



## Tema 2: JOIN

### JOIN y NATURAL JOIN

El JOIN es una operación que hace coincidir los registros de una tabla con los registros de otra, a partir de uno o más campos, de manera que las columnas puedan ser colocadas lado a lado en el resultado de la consulta, simulando que vienen de una sola tabla. La operación JOIN realiza una unión de tablas por medio de columnas en común.

Veamos qué pasa si aplicamos el operador JOIN a las tablas Profesor y Curso..

**Tabla 14: Operación JOIN en la tabla PROFESOR**

	id	nombre	apellido	fecha_nacimiento	salario
▶	1	Juan	Pérez	1990-06-06	55000
	2	María Emilia	Paz	1984-07-15	72000
	3	Martín	Correa	1987-12-07	63000
	4	Lucía	Díaz	1991-02-24	45000
	5	Raúl	Martínez	1980-10-15	85000
	6	Mabel	Ríos	1982-06-12	83000

Fuente: elaboración propia

**Tabla 15: Operación JOIN en la tabla CURSO**

	codigo	nombre	descripcion	cupo	turno	PROFESOR_id
▶	101	Algoritmos	Algoritmos y estructuras de datos	20	Mañana	1
	102	Matemática Discreta	NULL	20	Tarde	2
	103	Programación Java	POO en Java	35	Noche	4
	104	Programación Web	NULL	35	Noche	5
	105	Programación C#	.NET, Visual Studio 2019	30	Noche	6

Fuente: elaboración propia

Para este caso, el campo que relaciona ambas tablas es PROFESOR\_id. Sabiendo eso, se toma un registro de la tabla PROFESOR y se verifica que su valor (1, por ejemplo), exista en el campo PROFESOR\_id de la tabla CURSO. Debido a que el valor existe, al registro seleccionado se le agregan las columnas del registro coincidente de la otra tabla.



**Tabla 16: Operación JOIN entre tablas PROFESOR y CURSO (1)**

	id	nombre	apellido	fecha_nacimiento	salario	codigo	nombre	descripcion	cupo	turno	PROFESOR_id
▶	1	Juan	Pérez	1990-06-06	55000	101	Algoritmos	Algoritmos y estructuras de datos	20	Mañana	1

Fuente: elaboración propia

De la misma manera, se toma el segundo registro de PROFESOR y se verifica que el id = 2 exista en CURSO. Debido a que existe, se agrega un registro más al resultado, con los campos de PROFESOR y CURSO.

**Tabla 17: Operación JOIN entre tablas PROFESOR y CURSO (2)**

	id	nombre	apellido	fecha_nacimiento	salario	codigo	nombre	descripcion	cupo	turno	PROFESOR_id
▶	1	Juan	Pérez	1990-06-06	55000	101	Algoritmos	Algoritmos y estructuras de datos	20	Mañana	1
	2	María Emilia	Paz	1984-07-15	72000	102	Matemática Discreta	NULL	20	Tarde	2

Fuente: elaboración propia

Así continúa la operación hasta que finalmente se obtiene lo siguiente.

**Tabla 18: Operación JOIN entre tablas PROFESOR y CURSO (3)**

	id	nombre	apellido	fecha_nacimiento	salario	codigo	nombre	descripcion	cupo	turno	PROFESOR_id
▶	1	Juan	Pérez	1990-06-06	55000	101	Algoritmos	Algoritmos y estructuras de datos	20	Mañana	1
	2	María Emilia	Paz	1984-07-15	72000	102	Matemática Discreta	NULL	20	Tarde	2
	4	Lucía	Díaz	1991-02-24	45000	103	Programación Java	POO en Java	35	Noche	4
	5	Raúl	Martínez	1980-10-15	85000	104	Programación Web	NULL	35	Noche	5
	6	Mabel	Ríos	1982-06-12	83000	105	Programación C#	.NET, Visual Studio 2019	30	Noche	6

Fuente: elaboración propia

Es importante notar que el profesor “Martín Correa” no fue incluido en el resultado debido a que no dicta ningún curso.

**El ejemplo anterior puede escribirse de la siguiente forma:**

```
select *  
from profesor p join curso c on p.id = c.PROFESOR_id;
```



Cómo usamos solo la palabra “JOIN”, solo obtendremos los registros que coinciden entre las tablas. Con el uso de la palabra reservada “on” especificamos cuales son los campos de cada tabla que deben coincidir.

Si en ambas tablas la columna para realizar el JOIN tuviera el mismo nombre, podríamos usar el NATURAL JOIN qué es un tipo de JOIN que hace coincidir automáticamente las filas de aquellas columnas con el mismo nombre, por lo que no es necesario especificar las columnas de cruce con el ON.

