Contents

[cláusula JOIN 1](#_Toc101945409)

[UNION 3](#_Toc101945410)

[Ejemplos Con join 4](#_Toc101945411)

[Ejemplos con union 5](#_Toc101945412)

[Ejercicios con join y unión. 6](#_Toc101945413)

[Usar el JOIN para FILTRAR donde NO HAY COINCIDENCIA 7](#_Toc101945414)

[Vincular mas de una tabla con join 8](#_Toc101945415)

Cláusula JOINpermite **combinar registros de diferentes tablas horizontalmente**. Se complementa con la cláusula **ON** que nos permite establecer la condición por la cual queremos unir la tabla, que generalmente son campos que tienen en común las tablas.

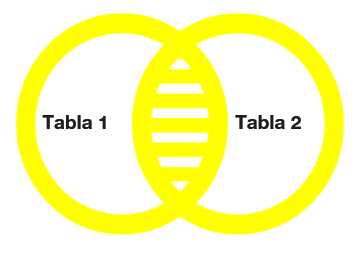
Diagram

Description automatically generated with medium confidence

interseccion

Datos en comun

A JOIN clause is used to combine rows from two or more tables, based on a related column between them.

El INNER JOIN o simplemente JOIN, retorna todas las filas de las dos tablas siempre y cuando haya coincidencia por el campo declarado en el ON. selects records that have matching values in both tables

select T1.C1, T2.C2

from T1 INNER JOIN T2

on T1.ColumnaenCOMUN = T2.ColumnaenCOMUN;

Graphical user interface

Description automatically generated with medium confidenceGraphical user interface, application, Teams

Description automatically generated

Por ejemplo…

select Personas.Apellido, Logros.Logro

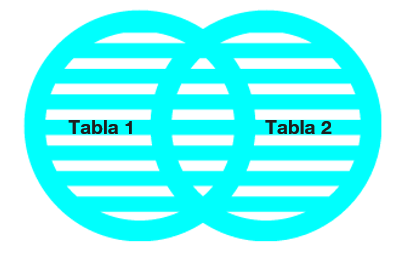
from Personas INNER JOIN Logros

on Personas.PersonaID = Logros.PersonaID;

como las tablas estan vinculadas, quiero que me devuelva la columna de la tabla 1 y la columna de la tabla 2 con los registros respectivos, es decir con los que coincidan. Se vinculan por el ID de la persona

Table

Description automatically generated

Full **OUTER JOIN** retorna todas las filas de la tabla de la izquierda (tabla1) y de la tabla de la derecha (tabla2).

El **OUTER JOIN** combina el resultado de **LEFT** y **RIGHT JOIN**. En este caso podemos tener valores **NULL** de ambos lados.

A picture containing icon

Description automatically generated

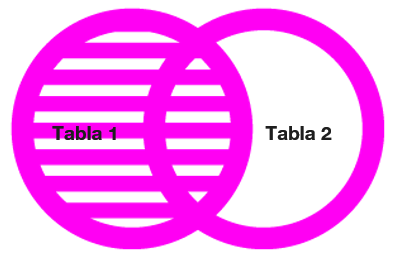
Graphical user interface, application, Teams

Description automatically generated

Cuando hablamos de IZQUIERDA o DERECHA se refiere a lo siguente:

from Staff right join Asignaturas on asignaturas.AsignaturasID=staff.Asignatura

a la izquierda del Join o a la derecha del join.

El **LEFT JOIN** retorna **todas** las **filas** de la **tabla izquierda** (tabla1), con las **filas coincidentes** en la **tabla derecha** (tabla2).

