**Autoestudio 02 - MBDA**

Jefer Alexis González Romero  
Angel Nicolas Cuervo Naranjo

**INVESTIGACION**

**A.** **NULL**

* ¿Qué significa?

**NULL** es un valor utilizado en una base de datos relacional cuando no se conoce o falta el valor, un valor en el campo de una tabla es vació.

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

**Operadores aritméticos:**

Como Null no es un dato con algún valor, sino un espacio vacío en la tabla, al operarlo con valores numéricos da como resultado un dato que es Null

**Operadores lógicos y de comparación:**

No es posible probar NULL con operadores de este tipo.

**B.** **JUNTA**

* ¿Cuáles son las diferencias entre junta interna y externa?

La diferencia entre la junta interna y la junta externa es que el **INNER JOIN** se utiliza únicamente para unir los datos de dos o más tablas que tienen relación entre ellos. En cambio el **OUTER JOIN,** además de hacer los que hace el **INNER JOIN,** también junta los datos que no se corresponden entre las dos o más tablas.

* ¿Qué opciones se tienen para la junta interna?

**INNER JOIN** tiene tres opciones que son: **JOIN,** quese usa para combinar filas de dos o más tablas, en función de una columna relacionada entre ellas. **NATURAL JOIN** la cualrelaciona dos tablas con los nombres de las columnas que tienen en común.Y, por último, **CROSS JOIN** que hace una multiplicación de todas las filas de una tabla por todas las filas de la otra tabla.

* ¿Qué opciones se tienen para la junta externa?

El **OUTER JOIN** tiene varias opciones con las que se pueden trabajar ya que tiene varios tipos de uniones las cuales son:

**FULL OUTER JOIN:** Este se usa para seleccionar los registros que van a completar la unión entre los datos en común entre ambas tablas. También selecciona los datos que se encuentran en la primera tabla que no cumplen con los requisitos de la unión que se quiere realizar. Y por último también selecciona los datos que no cumplen con los requisitos de unión de la segunda tabla.

**LEFT OUTER JOIN:** Al igual que el **FULL OUTER JOIN** selecciona los datos que completan la unión de los datos de las tablas que se quieren relacionar. Y los datos que no cumplen con los requisitos de la unión de la primera tabla también se seleccionan.

**RIGHT OUTER JOIN:** Al igual que el **FULL OUTER JOIN** y el **LEFT OUTER JOIN** selecciona los datos que completan la unión de ambas tablas, pero además selecciona los datos de la segunda lista que no cumplen con los requisitos de la unión, pero no los de la primera tabla.

**Bibliografía**

* <https://docs.microsoft.com/es-es/sql/connect/ado-net/sql/handle-null-values?view=sql-server-ver15#:~:text=Se%20utiliza%20un%20valor%20NULL,para%20tipos%20de%20datos%20num%C3%A9ricos>
* <https://ayuda.windev.es/es-ES/?2034007>
* <https://es.stackoverflow.com/questions/36/cu%C3%A1l-es-la-diferencia-entre-un-inner-y-un-outer-join>
* <https://firebird21.wordpress.com/2013/12/14/natural-join-y-cross-join/>

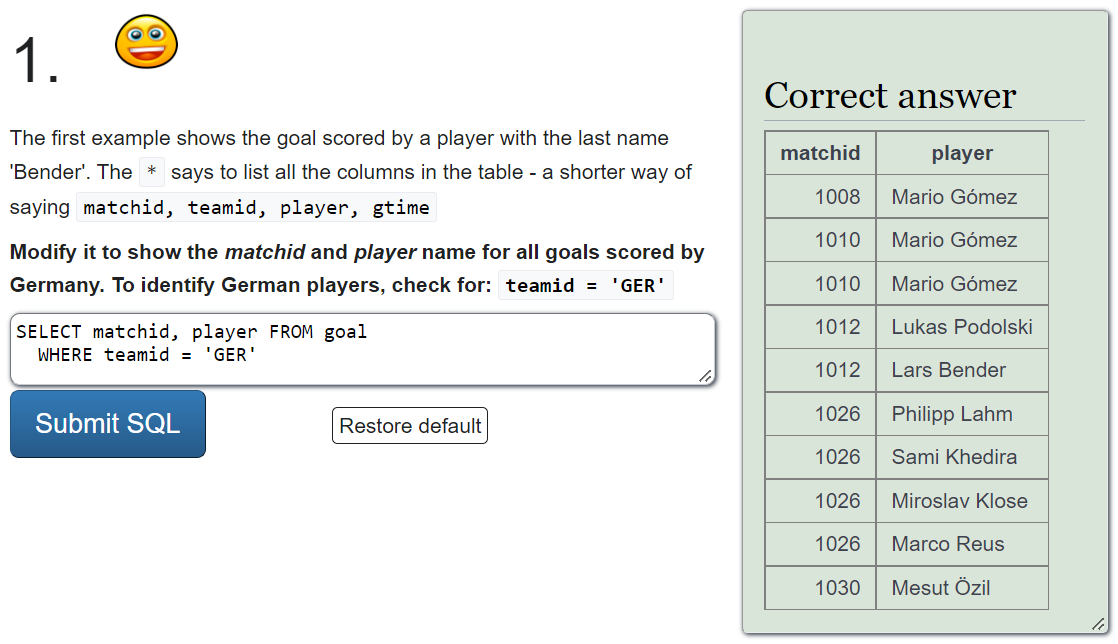
**PRÁCTICA**

**A.** Realicen los ejercicios propuestos en los siguientes tutoriales. Utilice el motor My SQL 5.

**JOIN**

SELECT matchid, player FROM goal

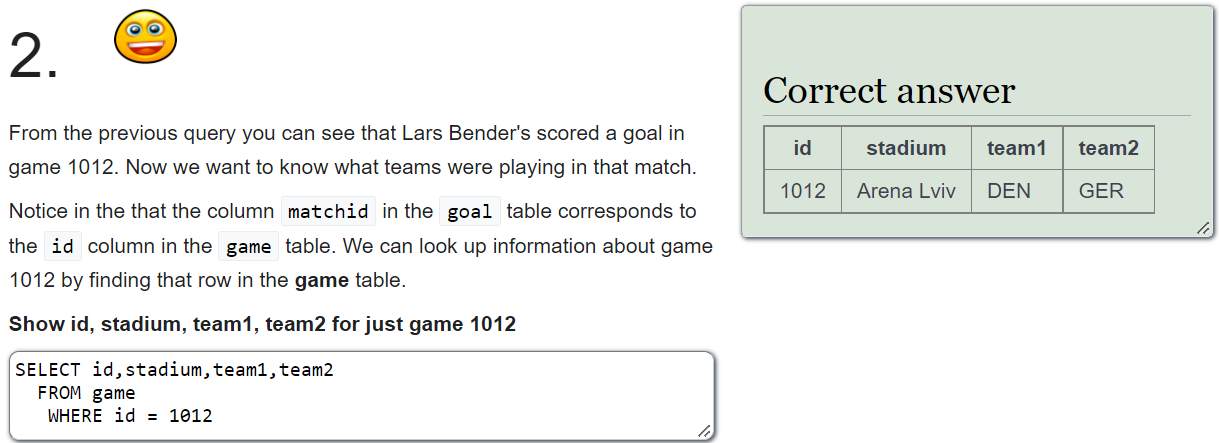
WHERE teamid = 'GER'



