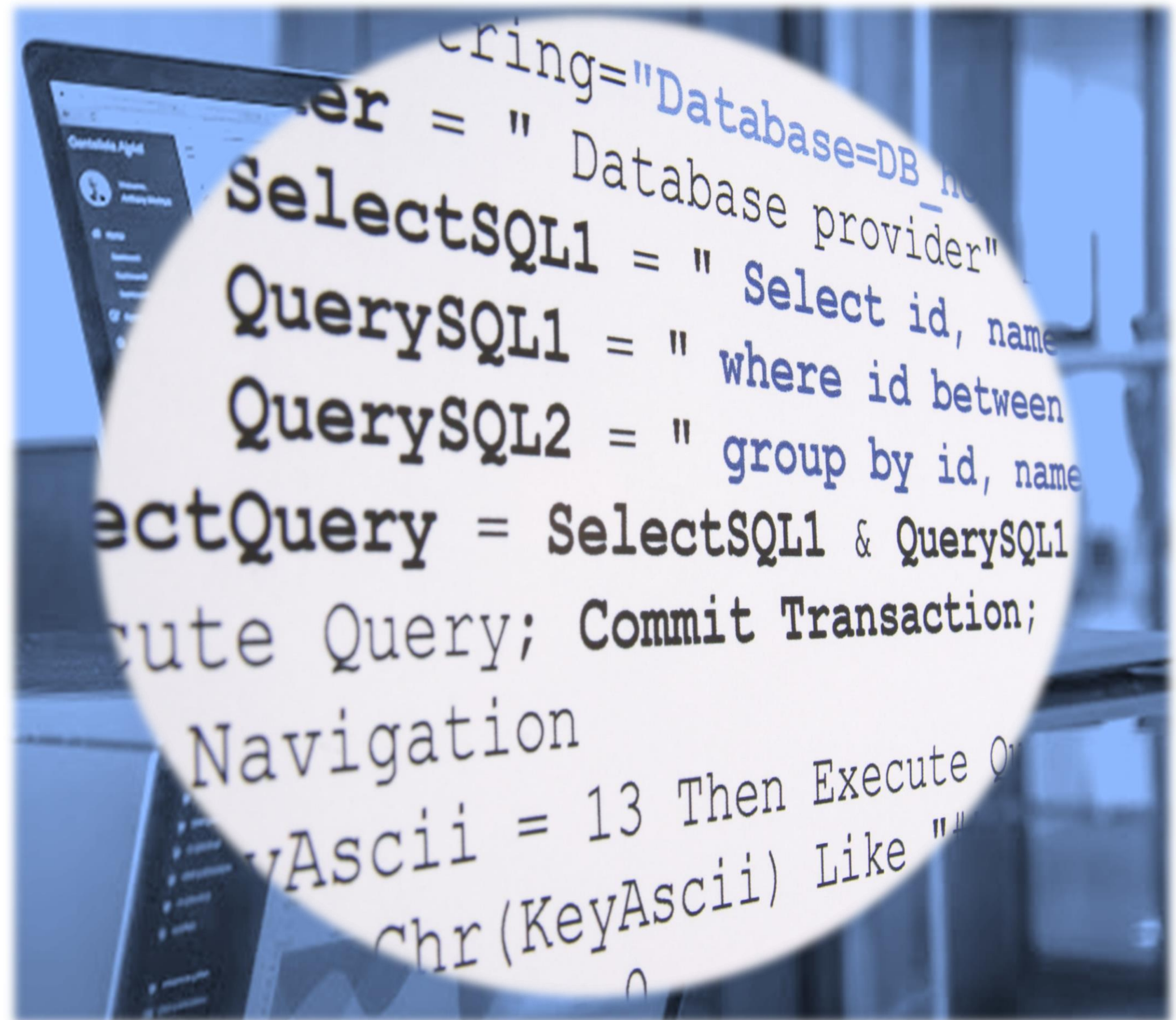


# SQL Server



---

## Sesión 2



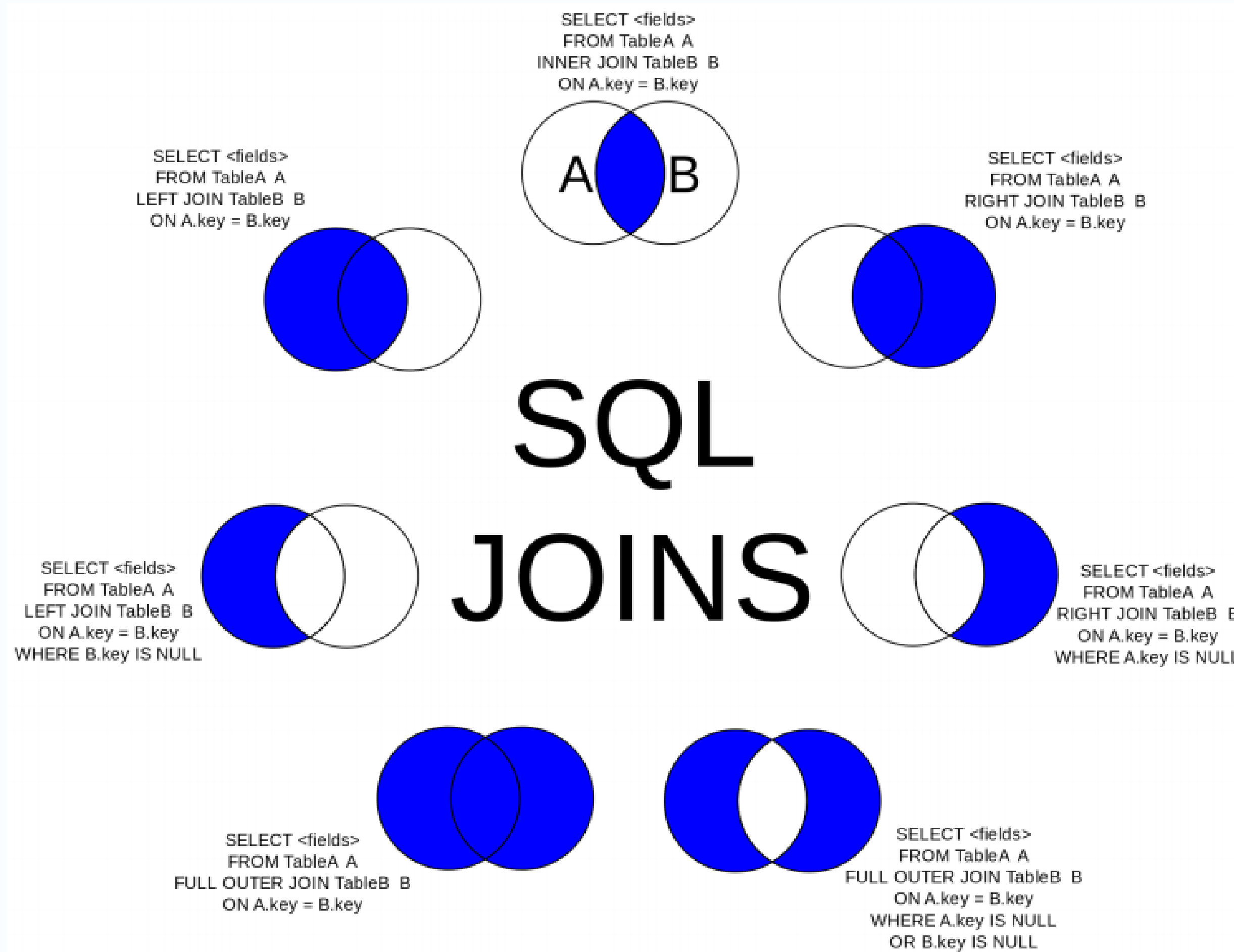
# Sesión 2 | Objetivo

- Identificar escenarios con Joins
- Querys con Inner Joins
- Querys con Outer Joins
- Querys con Cross Joins
- Querys con \*Self Joins
- Querys con ORDER BY
- Querys con WHERE
- Querys con TOP
- Querys con NULL



- Cuando se utiliza la clausula FROM se indica los datos que serán utilizados en el SELECT.
- El FROM puede contener varias tablas y luego estas generar un tabla virtual.

Tipo de Join	Descripción
Cross	Realiza todas las combinaciones(Un producto cartesiano).
Inner	Realiza un producto cartersiano y luego filtra los que cumplen la condición de match.
Outer	Realiza un producto cartersiano, manteniendo todos los valores de la primera tabla asi hagan match o no con la otra tabla de destino.