El resultado es **NULL** en “el lado derecho” cuando no hay coincidencia



Siguiendo el primer ejemplo…

--hay encargados que no tienen docentes asignados. En la tabla staff cada docente tiene un encargado

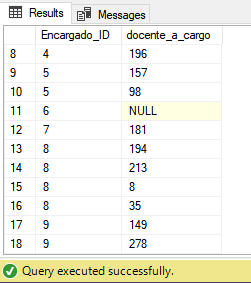
select

encargado.Encargado\_ID,

Staff.DocentesID as docente\_a\_cargo

from Encargado left join staff on staff.Encargado=Encargado.Encargado\_ID

order by Encargado.Encargado\_ID ;

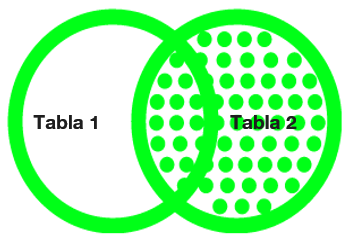


Entonces me devuelve, todos los registros de la tabla que esta a la izquierda del left join (si tienen mas de una coincidencia que me los trae repetidos)

Con los datos coincidentes de la tabla que esta a la derecha del left join, si no hay dato coincidente me pone NULL.

Para este ejemplo, si pusiera right join, me estaría perdiendo de información , porque me devuelve todos los docentes y su respectivo encargado. No hay docentes sin encargado asignado en este caso.

Right join es lo mismo pero con el lado derecho



UNION permite **combinar** el **resultado** de **varias consultas SELECT** al mismo tiempo. Es decir, todas las consultas se ejecutan por separado y luego se concatenan los resultados.

A diferencia del **JOIN**, **no** se **aplica** ningún tipo de **lógica relacional** para unir registros. Sin embargo la cantidad de columnas en cada consulta debe ser la misma, y las columnas de cada tabla deben tener el mismo tipo de datos

Select LogroID, PersonaID, Logro

FROM Logros1

UNION

Select LogroID, PersonaID, Logro

From Logros2

Basicamente esta uniendo tablas que tienen las mismas columnas

Table

Description automatically generated

# Ejemplos Con join

Tengo los nombres del staff junto con el código de la asignatura que les toca en una tabla (staff) y los nombres de las asignaturas con su código respectivo en otra (asignaturas).

Quiero ver el nombre apellido y el nombre directamente (no el ID)

select

staff.Nombre,

staff.apellido,

Asignaturas.nombre as curso\_asignado

from staff inner join Asignaturas

on asignaturas.asignaturasid=staff.asignatura

where LEFT(Apellido,1)='A'

para eso relaciono dos tablas a través de las columnas asignaturasid (PK en Asignaturas) y asignatura (FK en staff).

Tengo dos tablas relacionadas por un ID

select\*from personas

select\*from Logros

SELECT Personas.Apellido, Logros.Logro -- select tabla1.columna1,tabla2.columna2

FROM Personas INNER JOIN Logros -- from tabla1 inner join tabla2

ON Personas.PersonaID = Logros.PersonaID; -- on tabla1.columna3=tabla2.columna4 (columna 3 y 4 son iguales suelen ser ID)

Tedevuelve el logro con el nombre correspondiente. A través del ID que relaciona ambas tablas

Graphical user interface, text, application

Description automatically generated

**INNER JOIN** Retorna todas las filas de las dos tablas siempre y cuando haya coincidencia por el campo declarado en el ON

SELECT Personas.Apellido, Logros.Logro

FROM Personas LEFT JOIN Logros

ON Personas.PersonaID = Logros.PersonaID;

Graphical user interface, text, application

Description automatically generated

**LEFT JOIN** retorna todas las filas de la tabla izquierda, con las filas coincidentes en la tabla derecha

Se refiere a IZQUIERDA o DERECHA de la funcion JOIN

SELECT Personas.Apellido, Logros.Logro

FROM Personas RIGHT JOIN Logros

ON Personas.PersonaID = Logros.PersonaID;

**RIGHT JOIN** retorna todas las filas de la tabla derecha, con las filas coincidentes en la tabla izq.

Graphical user interface, text

Description automatically generated

SELECT Personas.Apellido, Logros.Logro

FROM Personas FULL JOIN Logros

ON Personas.PersonaID = Logros.PersonaID;

Graphical user interface, text, application

Description automatically generated

**OUTER JOIN** combina el resultado de LEFT y RIGHT JOIN. En este caso podemos tener valores **NULL** de ambos lados.

# Ejemplos con union

select\*from Logros1;

select\*from Logros2;

De la siguiente manera uno verticalmente las dos tablas en una sola

Graphical user interface, table

Description automatically generated

Select IDLogro, PersonaID, Logro

FROM Logros1

UNION

Select IDLogro, PersonaID, Logro

From Logros2;

select\*from Logros1

UNION

select\*from Logros2;

