

**Nombre: Iris Lizeth
Sanchez Garcia
Universidad de Los
Ángeles**

**Docente: I.S.C
Martin Cruz
Medinilla**

Semestre: Cuarto

**Actividad:
Investigación**

JOINS EN SQL

Los "joins" en SQL son operaciones que permiten combinar datos de múltiples tablas en una consulta. Estas operaciones son fundamentales en bases de datos relacionales, donde la información se organiza en tablas y la relación entre estas tablas es crucial para extraer datos relevantes. Los joins se basan en la idea de establecer una relación entre dos o más conjuntos de datos utilizando una clave común, que puede ser una columna compartida entre las tablas.

Al unir tablas mediante joins, puedes vincular filas de una tabla con filas de otra tabla según una condición de igualdad entre columnas. Esta condición se especifica en la cláusula de join, y las filas que cumplen con dicha condición se combinan en el resultado de la consulta.

Los joins permiten realizar consultas complejas que involucran múltiples conjuntos de datos relacionados. Puedes recuperar información detallada de varias tablas en una sola consulta, evitando la necesidad de realizar múltiples consultas y fusionar los resultados manualmente. Además, los joins pueden utilizarse para filtrar datos, ordenar resultados y realizar cálculos basados en la información combinada de las tablas.

Tipos de joins en sql

1. **INNER JOIN:** Devuelve solo las filas que tienen al menos una coincidencia en ambas tablas basadas en la condición de unión especificada.
2. **LEFT JOIN (o LEFT OUTER JOIN):** Devuelve todas las filas de la tabla de la izquierda (primera tabla mencionada en la consulta) y las filas coincidentes de la tabla de la derecha (segunda tabla mencionada en la consulta). Si no hay coincidencias en la tabla de la derecha, se devuelve NULL en las columnas de la tabla de la derecha.
3. **RIGHT JOIN (o RIGHT OUTER JOIN):** Similar al LEFT JOIN, pero devuelve todas las filas de la tabla de la derecha y las filas coincidentes de la tabla de la izquierda. Si no hay coincidencias en la tabla de la izquierda, se devuelve NULL en las columnas de la tabla de la izquierda.
4. **FULL JOIN (o FULL OUTER JOIN):** Devuelve todas las filas cuando hay una coincidencia en una de las tablas. Si no hay coincidencias, se devuelven NULL en las columnas de la tabla sin coincidencia.
5. **CROSS JOIN:** Devuelve el producto cartesiano de las tablas involucradas, es decir, combina cada fila de la primera tabla con cada fila de la segunda tabla. No hay una condición de unión en un CROSS JOIN.

Ejemplos de JOINS en SQL

- **INNER JOIN:** Este JOIN devuelve las filas cuando hay una coincidencia en ambas tablas.

```
SELECT Empleados.Nombre, Departamentos.Nombre AS Departamento  
FROM Empleados INNER JOIN Departamentos ON  
Empleados.ID_Departamento = Departamentos.ID;
```

- **LEFT JOIN:** Este JOIN devuelve todas las filas de la tabla de la izquierda (Empleados), incluso si no hay coincidencias en la tabla de la derecha (Departamentos).

```
SELECT Empleados.Nombre, Departamentos.Nombre AS Departamento  
FROM Empleados LEFT JOIN Departamentos ON  
Empleados.ID_Departamento = Departamentos.ID;
```

- **RIGHT JOIN:** Este JOIN devuelve todas las filas de la tabla de la derecha (Departamentos), incluso si no hay coincidencias en la tabla de la izquierda (Empleados).

```
SELECT Empleados.Nombre, Departamentos.Nombre AS Departamento  
FROM Empleados RIGHT JOIN Departamentos ON  
Empleados.ID_Departamento = Departamentos.ID;
```

- **FULL JOIN:** Este JOIN devuelve todas las filas de ambas tablas, coincidentes o no. Si no hay coincidencias, las columnas de la tabla sin coincidencia contendrán valores NULL.

```
SELECT Empleados.Nombre, Departamentos.Nombre AS Departamento  
FROM Empleados FULL JOIN Departamentos ON  
Empleados.ID_Departamento = Departamentos.ID;
```

- **CROSS JOIN:** Este JOIN devuelve el producto cartesiano de las dos tablas, es decir, combina cada fila de la primera tabla con cada fila de la segunda tabla.

```
SELECT Empleados.Nombre, Departamentos.Nombre AS Departamento  
FROM Empleados CROSS JOIN Departamentos;
```

Conclusión

En conclusión, puedo decir con seguridad que si los joins fueran personas, serían los maestros de ceremonias en la fiesta de la base de datos. Con su habilidad para unir tablas como si fueran cupidos de la información, hacen que incluso las relaciones más complejas entre datos parezcan un juego de niños.