SELECT id,stadium,team1,team2

FROM game

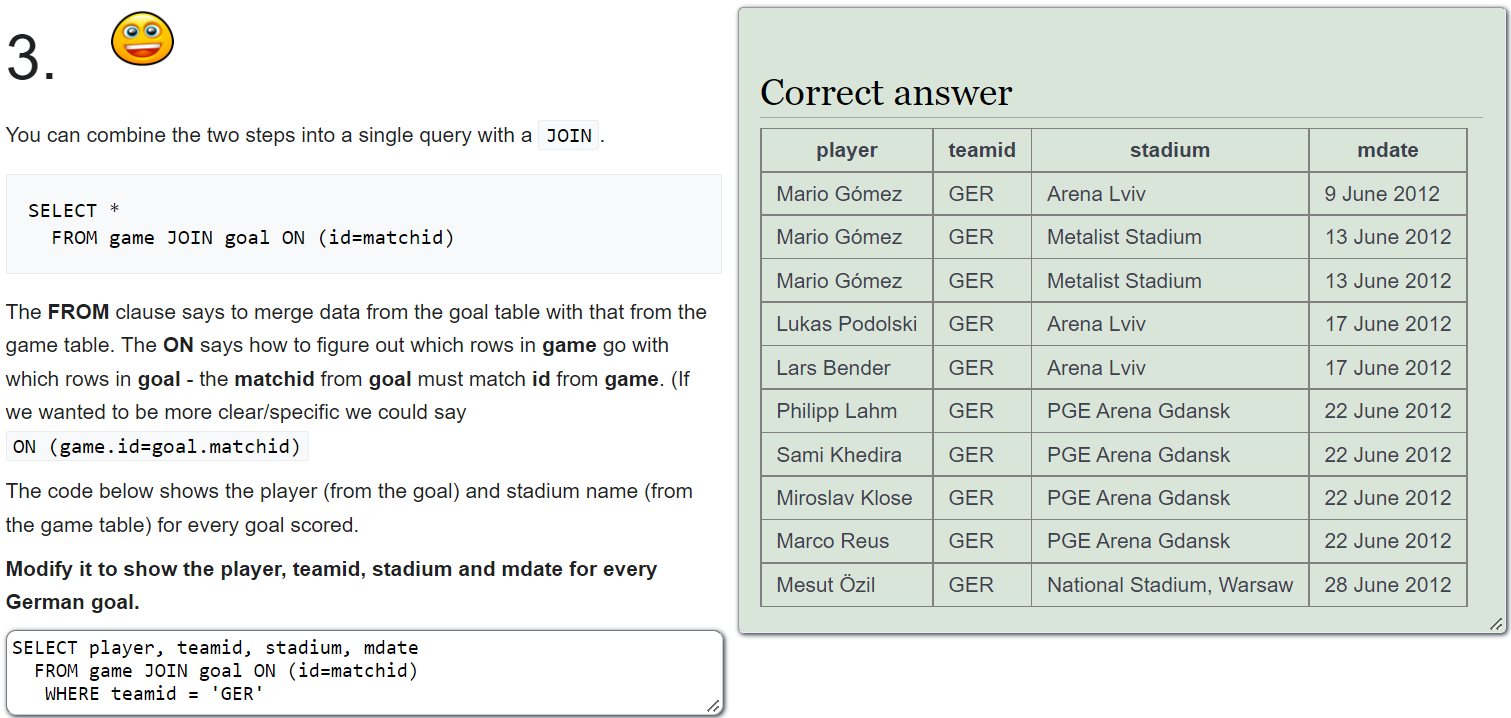
WHERE id = 1012



SELECT player, teamid, stadium, mdate

FROM game JOIN goal ON (id=matchid)

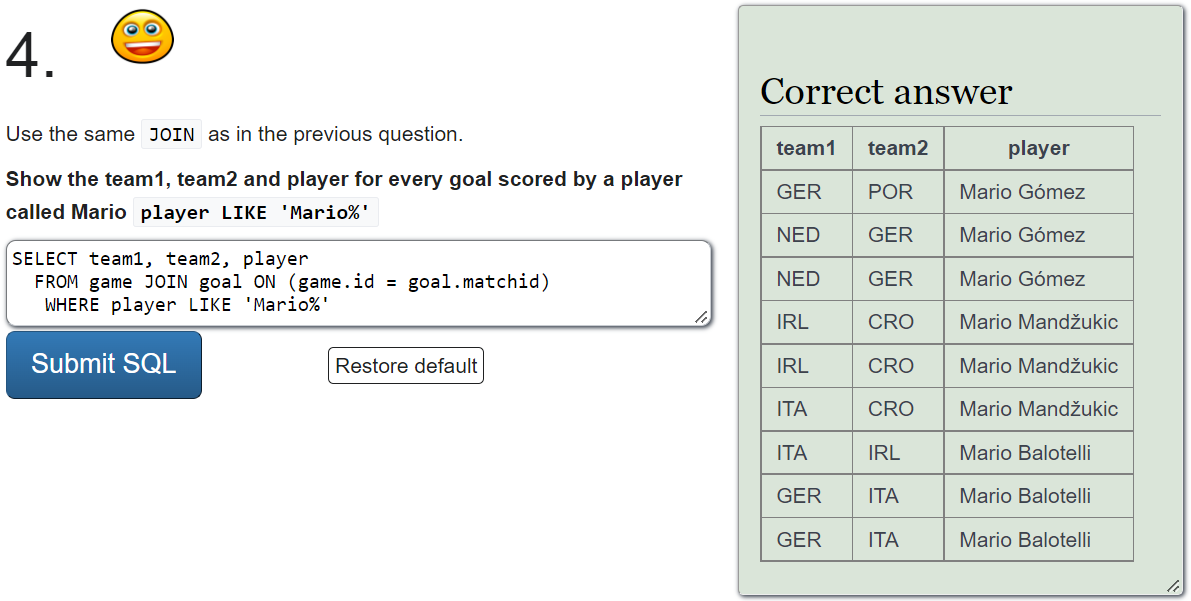
WHERE teamid = 'GER'



SELECT team1, team2, player

FROM game JOIN goal ON (game.id = goal.matchid)

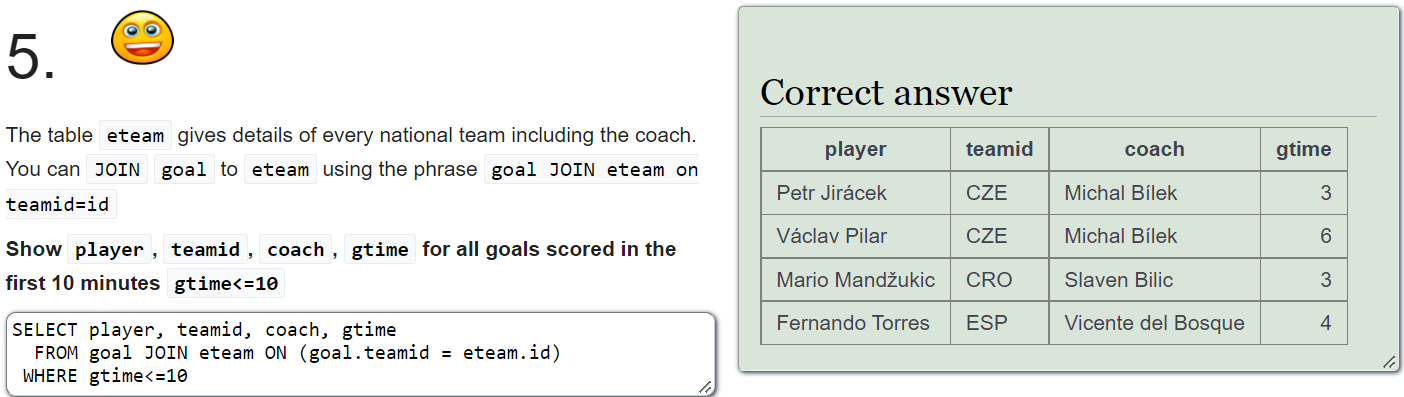
WHERE player LIKE 'Mario%'



SELECT player, teamid, coach, gtime

FROM goal JOIN eteam ON (goal.teamid = eteam.id)