Por ejemplo, si la columna id de la tabla PROFESOR se llamará “PROFESOR\_id”, la consulta anterior se simplificaría a:

```
select *  
from profesor p natural join curso c;  
que sería lo mismo que escribir:  
select *  
from profesor p join curso c on p. PROFESOR_id; =  
c.PROFESOR_id;
```

## INNER JOIN

El INNER JOIN tiene el mismo funcionamiento que el NATURAL JOIN y que el JOIN, con la diferencia de que para utilizar el INNER JOIN es necesario especificar las columnas de cruce en ambas tablas. En todos los casos, se obtiene el mismo resultado: solo los registros cuyos valores en el campo de cruce coincidan.

La consulta del punto anterior podría escribirse como:

```
select *  
from profesor inner join curso on profesor.id = curso.PROFESOR_id;
```

Si no queremos obtener todas las columnas de todas las tablas, podemos especificar dentro del SELECT, cuales queremos:

```
select profesor.*, curso.nombre  
from profesor inner join curso on profesor.id = curso.PROFESOR_id;
```

Con esa sentencia le indicamos a SQL que queremos todos los datos del profesor, pero solo el nombre del curso que dicta cada uno.



Tabla 19: Operación INNER JOIN

	id	nombre	apellido	fecha_nacimiento	salario	nombre
▶	1	Juan	Pérez	1990-06-06	55000	Algoritmos
	2	María Emilia	Paz	1984-07-15	72000	Matemática Discreta
	4	Lucía	Díaz	1991-02-24	45000	Programación Java
	5	Raúl	Martínez	1980-10-15	85000	Programación Web
	6	Mabel	Ríos	1982-06-12	83000	Programación C#

Fuente: elaboración propia

## LEFT JOIN

Otro tipo de JOIN muy utilizado es el LEFT JOIN que retorna todos los registros de la primera tabla más los registros de la segunda tabla cuyo valor para el campo de cruce coincida con el de la primera tabla.

Por ejemplo, si realizamos un LEFT JOIN entre PROFESOR y CURSO:

```
select *  
from profesor left join curso on profesor.id = curso.PROFESOR_id;
```

Tabla 20: Operación LEFT JOIN

	id	nombre	apellido	fecha_nacimiento	salario	codigo	nombre	descripcion	cupo	turno	PROFESOR_id
▶	1	Juan	Pérez	1990-06-06	55000	101	Algoritmos	Algoritmos y estructuras de datos	20	Mañana	1
	2	María Emilia	Paz	1984-07-15	72000	102	Matemática Discreta	NOVA	20	Tarde	2
	3	Martín	Correa	1987-12-07	63000	NOVA	NOVA	NOVA	NOVA	NOVA	NOVA
	4	Lucía	Díaz	1991-02-24	45000	103	Programación Java	POO en Java	35	Noche	4
	5	Raúl	Martínez	1980-10-15	85000	104	Programación Web	NOVA	35	Noche	5
	6	Mabel	Ríos	1982-06-12	83000	105	Programación C#	.NET, Visual Studio 2019	30	Noche	6

Fuente: elaboración propia

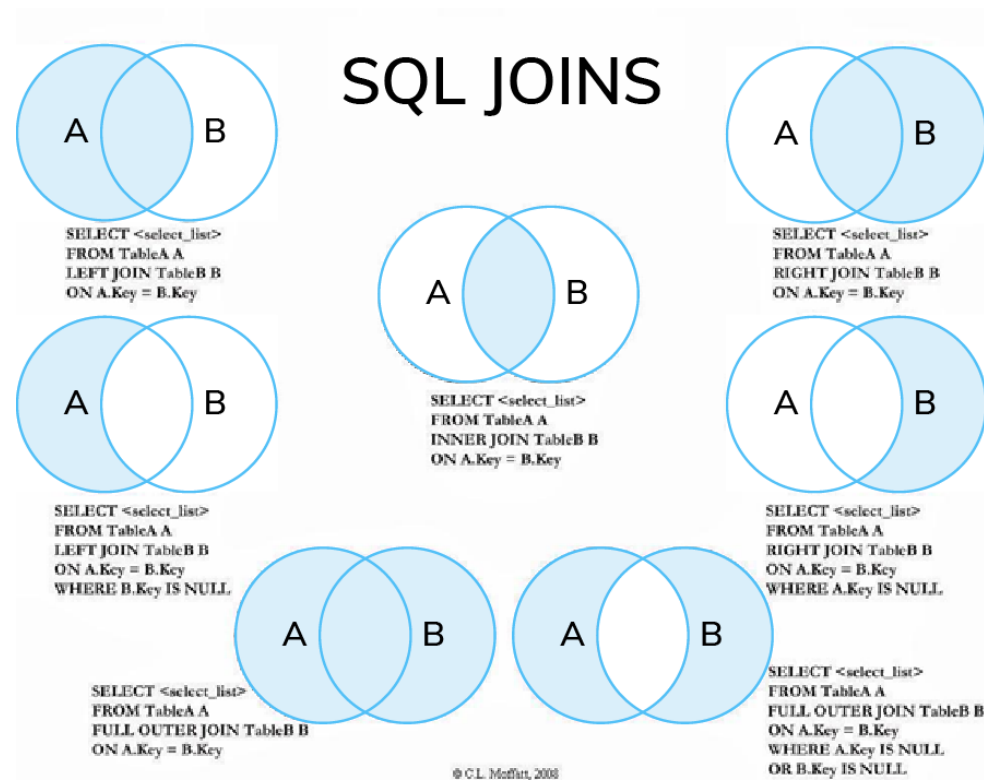
Obtenemos todos los profesores y los cursos que dictan. Para el caso del profesor “Martín Correa” que no dicta cursos, los campos de la tabla curso están vacíos.

