**Modelos y Bases de Datos**

**Autoestudio 2**

**INVESTIGACIÓN**

1. **Null**

**1. ¿Qué significa?**

En SQL, NULL es un *estado* que indica que el valor de ese item es desconocido o no existente. No es cero o blanco o una “cadena vacía” y no se comporta como ninguno de esos valores. Pocas cosas en SQL llevan a tanta confusión como NULL, y será difícil de entender mientras no entiendas la siguiente simple definición: NULL significa *desconocido*.

<https://www.firebirdsql.org/manual/es/nullguide-es-queesnull.html>

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

El valor NULL es un valor especial, y por tanto, no se puede comparar con los operadores aritméticos normales (=, >, <, <>), y en su lugar debemos utilizar los operadores IS y IS NOT.

Null no está considerado propiamente un “valor”, pero realmente sí que es un marcador que indica la ausencia de un valor. Debido a esto, las comparaciones con Null nunca resultan ciertas o falsas, pero si como un resultado lógico ternario, desconocidas (unknown).

<https://es.wikipedia.org/wiki/Null_(SQL)>

<http://sql.11sql.com/sql-valores-null.htm>

1. **Junta**

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

La junta interna es la combinación que dada la consulta proporciona la intersección de las dos tablas, que vendria siendo las dos filas que tienen en común. Por otra parte, en la junta externa en la unión no es necesario que un registro en una tabla tenga un registro relacionado en la otra.