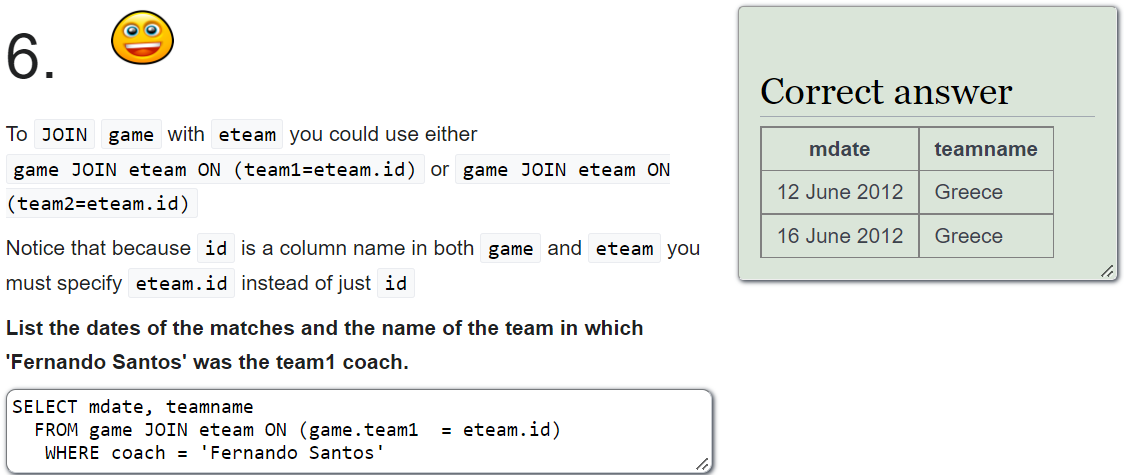
WHERE gtime<=10



SELECT mdate, teamname

FROM game JOIN eteam ON (game.team1 = eteam.id)

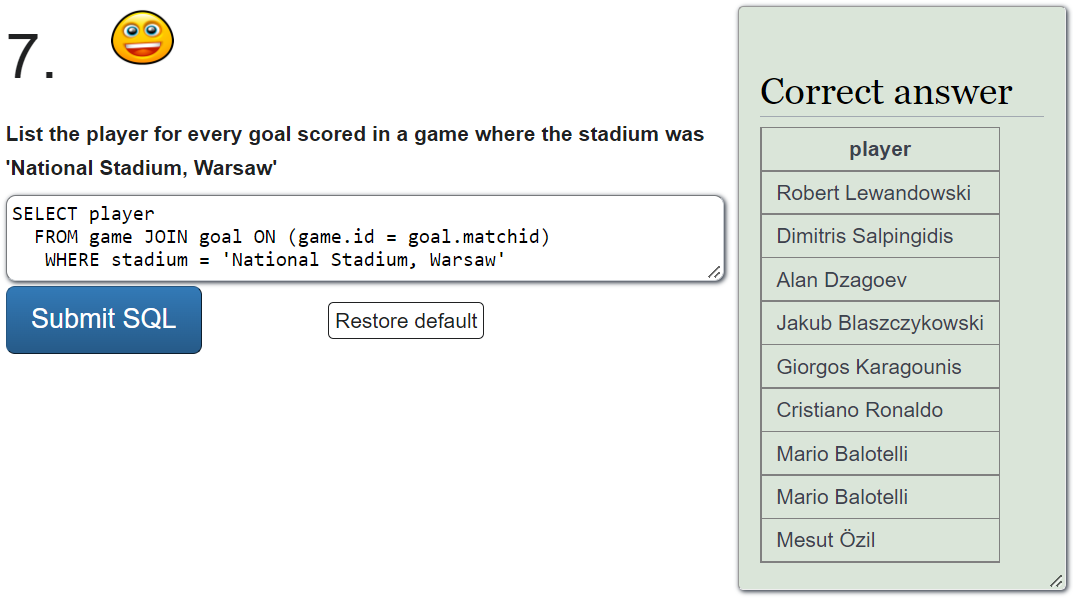
WHERE coach = 'Fernando Santos'



SELECT player

FROM game JOIN goal ON (game.id = goal.matchid)

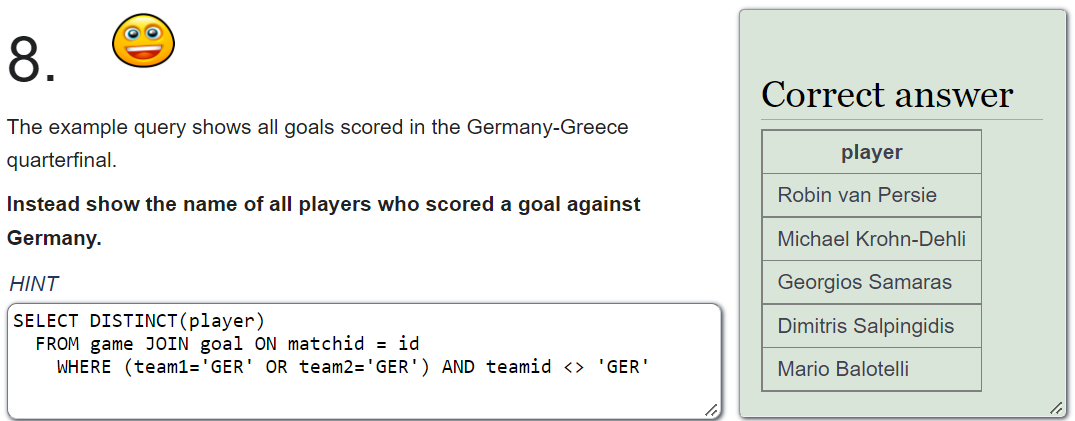
WHERE stadium = 'National Stadium, Warsaw'



SELECT DISTINCT(player)

FROM game JOIN goal ON matchid = id

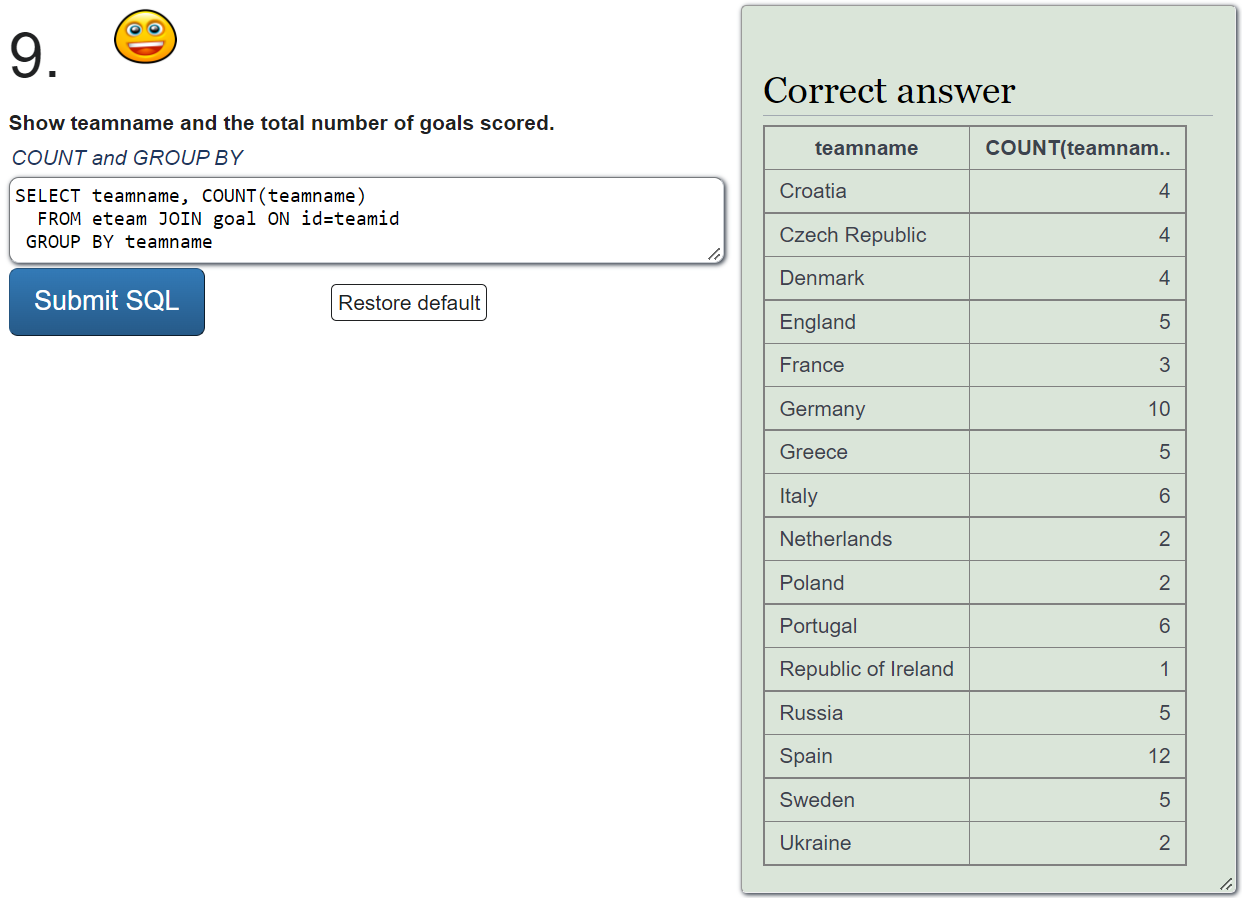
WHERE (team1='GER' OR team2='GER') AND teamid <> 'GER'