# Ejercicios con join y unión.

--1.Indicar por jornada la cantidad de docentes que dictan y sumar los costos. Esta información solo se desea visualizar para las asignaturas de desarrollo web. El resultado debe contener todos los valores registrados en la primera tabla, Renombrar la columna del calculo de la cantidad de docentes como cant\_docentes y la columna de la suma de los costos como suma\_total. Keywords: Asignaturas,Staff, DocentesID, Jornada, Nombre,costo.

select\*from staff

select\*from Asignaturas

select

count (t1.DocentesID) as Cant\_docentes,

t2.Jornada

from staff t1

inner join asignaturas t2

on t1.Asignatura=t2.AsignaturasID

where t2.nombre='desarrollo web'

group by t2.Jornada

hasta aca solo te cuenta los docentes por asignatura de desarrollo web.

select

t2.Jornada,

count (t1.DocentesID) as Cant\_docentes,

sum (t2.costo) as suma\_total

from staff t1

inner join asignaturas t2

on t1.Asignatura=t2.AsignaturasID

where t2.nombre='desarrollo web'

group by t2.Jornada

Table

Description automatically generated

--2.Se requiere saber el id del encargado, el nombre, el apellido y cuantos son los docentes que tiene asignados cada encargado. Luego filtrar los encargados que

--tienen como resultado 0 ya que son los encargados que NO tienen asignado un docente. Renombrar el campo de la operación como Cant\_Docentes. Keywords: Docentes\_id,Encargado, Staff, Nombre, Apellido,Encargado\_ID.

select

Encargado as ID\_Encargado,

count (docentesid) as Docentes\_asignados

from Staff

group by Encargado

--si quiero tener los datos del encargado, debo relacionar con la otra tabla

select

t2.Encargado\_ID,

t2.Nombre,

t2.Apellido,

count(t1.DocentesID) as Docentes\_acargo

from staff t1 left join Encargado t2

on t1.Encargado=t2.Encargado\_ID

group by t2.Encargado\_ID,t2.Nombre,t2.Apellido

having count(t1.DocentesID)=0

# Usar el JOIN para FILTRAR donde NO HAY COINCIDENCIA

--3. Se requiere saber todos los datos de asignaturas que no tienen un docente asignado.El modelo de la consulta debe partir desde la tabla

--staff. Keywords: Staff, Encargado, Asignaturas, costo, Area.

select DocentesID,Asignatura from staff

select\*from Asignaturas

Cada asignatura tiene un código. En la tabla asigaturas tengo la info de cada una de ellas.

Lo qe le estoy pidiendo en el próximo bloque es : seleccioname todos los campos de la tabla 2, y traeme aquellos registros que NO aparecen en la columna asignatura de la tabla 1

Tabla de la izquierda

Tabla de la derecha

select

t2.\*

from Staff t1 right join Asignaturas t2 on t1.Asignatura=t2.AsignaturasID

where t1.Asignatura is null

**RIGHT JOIN** retorna todas las filas de la tabla derecha, con las filas **coincidentes** (según el **ON**) en la tabla izq. Y según la condicion dada en el where

Si escribo el bloque de la siguiente manera:

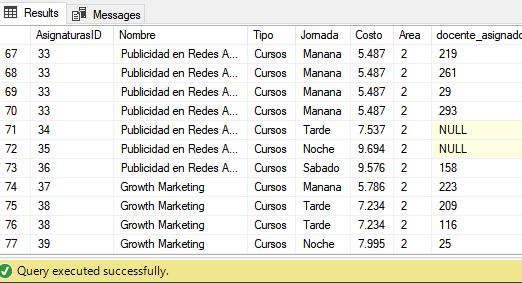
--para entenderlo mejor... puede no haber docentes asignados, en ese caso devuelve null.

select

t2.\*,

t1.DocentesID as docente\_asignado

from Staff t1 right join Asignaturas t2 on t2.AsignaturasID=t1.Asignatura



Donde no hay coincidencia me devuelve NULL. Por eso en el ejercicio agrego “where t1.asignatura is NULL “ . para que me filtre solo aquellos resultados donde no hubo conincidencia

