiglockhart' and 'London Road' are shown. If you are tired of these places try 'Fairmilehead' against 'Tollcross'*

```
SELECT a.company, a.num, stopa.name, stopb.name
FROM route a JOIN route b ON
(a.company=b.company AND a.num=b.num)
JOIN stops stopa ON (a.stop=stopa.id)
JOIN stops stopb ON (b.stop=stopb.id)
WHERE stopa.name= 'Craiglockhart' AND stopb.name = 'London Road'
```

7. *Give a list of all the services which connect stops 115 and 137 ('Haymarket' and 'Leith')*

```
SELECT DISTINCT a.company, a.num
FROM route a JOIN route b ON (a.company = b.company AND a.num = b.num)
WHERE a.stop = 115 AND b.stop = 137
```

8. *Give a list of the services which connect the stops 'Craiglockhart' and 'Tollcross'*

```

SELECT a.company, a.num
FROM route a JOIN route b ON (a.company = b.company AND a.num = b.num)
JOIN stops stopa ON (stopa.id = a.stop)
JOIN stops stopb ON (stopb.id = b.stop)
WHERE stopa.name = 'Craiglockhart' AND stopb.name = 'Tollcross'

```

9. *Give a distinct list of the stops which may be reached from 'Craiglockhart' by taking one bus, including 'Craiglockhart' itself, offered by the LRT company. Include the company and bus no. of the relevant services.*

```

SELECT stopb.name, a.company, a.num
FROM route a JOIN route b ON (a.company = b.company AND a.num = b.num)
JOIN stops stopa ON (stopa.id = a.stop)
JOIN stops stopb ON (stopb.id = b.stop)
WHERE stopa.name = 'Craiglockhart' AND a.company= "LRT"

```

10. *Find the routes involving two buses that can go from Craiglockhart to Lochend. Show the bus no. and company for the first bus, the name of the stop for the transfer, and the bus no. and company for the second bus.*

- No lo pudimos realizar debido a la dificultad del JOIN.

## SELF JOIN QUIZ

```
SELECT a.company, a.num, stopa.name, stopb.name
FROM route a JOIN route b ON (a.company=b.company AND a.num=b.num)
JOIN stops stopa ON (a.stop=stopa.id)
JOIN stops stopb ON (b.stop=stopb.id)
WHERE stopa.name='Tollcross'
```

```
SELECT a.company, a.num, stopa.name, stopb.name
FROM route a JOIN route b ON (a.company=b.company AND a.num=b.num)
JOIN stops stopa ON (a.stop=stopa.id)
JOIN stops stopb ON (b.stop=stopb.id)
WHERE stopz.name='Tollcross'
```

Score the test

Your score is: 3 out of 3

Category: Quizzes

C. Propongan preguntas que cumplan los siguientes requerimientos.

- **5 consultas: una para cada operador de conjuntos**

- UNION

- ¿Cuáles son los nombres de las bandas y los músicos?

```
SELECT m_name
FROM musician
UNION
SELECT band_name
FROM band
```

Submit SQL

restore default

Result:

m_name
Fred Bloggs
John Smith
Helen Smyth
Harriet Smithson
James First
Theo Mengel
Sue Little

- UNION ALL

- ¿Cuáles son las fechas relevantes en la música?

```
SELECT con_date
FROM concert
UNION ALL
SELECT born
FROM musician
UNION ALL
```

Submit SQL

restore default

Result:

con_date
Fri, 06 Jan 1995 00:00:00 GMT
Wed, 08 May 1996 00:00:00 GMT
Sat, 03 Jun 1995 00:00:00 GMT
Sat, 20 Sep 1997 00:00:00 GMT
Tue, 21 Feb 1995 00:00:00 GMT
Mon, 12 Apr 1993 00:00:00 GMT
Fri, 14 May 1993 00:00:00 GMT

○ INTERSECT

■ ¿Qué géneros de música comparten las bandas y los intérpretes?

```
SELECT band_type
FROM band
INTERSECT
SELECT perf_type
FROM performer
```

Submit SQL

restore default

Result:

band_type
classical
jazz

○ EXTRACT

■ ¿En qué años nacieron los músicos?



```
SELECT EXTRACT(YEAR FROM born)
FROM musician
```