SELECT teamname, COUNT(teamname)

FROM eteam JOIN goal ON id=teamid

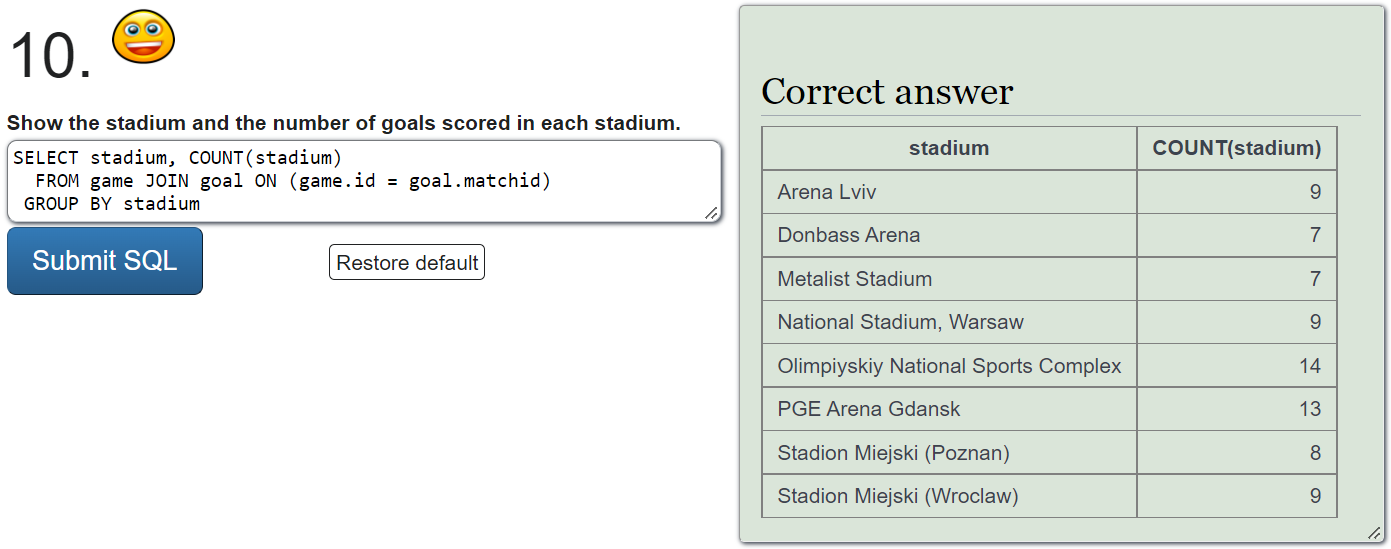
GROUP BY teamname



SELECT stadium, COUNT(stadium)

FROM game JOIN goal ON (game.id = goal.matchid)

GROUP BY stadium

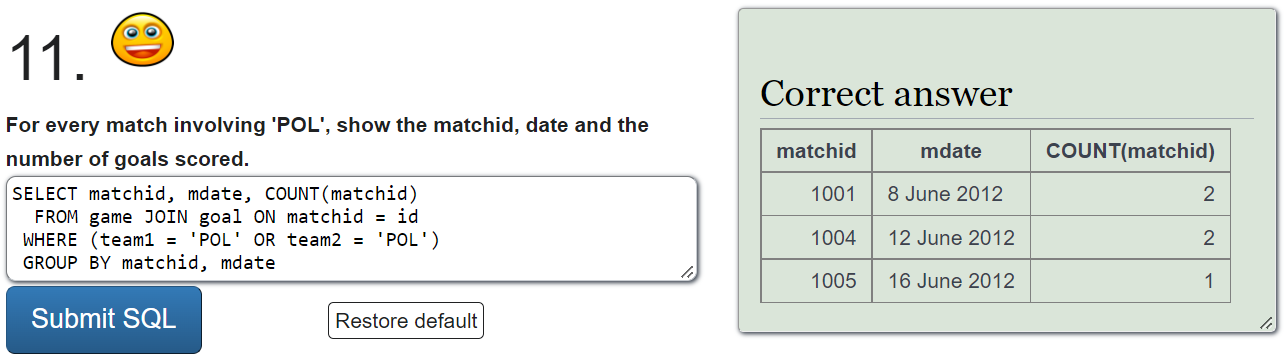


SELECT matchid, mdate, COUNT(matchid)

FROM game JOIN goal ON matchid = id

WHERE (team1 = 'POL' OR team2 = 'POL')

GROUP BY matchid, mdate

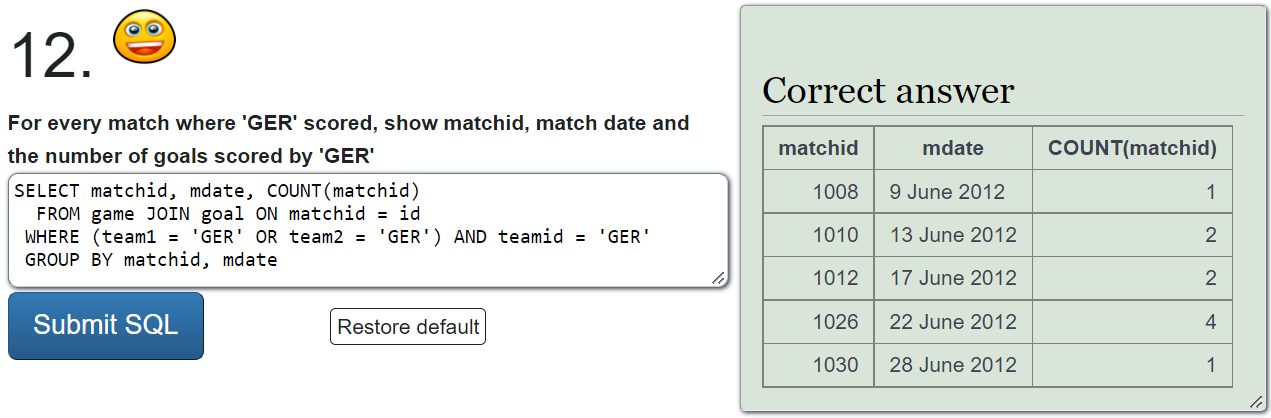


SELECT matchid, mdate, COUNT(matchid)

FROM game JOIN goal ON matchid = id

WHERE (team1 = 'GER' OR team2 = 'GER') AND teamid = 'GER'

GROUP BY matchid, mdate



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, matchid, team1, team2

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

**QUIZ**

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica

Descripción generada automáticamente con confianza mediaInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteTabla

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteTexto

Descripción generada automáticamente

Tabla

Descripción generada automáticamente**MORE JOIN**

SELECT id, title

FROM movie

WHERE yr=1962

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

SELECT yr

FROM movie

WHERE title = 'Citizen Kane'

Imagen que contiene Patrón de fondo

Descripción generada automáticamente

SELECT id, title, yr

FROM movie

WHERE title LIKE '%Star Trek%'

ORDER BY yr

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

SELECT id, title, yr

FROM movie

WHERE title LIKE '%Star Trek%'

