

Sentencia JOIN en SQL

Santiago Martin Pereira

Curso de Tecnólogo en Análisis y Desarrollo de Sistemas– Instituto Federal Sul-Rio-Grandense (IFSUL) – Campus de Santana do Livramento
RS– Brasil

santiago.martin@estudiantes.utec.edu.uy

Abstract. *This article describes the syntax that should be implemented in the SQL language when making a union between tables at the moment of making a SELECT, for that we can use INNER JOIN, LEFT JOIN or RIGHT JOIN, each function has a purpose that will be described together to its behavior with examples of each function.*

Resumen. *Este artículo describe las sintaxis que deberán ser implementadas en el lenguaje SQL al momento de realizar una unión entre tablas al instante de hacer un SELECT, para eso podemos utilizar INNER JOIN, LEFT JOIN o RIGHT JOIN, cada función tiene una finalidad que será descrita junto a su comportamiento con ejemplos de cada función.*

1. Introducción

La sentencia SQL JOIN se utiliza cuando queremos realizar un Select para obtener los registros de una base de dato, pero comparando la información de dos tablas. Para eso primero se debe pasar el nombre de una de las tablas, después el tipo de Join, en seguida la segunda tabla que se desea unir y al final se debe pasa una condición para comparar dichas tablas. De esta manera se genera una tabla temporal con el resultado que se obtiene del Join, para poder retornarlo y/o mostrarlo como un select normal

FROM Tabla1 [INNER/LEFT/RIGHT] JOIN Tabla2 ON Condiciones_Vinculos_Tablas

2. INNER JOIN

Para obtener la unión central entre dos tablas debemos utilizar la instrucción INNER JOIN. Esta devuelve únicamente aquellos registros que cuentan con valores coincidentes en los campos indicados al momento de realizar la comparación entre las tablas.

Así, para seleccionar los registros comunes entre la *TablaA* y la *TablaB* que tengan correspondencia entre ambas tablas por el campo *ID*, escribiríamos:

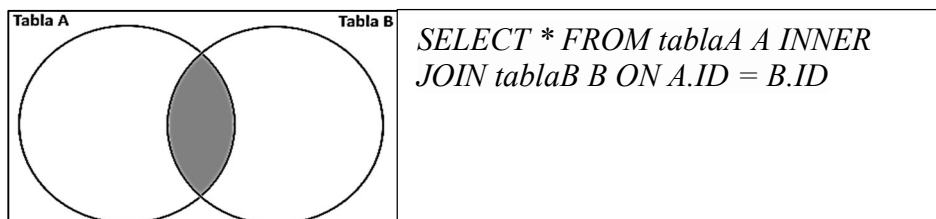


Ilustración 1 :Inner Join de manera grafica

3. LEFT JOIN

Se obtienen todos los registros de la tabla colocada a la izquierda, aunque no tengan correspondencia en la tabla de la derecha. Además de los registros que coinciden entre

ambas tablas. Es decir, aquellos registros que podemos obtener con el INNER JOIN, pero sumando todos los registros de la tabla declarada a la izquierda.

Así, para seleccionar los registros comunes entre la *TablaA* y la *TablaB* que tengan correspondencia entre ambas tablas por el campo *ID*, además de todos los registros de la *TablaA*, escribiríamos.

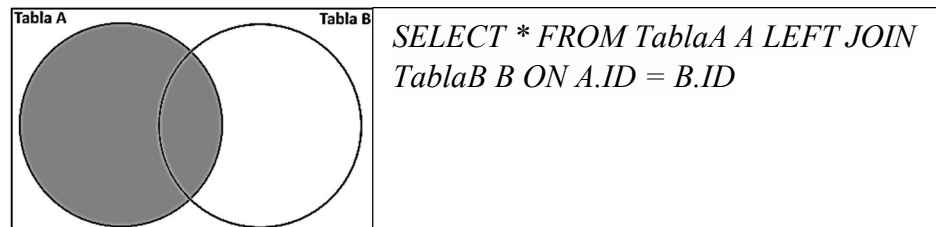


Ilustración 2: Left Join de manera grafica

4. RIGHT JOIN

Similar al ejemplo anterior, usando RIGHT JOIN se obtienen todos los registros de la tabla de la derecha, aunque no tengan correspondencia en la tabla de la izquierda. Además de los registros coincidentes entre las mismas.

Así, para seleccionar los registros comunes entre la *TablaA* y la *TablaB* que tengan correspondencia entre ambas tablas por el campo *ID*, además de todos los registros de la *TablaB*, escribiríamos.

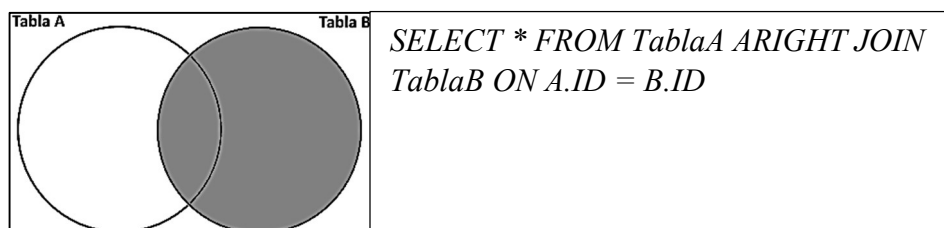


Ilustración 3: Right Join de manera grafica

Conclusión.

Los Join son una herramienta importante que complementan el Select, en el momento de realizar una búsqueda en la base de datos, se efectúa una comparación entre campos de dos tablas diferentes, con la intención de retornar datos que nos permitan tener una mayor comprensión de dicha información. Por ejemplo. Podemos cumplir con los requisitos de un programa que requiera toda la información de los FUNCIONARIOS y de los DEPARTAMENTOS en donde trabajan. Siendo FUNCIONARIOS y DEPARTAMENTOS tablas diferentes, tendremos que realizar un INNER JOIN para poder garantizar que se retornara apenas a los funcionarios que trabajen en algún departamento, excluyendo los demás registros.

```
SELECT * FROM Funcionarios F INNER JOIN Departamentos D ON F.Id_Dep= D.ID
```

-

Referencias.

ARES, Luis A. González. Ampliación de SQL.

Disponível: http://quegrande.org/apuntes/EI/4/BD2/practicas/09-10/ampliacion_de_sql.pdf

HEUSER, Carlos A. Projeto de Banco de Dados. 4ª Edição.