Submit SQL

restore default

Result:

EXTRACT(YEAR ..
1948
1950
1948
1909
1965
1948
1945

- IN

- ¿Intérpretes e instrumentos que hacen jazz?

```
SELECT *
FROM performer
WHERE perf_type IN ("jazz")
```

Submit SQL

restore default

Result:

perf_no	perf_is	instrument	perf_type
3	6	banjo	jazz
5	12	guitar	jazz
7	16	trumpet	jazz
9	20	bass	jazz
10	2	flute	jazz
11	20	cornet	jazz
12	6	violin	jazz

**6 consultas: una para cada operador de junta**

- JOIN

- ¿En qué ciudad y país viven los músicos?

```
SELECT m_name, place_town, place_country
FROM musician
JOIN place ON living_in = place_no
```

Submit SQL

restore default

Result:

m_name	place_town	place_country
Fred Bloggs	Edinburgh	Scotland
John Smith	New York	USA
Helen Smyth	Birmingham	England
Harriet Smithson	Glasgow	Scotland
James First	London	England
Theo Mengel	Manchester	England
Sue Little	Amsterdam	Netherlands

- NATURAL JOIN
  - No se puede hacer el ejercicio porque en la base de datos de musician's no contamos con 2 tabla que tengan el mismo nombre de columna para poder efectuar la búsqueda
- CROSS JOIN
  - ¿Todos los posibles cantantes de cada canción?

```
SELECT m_name, c_title
FROM musician
CROSS JOIN composition
```

Submit SQL

restore default

Result:

m_name	c_title
Fred Bloggs	Opus 1
Fred Bloggs	Here Goes
Fred Bloggs	Valiant Knight
Fred Bloggs	Little Piece
Fred Bloggs	Simple Song
Fred Bloggs	Little Swing Song
Fred Bloggs	Fast Journey

- LEFT JOIN
  - ¿A qué bandas pertenece cada músico?

```
SELECT player, band_name
FROM band
LEFT JOIN plays_in ON band_id = band_no
```

Submit SQL

restore default

Result:

player	band_name
1	ROP
1	Oh well
3	ROP
4	ROP
4	Oh well
5	ROP
6	ROP

- RIGHT JOIN

- ¿En donde quedan los salones en los que se hicieron los conciertos?

```
SELECT concert_venue, place_town
FROM concert
LEFT JOIN place ON concert_in = place_no
```

Submit SQL

restore default

Result:

concert_venue	place_town
Bridgewater Hall	Manchester
Bridgewater Hall	Manchester
Usher Hall	Edinburgh
Assembly Rooms	Edinburgh
Festspiel Haus	Salzburg
Royal Albert Hall	London
Concertgebouw	Amsterdam

- FULL JOIN

- ¿Qué compositor compuso cada canción?

```
SELECT cmpr_no, c_title
FROM composition
FULL JOIN has_composed ON cmpr_no=c_no
```

Submit SQL

restore default

Result:

cmpr_no	c_title
1	Opus 1
1	Simple Love Song
2	Fast Drumming
3	Here Goes
3	Blue Roses
3	Velvet Rain
3	Long Rythms

2 consultas: una para cada operador de desconocido

- ISNULL
  - Dime cuando el músico sigue vivo

```
SELECT m_name, born, ISNULL(died)
FROM musician
```

Submit SQL

restore default

Result:

m_name	born	ISNULL(died)
Fred Bloggs	Fri, 02 Jan 1948 00:00:00 GMT	1
John Smith	Fri, 03 Mar 1950 00:00:00 GMT	1
Helen Smyth	Sun, 08 Aug 1948 00:00:00 GMT	1
Harriet Smithson	Sun, 09 May 1909 00:00:00 GMT	0
James First	Thu, 10 Jun 1965 00:00:00 GMT	1
Theo Mengel	Thu, 12 Aug 1948 00:00:00 GMT	1
Sue Little	Wed, 21 Feb 1945 00:00:00 GMT	1

- Tenemos un error al intentar usar el segundo parámetro salimos de los parámetros permitidos.

```
SELECT m_name, born, ISNULL(died, 'Esta vivo')
FROM musician
```