ORDER BY yr

Imagen que contiene Patrón de fondo

Descripción generada automáticamente

SELECT id

FROM movie

WHERE title = 'Casablanca'

Aplicación

Descripción generada automáticamente con confianza baja

SELECT name

FROM casting JOIN actor

ON (casting.actorid = actor.id)

WHERE movieid = 11768

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza baja

SELECT name

FROM actor JOIN (casting JOIN movie

ON (casting.movieid = movie.id))

ON (casting.actorid = actor.id)

WHERE title = 'Alien'

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

SELECT title

FROM actor JOIN (casting JOIN movie

ON (casting.movieid = movie.id))

ON (casting.actorid = actor.id)

WHERE name = 'Harrison Ford'

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

SELECT title

FROM actor JOIN (casting JOIN movie

ON (casting.movieid = movie.id))

ON (casting.actorid = actor.id)

WHERE name = 'Harrison Ford' AND ord != 1

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

SELECT title, name

FROM actor JOIN (casting JOIN movie

ON (casting.movieid = movie.id))

ON (casting.actorid = actor.id)

WHERE yr = 1962 AND ord = 1

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza baja

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

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

SELECT title, name

FROM actor JOIN (casting JOIN movie

ON (casting.movieid = movie.id))

ON (casting.actorid = actor.id)

WHERE ord = 1 AND movieid IN (SELECT movieid

FROM actor JOIN casting ON (actor.id = casting.actorid)

WHERE name = 'Julie Andrews')

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

SELECT name FROM casting JOIN actor ON (casting.actorid = actor.id) WHERE ord = 1

GROUP BY name

HAVING COUNT(\*) >= 15

Texto

Descripción generada automáticamente

SELECT title, COUNT(\*)

FROM movie JOIN casting ON (movie.id = casting.movieid)

WHERE yr = 1978

GROUP BY title

ORDER BY COUNT(\*) desc, title

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

SELECT name

FROM actor JOIN casting ON (actor.id = casting.actorid)

WHERE movieid IN (SELECT movieid

FROM actor JOIN casting ON (actor.id = casting.actorid)

WHERE name = 'Art Garfunkel') AND name != 'Art Garfunkel'

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**QUIZ**

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteTabla

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteImagen que contiene Texto

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteTabla

Descripción generada automáticamenteForma

Descripción generada automáticamente con confianza media

**USIN NULL**

SELECT name

FROM teacher

WHERE dept is NULL

Imagen que contiene Patrón de fondo

Descripción generada automáticamente

SELECT teacher.name, dept.name

FROM teacher INNER JOIN dept

ON (teacher.dept=dept.id)

Tabla

Descripción generada automáticamente con confianza baja

SELECT teacher.name, dept.name

FROM teacher LEFT JOIN dept

ON (teacher.dept=dept.id)

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

SELECT teacher.name, dept.name

FROM teacher RIGHT JOIN dept

ON (teacher.dept=dept.id)

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

SELECT name, COALESCE(mobile, '07986 444 2266')

FROM teacher

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

SELECT teacher.name, COALESCE(dept.name, 'None')

FROM teacher LEFT JOIN dept ON (dept.id = teacher.dept)

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

SELECT COUNT(name), COUNT(mobile)

FROM teacher

Imagen que contiene Gráfico

Descripción generada automáticamente

SELECT dept.name, COUNT(dept)

FROM teacher RIGHT JOIN dept ON (teacher.dept = dept.id)

GROUP BY dept.name

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

SELECT name, CASE WHEN dept IN ('1', '2')

THEN 'Sci'

ELSE 'Art'

END

FROM teacher

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

SELECT name, CASE WHEN dept IN ('1', '2')

THEN 'Sci'

WHEN dept = 3

THEN 'Sci'

ELSE 'None'

END

FROM teacher

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

**QUIZ**

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica

Descripción generada automáticamente con confianza mediaImagen que contiene Forma

Descripción generada automáticamenteTexto

Descripción generada automáticamenteForma

Descripción generada automáticamente con confianza mediaInterfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamenteTabla

Descripción generada automáticamenteForma, Rectángulo

Descripción generada automáticamente

**SELF JOIN**

SELECT COUNT(id)

FROM stops

Tabla

Descripción generada automáticamente con confianza baja

SELECT id

FROM stops

WHERE name = 'Craiglockhart'Imagen que contiene Patrón de fondo

Descripción generada automáticamente

SELECT id, name

FROM route JOIN stops ON (stops.id = route.stop)

WHERE company = 'LRT' AND num = 4

ORDER BY pos

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

SELECT company, num, COUNT(\*)

FROM route WHERE stop=149 OR stop=53

GROUP BY company, num

HAVING COUNT(\*) >= 2

Texto, Aplicación

Descripción generada automáticamente con confianza media

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

Interfaz de usuario gráfica, Texto, Aplicación, Word

Descripción generada automáticamente

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'

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

SELECT 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'

GROUP BY a.company, a.num

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

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 (a.stop=stopa.id)

JOIN stops stopb ON (b.stop=stopb.id)

WHERE stopa.name='Craiglockhart' AND stopb.name='Tollcross'

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

SELECT stopb.name, b.company, b.num

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 b.company = 'LRT'

ORDER BY b.num, b.pos

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

SELECT a.num, a.company, stopb.name, c.num, c.company

FROM route a JOIN route b ON (a.company=b.company AND a.num = b.num) JOIN route c JOIN route d ON (c.company=d.company AND c.num = d.num)

JOIN stops stopa ON (a.stop=stopa.id)

JOIN stops stopb ON (b.stop=stopb.id)

JOIN stops stopc ON (c.stop=stopc.id)

JOIN stops stopd ON (d.stop=stopd.id)

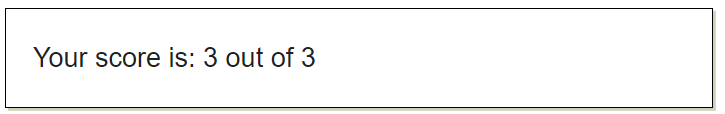
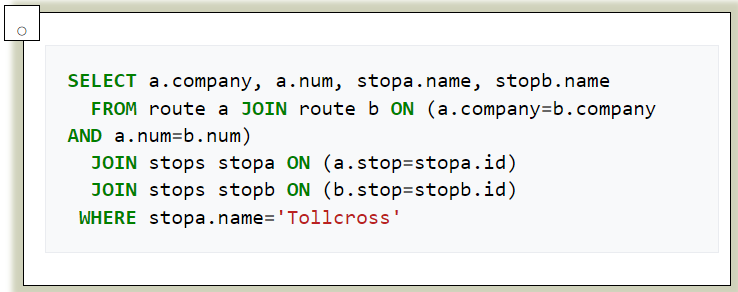
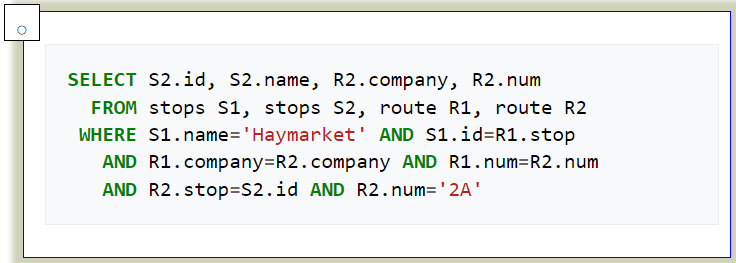
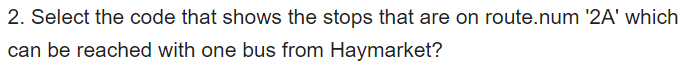
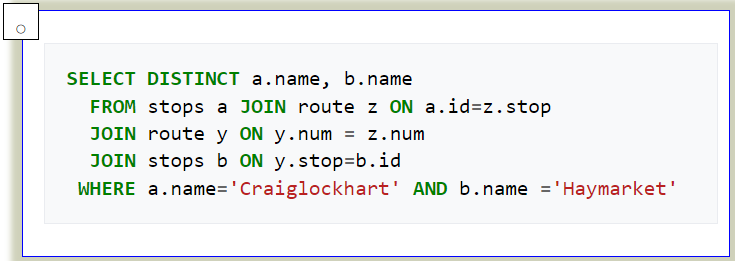
WHERE stopa.name='Craiglockhart' and stopd.name='Lochend' AND stopb.name=stopc.name