## Otros tipos de JOIN



Existen muchos tipos de JOIN. Podemos ilustrarlos a través de la siguiente imagen.

**Figura 5: Tipos de JOIN**



Fuente: [imagen sin título sobre tipos de JOIN], 2013, <https://bit.ly/3kBi0kN>

En la imagen se puede ver la sintaxis de cada tipo de JOIN y un gráfico de los resultados que retornan.

## Documentación

### Principales tipos de JOIN con ejemplos

Fuente: **W3 Schools**, (s.f.). SQL Joins. Recuperado de [https://www.w3schools.com/sql/sql\\_join.asp](https://www.w3schools.com/sql/sql_join.asp)

## Videos

### JOIN con MySQL Server

Fuente: **Edu 4 Java** [edu4java]. (13 de noviembre de 2010). Select join con mysql workbench. Video Tutorial 5 SQL en español



[YouTube]. Recuperado de  
[https://www.w3schools.com/sql/sql\\_join.asp](https://www.w3schools.com/sql/sql_join.asp)

## Ejercicio 2

Dadas las siguientes tablas:

**Tabla 21: PROFESOR**

	id	nombre	apellido	fecha_nacimiento	salario
►	1	Juan	Pérez	1990-06-06	55000
	2	María Emilia	Paz	1984-07-15	72000
	3	Martín	Correa	1987-12-07	63000
	4	Lucía	Díaz	1991-02-24	45000
	5	Raúl	Martínez	1980-10-15	85000
	6	Mabel	Ríos	1982-06-12	83000

Fuente: elaboración propia

**Tabla 22: CURSO**

	codigo	nombre	descripcion	cupos	turno	PROFESOR_id
►	101	Algoritmos	Algoritmos y estructuras de datos	20	Mañana	1
	102	Matemática Discreta	NULL	20	Tarde	2
	103	Programación Java	POO en Java	35	Noche	4
	104	Programación Web	NULL	35	Noche	5
	105	Programación C#	.NET, Visual Studio 2019	30	Noche	6

Fuente: elaboración propia

Donde el campo PROFESOR\_id de la tabla CURSO es una clave foránea que referencia a PROFESOR, responde las siguientes preguntas.

- 1) ¿Es posible aplicar un NATURAL JOIN entre esas tablas?
  - a. Sí
  - b. No. No tienen campos con el mismo nombre**
- 2) ¿Cuál de las siguientes consultas devuelve el resultado?

**Tabla 23: Ejercicio**



	Apellido	Nombre	Turno
►	Pérez	Algoritmos	Mañana
	Paz	Matemática Discreta	Tarde
	Díaz	Programación Java	Noche
	Martínez	Programación Web	Noche
	Ríos	Programación C#	Noche

Fuente: elaboración propia

a. `SELECT * FROM PROFESOR  
INNER JOIN CURSO ON PROFESOR.id =  
CURSO.PROFESOR_id;`

`SELECT PROFESOR.Apellido, CURSO.Nombre ,  
CURSO.Turno`

b. `FROM PROFESOR LEFT JOIN CURSO ON PROFESOR.id =  
CURSO.PROFESOR_id;`

c. `SELECT PROFESOR.Apellido, CURSO.Nombre,  
CURSO.Turno  
FROM PROFESOR INNER JOIN CURSO ON PROFESOR.id  
= CURSO.PROFESOR_id;`

3) Indica, en el recuadro de texto, qué resultado se obtiene después de ejecutar:

`SELECT profesor.Apellido , curso.Nombre, curso.turno  
FROM profesor LEFT JOIN curso ON profesor.id =  
curso.PROFESOR_id;`