Submit SQL restore default

**Error:**

Incorrect parameter count in the call to native function 'ISNULL'

- COALESCE
  - Dime cuando el músico sigue vivo.

```
SELECT m_name, born, COALESCE(died, 'Esta vivo')
FROM musician
```

Submit SQL restore default

**Result:**

m_name	born	COALESCE(died..
Fred Bloggs	Fri, 02 Jan 1948 00:00:00 GMT	Esta vivo
John Smith	Fri, 03 Mar 1950 00:00:00 GMT	Esta vivo
Helen Smyth	Sun, 08 Aug 1948 00:00:00 GMT	Esta vivo
Harriet Smithson	Sun, 09 May 1909 00:00:00 GMT	1980-09-20
James First	Thu, 10 Jun 1965 00:00:00 GMT	Esta vivo
Theo Mengel	Thu, 12 Aug 1948 00:00:00 GMT	Esta vivo
Sue Little	Wed, 21 Feb 1945 00:00:00 GMT	Esta vivo

3 consultas: una para cada uno de los tipos de operadores lógicos

- EXISTS
  - ¿Artistas que hacen parte de una banda?

```
SELECT m_name
FROM musician
WHERE EXISTS (
  SELECT 1
  FROM plays_in
  WHERE m_no = player
```

Submit SQL restore default

Result:

m_name
Fred Bloggs
Helen Smyth
Harriet Smithson
James First
Theo Mengel
Sue Little
Harry Forte

- ANY
  - ¿Qué canciones tienen más reproducciones que el compositor con menos canciones?

```
SELECT c_title, c_in
FROM composition
WHERE c_in > ANY (
  SELECT comp_is
  FROM composer
);
```

Submit SQL restore default

Enviar el valor predeterminado de restauración de SQL

Result: Resultado:

c_title c_title	c_in c_in
Here Goes Aquí va	2
Valiant Knight Caballero valiente	3
Little Piece Pedacito	4
Simple Song Canción sencilla	5
Little Swing Song	6
Pequeña canción de swing	
Fast Journev Viaje rápido	7

- ALL
  - ¿Qué músicos nacieron después del último músico fallecido?

```
SELECT m_name, born
FROM musician
WHERE born > ALL (
  SELECT born
  FROM musician
  WHERE died IS NOT NULL
```

Submit SQL restore default

Enviar el valor predeterminado de restauración de SQL

**Result: Resultado:**

m_name m_name	born nacer
John Smith Juan Pérez	Fri, 03 Mar 1950 00:00:00 GMT Fri, 03 Mar 1950 00:00:00 GMT
Helen Smyth Helen Smyth	Sun, 08 Aug 1948 00:00:00 GMT Sun, 08 Aug 1948 00:00:00 GMT
James First Santiago Primero	Thu, 10 Jun 1965 00:00:00 GMT Thu, 10 Jun 1965 00:00:00 GMT

- CASE:  
Clasificar si un músico usa el violín, si no lo es, entonces llamarlo “Otro” :  
Musicians: easy questions

1.

Give the organiser's name of the concert in the Assembly Rooms after the first of Feb, 1997.

```
SELECT m_name, instrument, CASE WHEN instrument = "violin"
THEN 'Violinista' ELSE 'Otro' END
FROM musician JOIN performer ON (perf_is = m_no)
```

Submit SQL

restore default

John Smith	violin	Violinista
Harriet Smithson	viola	Otro
Theo Mengel	banjo	Otro
Harry Forte	violin	Violinista
Davis Heaven	guitar	Otro
Alan Fluff	violin	Violinista
James Quick	trumpet	Otro
Elsie James	viola	Otro
Louise Simpson	bass	Otro
John Smith	flute	Otro
Louise Simpson	viola	Otro

**REFERENCIAS:**

[https://www.w3schools.com/Sql/sql\\_join\\_left.asp](https://www.w3schools.com/Sql/sql_join_left.asp)

[https://www.w3schools.com/sql/sql\\_null\\_values.asp](https://www.w3schools.com/sql/sql_null_values.asp)

<https://www.youtube.com/watch?v=jYYMMWlyFEI>

<https://www.freecodecamp.org/espanol/news/tutorial-de-uniones-en-sql/>