GROUP BY a.num, a.company, stopb.name, c.num, c.company

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

**QUIZ**



**C.** Propongan preguntas que cumplan los siguientes requerimientos. Usen la base de datos Musicians

* 5 consultas: una para cada operador de conjuntos

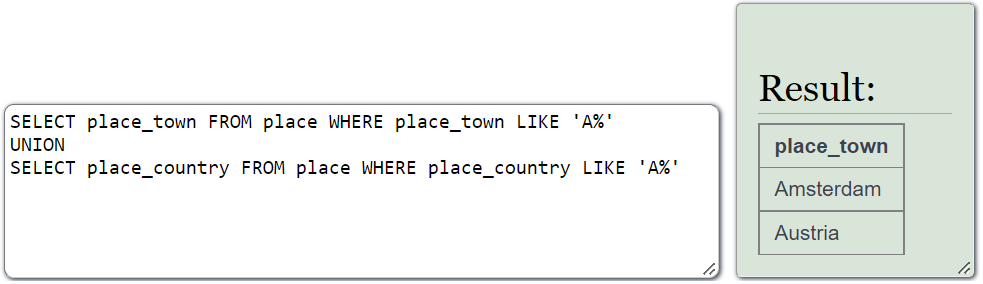
**UNION**

¿Cuáles son los países o ciudades que comienzan por ‘A’?

SELECT place\_town FROM place WHERE place\_town LIKE 'A%'

**UNION**

SELECT place\_country FROM place WHERE place\_country LIKE 'A%'



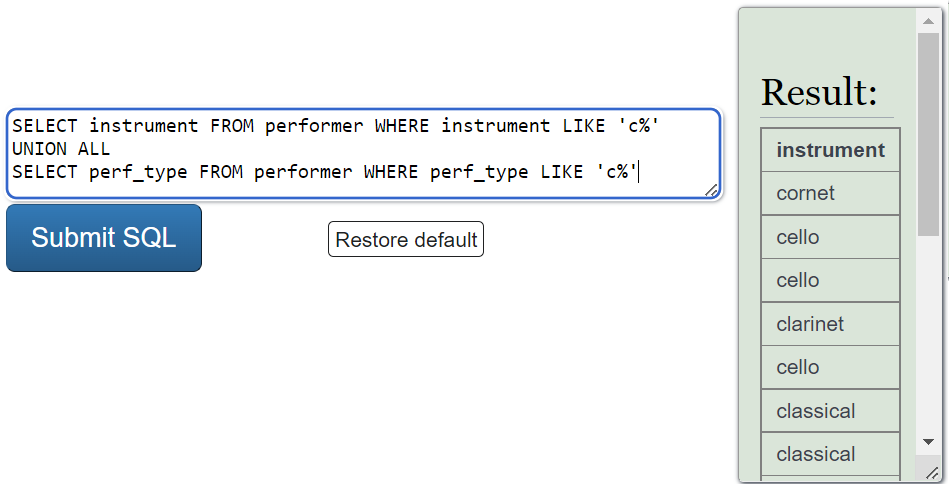
**UNION ALL**

¿Cuáles instrumentos o géneros musicales empiezan por ‘c’? se pueden repetir tanto género como instrumentos.

SELECT instrument FROM performer WHERE instrument LIKE 'c%'

**UNION**

SELECT perf\_type FROM performer WHERE perf\_type LIKE 'c%'



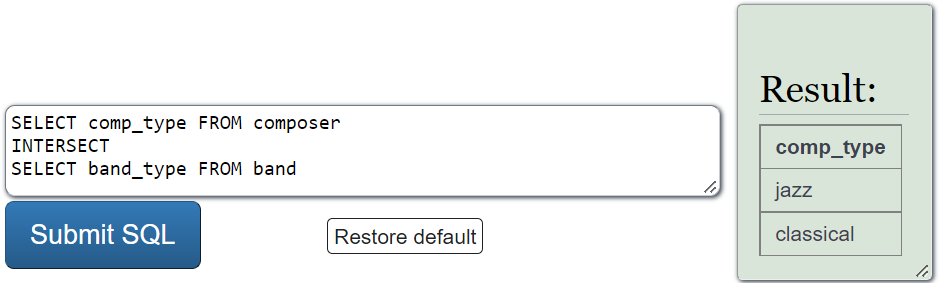
**INTERSECT**

¿Qué géneros de música están presentes tanto en las bandas como en las composiciones?

SELECT comp\_type FROM composer

**INTERSECT**

SELECT band\_type FROM band



**EXTRACT**

¿En qué años y sedes se hicieron los diferentes conciertos?

SELECT concert\_venue AS Sede, **EXTRACT** (YEAR FROM con\_date) AS Año

FROM concert



**IN**

¿Qué músicos son compositores?

SELECT m\_name

FROM musician

WHERE m\_no **IN** (SELECT comp\_is FROM composer)

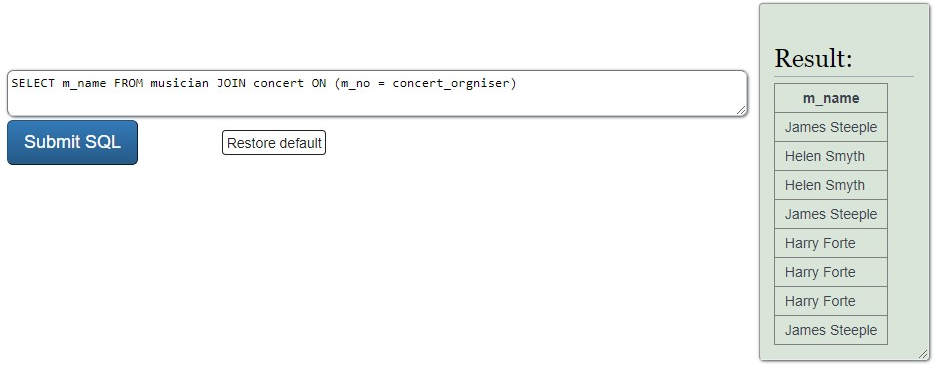


* 6 consultas: una para cada operador de junta

**JOIN**

¿Cuáles son los nombres de los organizadores?

SELECT m\_name FROM musician **JOIN** concert ON (m\_no = concert\_orgniser)



**NATURAL JOIN**

En la base de datos musicians no hay columnas de diferentes tablas con el mismo nombre para hacer la junta natural.

**CROSS JOIN**

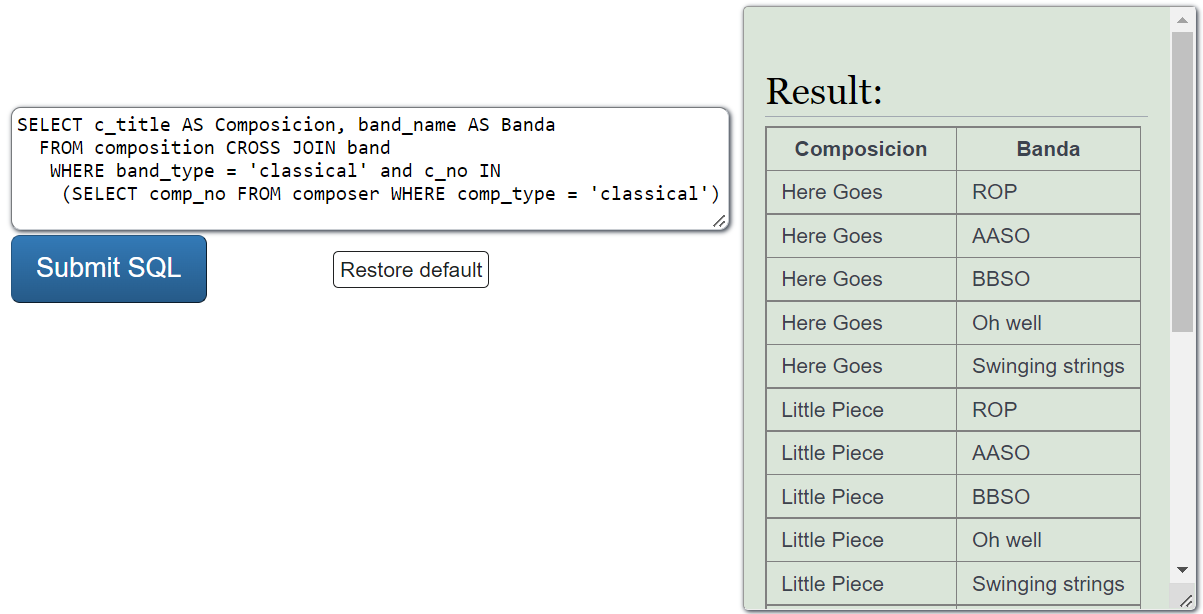
¿Cuáles son las diferentes combinaciones entre las composiciones compuestas por compositores de música clásica y las bandas que tocan música clásica?