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

* JOIN
* CROSS JOIN
* NATURAL JOIN

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

* LEFT JOIN
* RIGHT JOIN
* FULL JOIN

**PRÁCTICA**

**JOIN**

1. **SELECT matchid,player FROM goal**

**WHERE teamid = 'GER'**

1. **SELECT id,stadium,team1,team2**

**FROM game WHERE id = 1012**

1. **SELECT player,teamid,stadium,mdate**

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

**WHERE teamid = ‘GER’**

1. **SELECT team1,team2,player**

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

**WHERE player LIKE 'Mario%'**

1. **SELECT player, teamid,coach,gtime**

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

**WHERE gtime<=10**

1. **SELECT mdate,teamname**

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

**WHERE coach = ‘Fernando Santos’**

1. **SELECT player**

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

**WHERE stadium = 'National Stadium, Warsaw'**

1. **PREGUNTAR**
2. **SELECT teamname, COUNT(gtime)**

**FROM eteam JOIN goal ON id=teamid**

**GROUP BY teamname**

1. **SELECT stadium, COUNT(gtime)**

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

**GROUP BY stadium**

1. **SELECT matchid,mdate,z FROM**

**(select matchid,count(teamid) as z**

**FROM game JOIN goal ON id= matchid**

**WHERE team1= 'POL' or team2= 'POL'**

**GROUP BY matchid) AS X**

**JOIN game ON id= matchid**

1. **SELECT matchid,mdate, z FROM**

**(SELECT matchid,count(teamid) AS z**

**FROM game JOIN goal ON Id= matchid**

**WHERE (team1= 'GER' or team2= 'GER') and teamid= 'GER'**

**GROUP BY matchid) AS x JOIN game ON id= matchid**

**MORE JOIN OPERATION**

1. **SELECT id, title**

**FROM movie**

**WHERE yr=1962**

1. **SELECT yr**

**FROM movie**

**WHERE title = 'Citizen Kane'**

1. **SELECT yr**

**FROM movie**

**WHERE title = 'Citizen Kane'**

1. **SELECT id**

**FROM actor**

**WHERE name = 'Glenn Close'**

1. **SELECT id**

**FROM movie**

**WHERE title = 'Casablanca'**

1. **SELECT name**

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

**WHERE movieid = 11768**

1. **SELECT name**

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

**WHERE movieid = (SELECT id FROM movie WHERE title = ‘Alien’)**

1. **SELECT title**

**FROM movie JOIN casting ON(id = movieid) JOIN actor ON(actorid = actor.id)**

**WHERE actor.name = ‘Harrison Ford’**

1. **-**
2. **SELECT title,name**

**FROM movie JOIN casting ON(movie.id = casting.movieid) JOIN actor ON(casting.actorid = actor.id)**

**WHERE ord 1 AND yr = 1962**

1. **SELECT yr,COUNT(title) FROM**

**movie JOIN casting ON movie.id=movieid JOIN actor ON (actorid=actor.id) WHERE name='John Travolta' GROUP BY yr**

**HAVING COUNT(title)=(SELECT MAX(c)**

**FROM (SELECT yr,COUNT(title) AS c FROM movie JOIN casting ON movie.id=movieid JOIN actor ON actorid=actor.id where name='John Travolta' GROUP BY yr) AS t)**

1. **SELECT title,name**

**FROM movie JOIN casting ON (movie.id = casting.movieid and ord=1) JOIN actor ON(casting.actorid = actor.id)**

**WHERE movie.id IN (SELECT movieid FROM casting WHERE actorid IN ( SELECT id FROM actor WHERE name='Julie Andrews'))**

1. **SELECT actor.name**

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

**WHERE ord = 1 GROUP BY name HAVING COUNT(\*) >= 30**

1. **SELECT movie.title,count(actorid)**

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

**WHERE movie.yr= 1978 GROUP BY movie.id**

**ORDER BY COUNT(actorid) DESC**

**USING NULL**

1. **SELECT name FROM teacher**

**WHERE dept IS NULL**

1. **SELECT teacher.name, dept.name**

**FROM teacher INNER JOIN dept**

**ON (teacher.dept=dept.id)**

1. **SELECT teacher.name, dept.name**

**FROM teacher LEFT JOIN dept**

**ON (teacher.dept=dept.id)**

1. **SELECT teacher.name, dept.name**

**FROM teacher RIGHT JOIN dept**

**ON (teacher.dept=dept.id)**

1. **SELECT teacher.name, COALESCE(teacher.mobile,'07986 444 2266') FROM teacher**
2. **SELECT teacher.name, COALESCE(dept.name,'None') FROM teacher**

**LEFT JOIN dept ON teacher.dept = dept.id**

1. **SELECT COUNT(name), COUNT(mobile) FROM teacher**
2. **SELECT dept.name, COUNT(teacher.dept) FROM teacher**

**RIGHT JOIN dept ON dept.id = teacher.dept**

**GROUP BY dept.name**

1. **SELECT name, CASE WHEN dept IN (1,2) THEN 'Sci'**

**ELSE 'Art'**

**END**

**FROM teacher**

1. **SELECT name, CASE WHEN dept IN (1,2) THEN 'Sci'**

**WHEN dept = 3 THEN 'Art'**

**ELSE 'None'**

**END**

**FROM teacher**

**SELF JOIN**

1. **SELECT COUNT(name) AS stops FROM stops**
2. **SELECT id FROM stops**

**WHERE name = ‘Creiglockhart’**

1. **SELECT id, name**

**FROM route INNER JOIN stops**

**ON (id = stop) WHERE num = 4 AND company = ‘LRT’**

1. **SELECT company,num,COUNT(\*)**

**FROM route WHERE stop = 149 OR stop = 53**

**GROUP BY company,num HAVING COUNT(\*) = 2**

1. **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**

1. **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'**

1. **SELECT a.company, a.num**

**FROM route a, route b**

**WHERE a.num = b.num AND (a.stop = 115 AND b.stop = 137)**

**GROUP BY num;**

1. **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';**

1. **SELECT DISTINCT name, a.company, a.num**

**FROM route a**

**JOIN route b ON (a.company = b.company AND a.num = b.num)**

**JOIN stops ON a.stop = stops.id**

**WHERE b.stop = 53;**

1. **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 = 'Sighthill'**

**AND stopb.name = stopc.name**

**ORDER BY LENGTH(a.num), b.num, stopb.name, LENGTH(c.num), d.num;**