# Querys com Inner Joins

- Retorna las filas que cumplieron con la condición luego de realizar el producto cartesiano.
- Para utilizar la condición se utiliza la palabra ON

```
SELECT o.orderid,  
       o.orderdate,  
       od.productid,  
       od.unitprice,  
       od.qty  
FROM Sales.Orders AS o  
      INNER JOIN Sales.OrderDetails AS od  
      ON o.orderid = od.orderid;
```



# IT'S TIME TO WORKOUT



# Querys com Outer Joins

- Retorna las filas de la primera tabla y las filas coincidentes de la segunda tabla.
- Todas las filas de la primera tabla son mantenidas.
- Los datos que no hicieron match con la primera tabla devuelven valores NULL.

# Querys con Outer Joins

- Retorna todas las filas de t1 y las que coincidan con t2

```
FROM t1 LEFT OUTER JOIN t2 ON  
t1.col = t2.col
```

- Retorna todas las filas de t2 y las que coincidan con t1

```
FROM t1 RIGHT OUTER JOIN t2 ON  
t1.col = t2.col
```

- Retorna todas las filas de t1 que no hicieron match con t2

```
FROM t1 LEFT OUTER JOIN t2 ON  
t1.col = t2.col  
WHERE t2.col IS NULL
```





# IT'S TIME TO WORKOUT



# Querys con Cross Joins

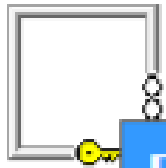
- Retorna todas las combinaciones de filas de la primera tabla con una segunda tabla(un producto catersiano).
- Lo recomendable es no utilizarlo por su alto costo.

```
SELECT ...  
FROM t1 CROSS JOIN t2
```

# Querys con Self Joins

- Es una variante de las otras combinaciones, pero aquí la primera tabla y segunda tabla son la misma tabla.
- Ejemplo: Se quiere retornar a los datos de los empleados así como los datos de su jefe

```
SELECT e.empid, e.lastname,  
       e.title, e.mgrid, m.lastname  
FROM   HR.Employees AS e  
JOIN   HR.Employees AS m  
ON     e.mgrid=m.empid;
```



empid
lastname
firstname
title
titleofcourtesy
birthdate
hiredate
address
city
region
postalcode
country
phone
mgrid

# IT'S TIME TO WORKOUT



- La clausula ORDER BY permite ordenar las filas devueltas luego de aplicar el SELECT
- Si no se utiliza el ORDER BY las filas no necesariamente son devueltas en orden.
- Los ORDER BY deben ir acompañados de :
  - Nombre de la columna
  - Alias de la columna
  - Posición de la columna(no recomendado)
- También tiene se tiene que indicar el orden si será ASC o DESC
- Por defecto el valor es ASC

- En la clausula **WHERE** se indican predicados para realizar comparaciones, en caso el valor sea **TRUE** esas filas serán aceptadas.
- Los valores que van en el **WHERE** son candidatas a convertirse en índices de tablas.

```
SELECT contactname, country  
FROM Sales.Customers  
WHERE country = N'Spain';
```

```
SELECT orderid, orderdate  
FROM Sales.Orders  
WHERE orderdate > '20070101';
```

- TOP
  - La clausula TOP me permite limitar el número o porcentaje de filas que serán retornados en al consulta.
  - Tiene la siguiente estructura:
    - SELECT TOP (N) | TOP (N) PERCENT
    - SELECT TOP (N) WITH TIES

Nota: La clausula TOP es propia de Microsoft



- SQL Server utiliza la clausula NULL para indicar que no existe valor.
- Para realizar filtros en con valores NULL se utilizan las clausulas IS NULL o IS NOT NULL

- No Correcto

```
SELECT Nombre  
FROM Tiendas  
WHERE pais <> NULL;
```

- Correcto

```
SELECT Nombre  
FROM Tiendas  
WHERE pais IS NOT NULL;
```



# TIPS Y RECOMENDACIONES

- Practicar los ejercicios prácticos realizados en clase.
- Tener un orden al realizar las operaciones en SQL.

**Gracias...**