SELECT c\_title AS Composicion, band\_name AS Banda

FROM composition **CROSS JOIN** band

WHERE band\_type = 'classical' and c\_no IN

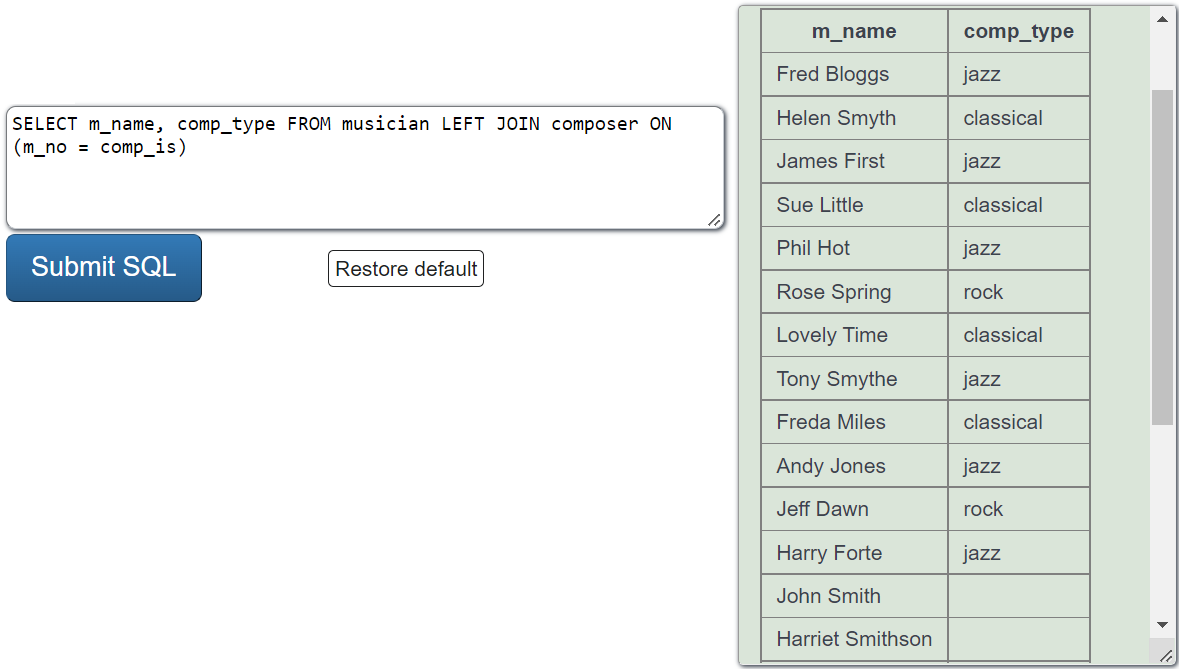
(SELECT comp\_no FROM composer WHERE comp\_type = 'classical')



**LEFT JOIN**

¿Cuáles son los músicos que hay? Si son compositores mostrar el género de música que componen

SELECT m\_name, comp\_type FROM musician **LEFT JOIN** composer ON (m\_no = comp\_is)

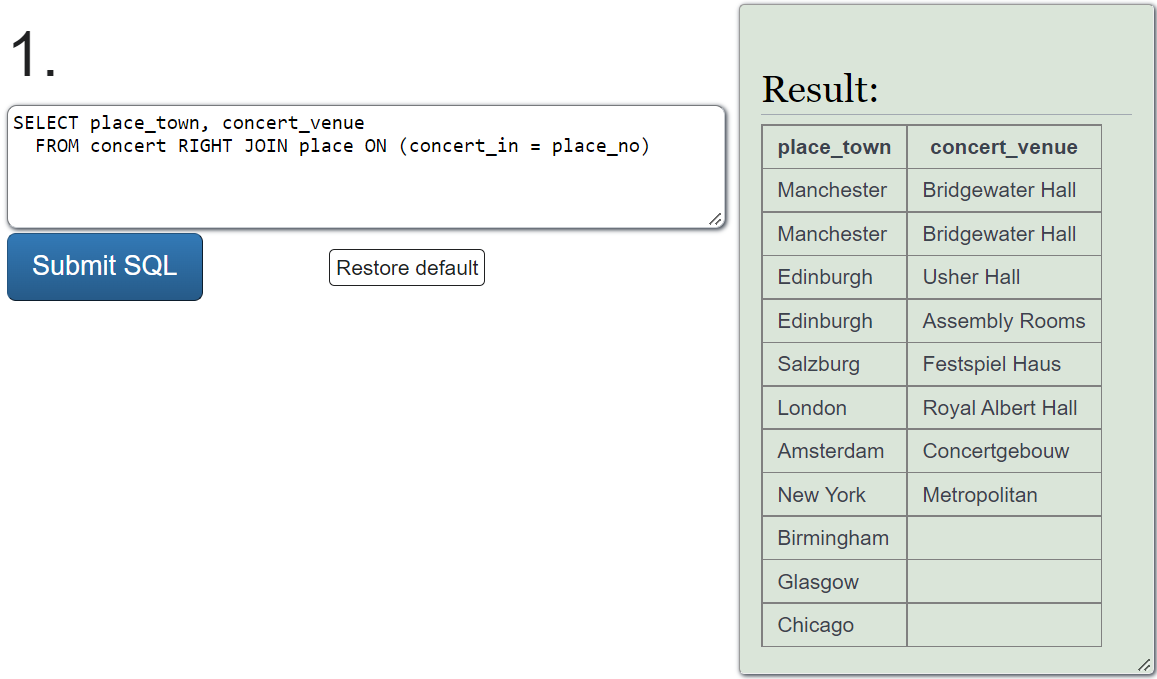


**RIGTH JOIN**

¿Cuáles ciudades hay? Si en alguna se realizó un concierto indicar la sede en la que se hizo.

SELECT place\_town, concert\_venue

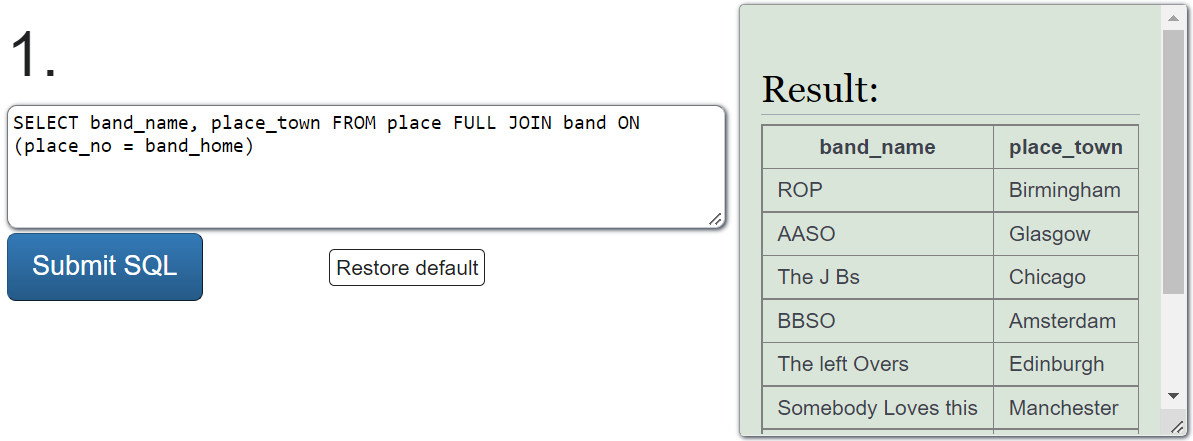
FROM concert RIGHT JOIN place ON (concert\_in = place\_no)



