

11. Lectura Inner y Outer Joins en SQL

En SQL, las **uniones (joins)** permiten combinar registros de dos o más tablas basándose en una condición relacionada.

Tipos Principales de Joins:

- **INNER JOIN:**
Combina registros de dos tablas siempre que haya concordancia de valores en un campo común.
- **OUTER JOIN:**
Incluye registros que pueden no tener correspondencia en la otra tabla, subdividiéndose en:
- **LEFT JOIN (o LEFT OUTER JOIN):**
Incluye todos los registros de la tabla izquierda y los registros coincidentes de la tabla derecha.
- **RIGHT JOIN (o RIGHT OUTER JOIN):**
Incluye todos los registros de la tabla derecha y los registros coincidentes de la tabla izquierda.
- **INNER JOIN (Unión Interna)**
Devuelve solo las filas donde existe una coincidencia en ambas tablas

Sintaxis.sql

```
1  SELECT columnas
2  FROM tabla1 INNER JOIN tabla2
3  ON tabla1.columna_comun = tabla2.columna_comun;
4
```

Ejemplo.sql

```
1
2  SELECT NombreCategoria, NombreProducto
3  FROM Categorías INNER JOIN Productos
4  ON Categorías.IDCategoria = Productos.IDCategoria;
5
6
```

En este ejemplo, se seleccionan los nombres de categorías y productos donde el `IDCategoria` coincide en ambas tablas.

Consideraciones:

- Los campos combinados deben ser del mismo tipo de dato, pero no necesariamente deben tener el mismo nombre.
- Si se intenta combinar campos que contienen datos de tipo `Memo` u `Objeto OLE`, se producirá un error.

LEFT JOIN y RIGHT JOIN (Uniones Externas)

Los **OUTER JOIN** permiten incluir registros que no tienen correspondencia en la otra tabla.

LEFT JOIN:

Incluye todos los registros de la tabla izquierda y los registros coincidentes de la tabla derecha. Si no hay coincidencia, las columnas de la tabla derecha contendrán `NULL`.

Sintaxis.sql

```
1  SELECT columnas
2  FROM tabla1 LEFT JOIN tabla2
3  ON tabla1.columna_comun = tabla2.columna_comun;
4
```

RIGHT JOIN:

Incluye todos los registros de la tabla derecha y los registros coincidentes de la tabla izquierda. Si no hay coincidencia, las columnas de la tabla izquierda contendrán `NULL`.

Sintaxis.sql

```
1  SELECT columnas
2  FROM tabla1 RIGHT JOIN tabla2
3  ON tabla1.columna_comun = tabla2.columna_comun;
```

Ejemplo de LEFT JOIN:

Ejemplo_Left.sql

```
1  SELECT Departamentos.NombreDepartamento, Empleados.NombreEmpleado
2  FROM Departamentos LEFT JOIN Empleados
3  ON Departamentos.IDDepartamento = Empleados.IDDepartamento;
4
```

Este ejemplo selecciona todos los departamentos, incluyendo aquellos que no tienen empleados asignados.

Consideraciones:

- La sintaxis puede variar según el sistema de gestión de bases de datos. Por ejemplo, en SQL Server es común utilizar `LEFT OUTER JOIN` y `RIGHT OUTER JOIN`, mientras que en otros sistemas la palabra `OUTER` es opcional.

Anidamiento de Joins

Es posible anidar múltiples `JOIN` en una instrucción SQL para combinar más de dos tablas.

Ejemplo.sql

```
1  SELECT campos
2  FROM tabla1 INNER JOIN (tabla2 INNER JOIN tabla3
```

```

3  ON tabla2.campo_comun = tabla3.campo_comun)
4  ON tabla1.campo_comun = tabla2.campo_comun;
5

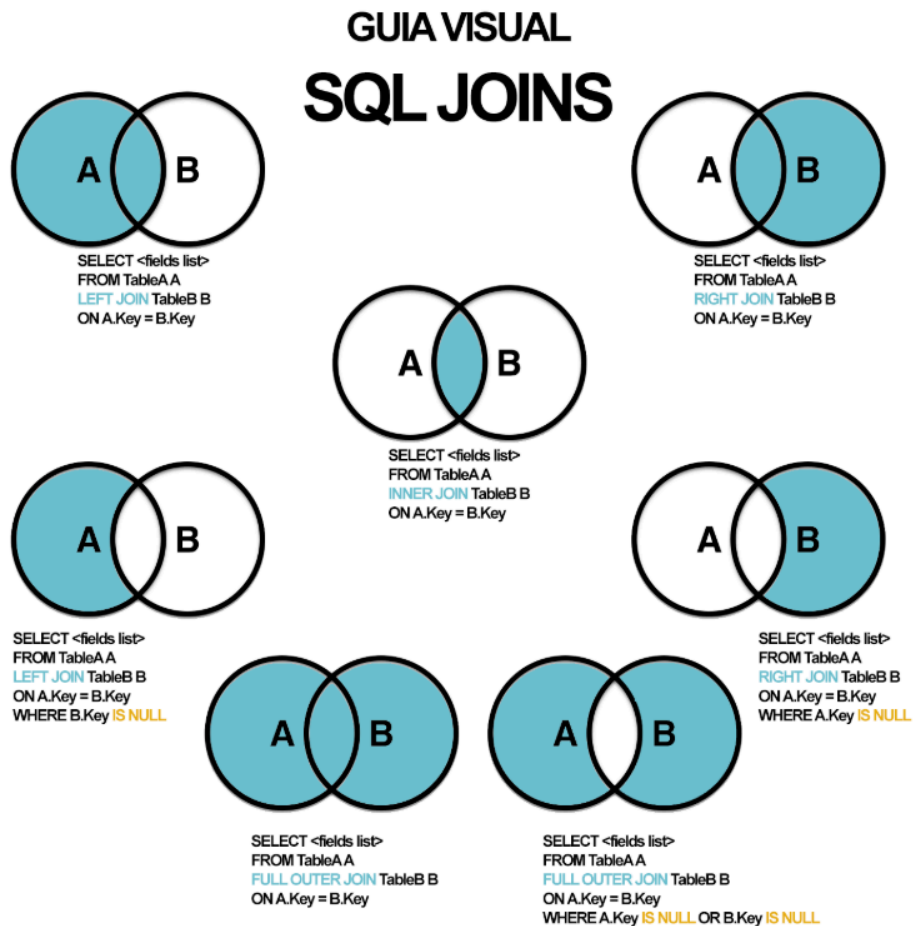
```

En este ejemplo, primero se realiza una unión entre `tabla2` y `tabla3`, y luego el resultado se une con `tabla1`.

Consideraciones:

- Un `LEFT JOIN` o `RIGHT JOIN` puede anidarse dentro de un `INNER JOIN`, pero un `INNER JOIN` no puede anidarse dentro de un `LEFT JOIN` o `RIGHT JOIN`.

Guía visual



Dudas

¿Qué operadores de comparación se pueden usar en una cláusula `JOIN` ?

¿Qué sucede si se combinan campos de diferentes tipos de datos en un `JOIN` ?