# Vincular mas de una tabla con join

--4.Se quiere conocer la siguiente información de los docentes. El nombre completo concatenar el nombre y el apellido. Renombrar NombresCompletos,

--el documento, hacer un cálculo para conocer los meses de ingreso. Renombrar meses\_ingreso, el nombre del encargado. Renombrar NombreEncargado,

--el teléfono del encargado. Renombrar TelefonoEncargado, el nombre del curso o carrera, la jornada y el nombre del área. Solo se desean visualizar

--solo los que llevan más de 25 meses. Ordenar los meses de ingreso de mayor a menor. Keywords: Encargo, Area, Staff, jornada, fecha ingreso.

select

concat (nombre,' ',Apellido) as NombreCompleto,

Documento,

datename (month,[Fecha Ingreso]) as MesIngresado,

Encargado as NombreEncargado

from staff

Table

Description automatically generatedlo que obtengo hasta aca

En la tabla staff lo que tengo del encargado es su ID. Si quiero que aparezca el nombre debo vincular con otra tabla

select

concat (t1.nombre,' ',t1.Apellido) as NombreCompleto,

t1.Documento,

datename (month,[Fecha Ingreso]) as MesIngresado,

DATEDIFF(MONTH,t1.[Fecha Ingreso],GETDATE()) AS mesesdetrabajo,

concat (t2.Nombre,' ',t2.Apellido) as NombreEncargado,

t2.Telefono as Tel\_Encargado

from staff t1 inner join Encargado t2

on t1.Encargado=t2.Encargado\_ID

--NOTA en un join se puede usar concat con registros de dos tablas diferentes

Table

Description automatically generated

select

concat (t1.nombre,' ',t1.Apellido) as NombreCompleto,

t1.Documento,

datename (month,[Fecha Ingreso]) as MesIngresado,

DATEDIFF(MONTH,t1.[Fecha Ingreso],GETDATE()) AS mesesdetrabajo,

concat (t2.Nombre,' ',t2.Apellido) as NombreEncargado,

t2.Telefono as Tel\_Encargado,

t3.Nombre as Curso\_Carrera,

t3.Jornada,

t4.nombre as Area

from staff t1

inner join Encargado t2 on t1.Encargado=t2.Encargado\_ID

inner join Asignaturas t3 on t1.asignatura=t3.AsignaturasID

inner join Area t4 on t3.Area=t4.AreaID

where DATEDIFF(MONTH,t1.[Fecha Ingreso],GETDATE())>25

order by mesesdetrabajo desc

Asi lo resolvio en clase:

select

concat (t1.nombre,' ',t1.Apellido) as NombreCompleto,

t1.Documento,

datename (month,[Fecha Ingreso]) as MesIngresado,

DATEDIFF(MONTH,t1.[Fecha Ingreso],GETDATE()) AS mesesdetrabajo,

concat (t2.Nombre,' ',t2.Apellido) as NombreEncargado,

t2.Telefono as Tel\_Encargado,

t3.Nombre as Curso\_Carrera,

t3.Jornada,

t4.nombre as Area

from staff t1

inner join Encargado t2 on t1.Encargado=t2.Encargado\_ID

inner join Asignaturas t3 on t1.asignatura=t3.AsignaturasID

inner join Area t4 on t3.Area=t4.AreaID

GROUP by

t1.Documento,t1.Nombre,t1.Apellido,t1.[Fecha Ingreso],t2.Nombre,t2.Apellido,t2.Telefono,t3.Nombre,t3.Jornada,t4.Nombre

HAVING DATEDIFF(MONTH,t1.[Fecha Ingreso],GETDATE())>25

order by mesesdetrabajo desc

--5.Se requiere una listado unificado con nombre, apellido, documento y una marca indicando a que base corresponde. Renombrar como Marca Keywords: Encargo, Staff, Estudiantes

select Nombre,Apellido,Documento,'Encargado' as Marca

from Encargado

UNION

select Nombre,Apellido,Documento,'Estudiantes' as Marca

from Estudiantes

UNION

select Nombre,Apellido,Documento,'Staff' as Marca

from Staff

Table

Description automatically generated