**FULL JOIN**

¿Cuáles son todas las bandas y todos los lugares? Relacionando la banda con el lugar donde se ubican

SELECT band\_name, place\_town FROM place **FULL JOIN** band ON (place\_no = band\_home)



* 2 consultas: una para cada operador de desconocido

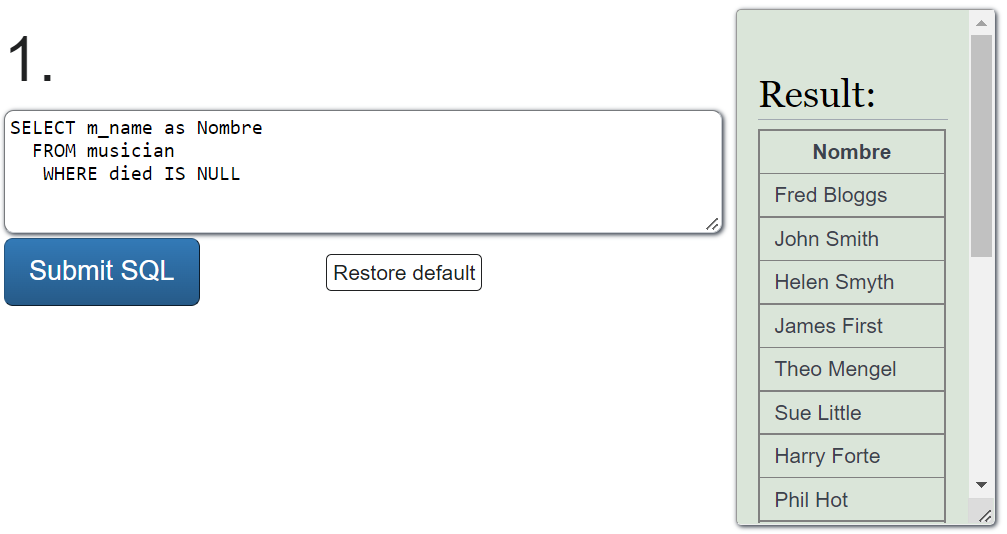
**ISNULL**

¿Cuáles son los músicos que no han muerto?

SELECT m\_name as Nombre

FROM musician

WHERE died **IS NULL**

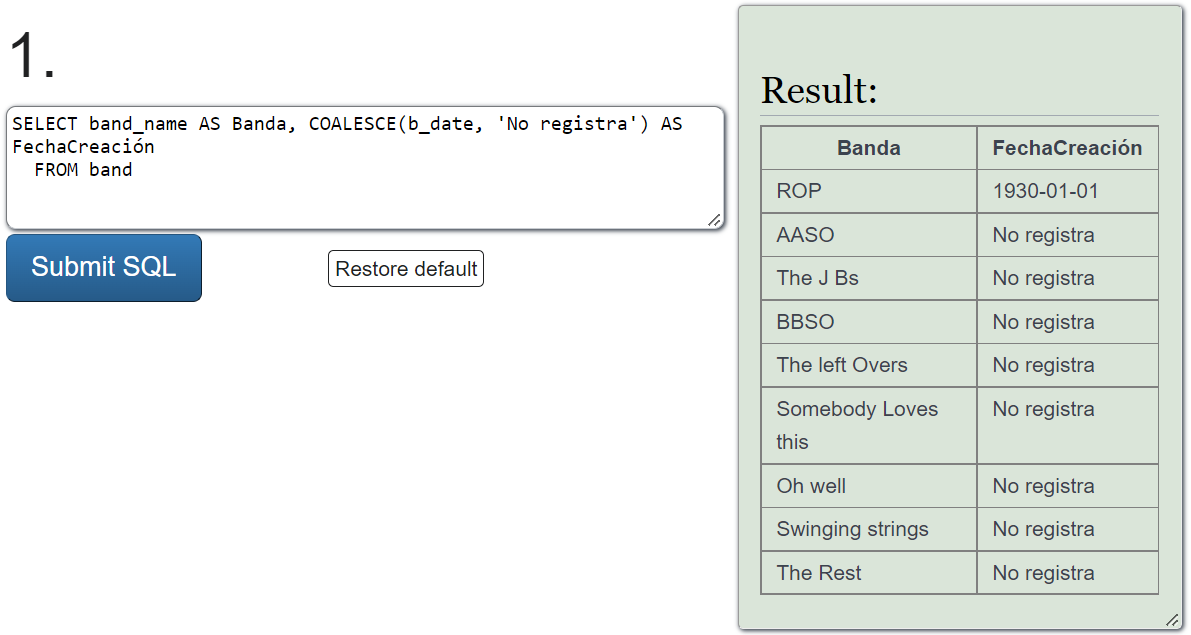


**COALESCE**

¿Cuáles son las fechas de creación de las diferentes bandas?

SELECT band\_name AS Banda, **COALESCE**(b\_date, 'No registra') AS FechaCreación

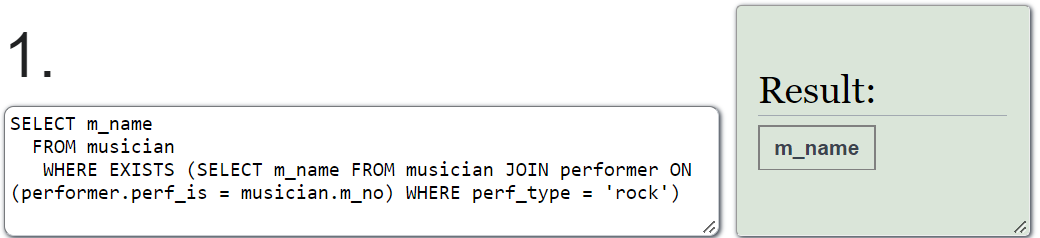
FROM band



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

**EXISTS**

Si existe un artista que toque el género rock mostrar el nombre de todos los músicos.



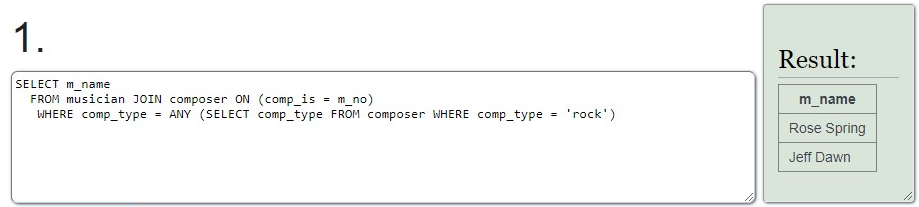
**ANY**

Muestre los compositores de rock si hay

SELECT m\_name

FROM musician JOIN composer ON (comp\_is = m\_no)

WHERE comp\_type = ANY (SELECT comp\_type FROM composer WHERE comp\_type = 'rock')



**ALL**

¿Cuáles son las bandas que en todos sus conciertos fue organizado por ‘Helen Smyth’?

SELECT gave

FROM performance

WHERE performed\_in = **ALL** (SELECT concert\_no

FROM concert JOIN musician ON (m\_no = concert\_orgniser)

WHERE m\_name = 'Helen Smyth')

* 1 consulta: para el operador CASE

¿Cuáles es el tipo de cada instrumento que hay?

SELECT distinct instrument,

CASE WHEN instrument IN ('violin', 'viola', 'banjo', 'guitar', 'bass', 'cello') THEN 'Cuerda'

WHEN instrument = 'drums' THEN 'Percusión'

ELSE 'Viento'

END AS Tipo

FROM performer

