

Estructura del bloque 'Data Analysis con Python'

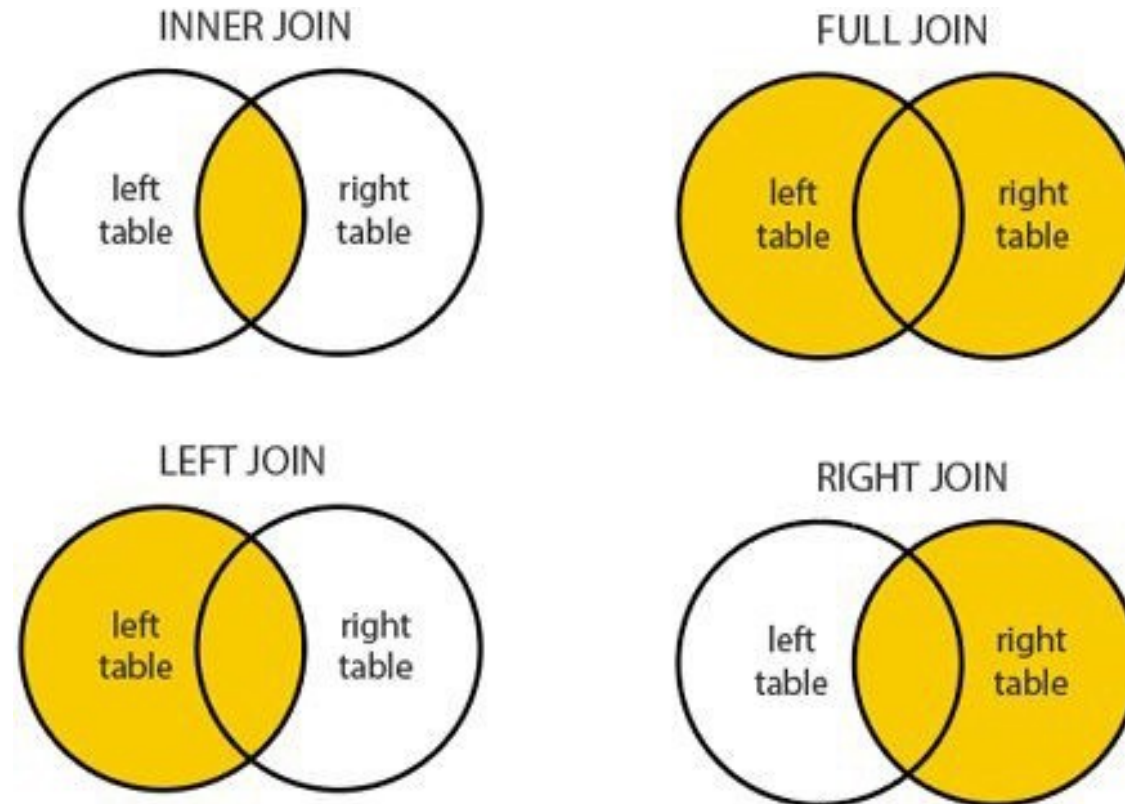
		Extraer, transformar y cargar datos (ETL)	Impartido	1.1	Ciclo I ETL
		Estadística descriptiva I & Azar y probabilidad	Impartido	1.2	
		Excel: preparar los datos & Power Query	Impartido	1.3	
		SQL: "JOINS" y "Queries"		1.4	
		Taller Case Study I		1.5	
Ciclo II Análisis exploratorio I	2.1	Fundamentos análisis exploratorio I & Estadística descriptiva II			Ciclo III Análisis exploratorio II
	2.2	Python: Análisis exploratorio			
	2.3	Taller Case Study II			
		Fundamentos análisis exploratorio II & Inferencia estadística		3.1	
		Python: Análisis exploratorio II		3.2	
		Taller Case Study III		3.3	Ciclo IV Data mining
	4.1	Fundamentos "Data Mining" & Análisis multivariante			
	4.2	Python: "Data Mining"			
	4.3	Taller Case Study IV			
		Dashboard Case Study			

JOINS en SQL-II

JOINS BÁSICAS

¿Qué es un JOIN en SQL? ¿por qué implementamos JOINS?

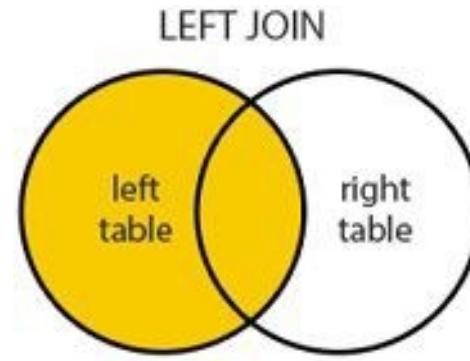
Los JOINS en SQL son utilizados para combinar dos o más tablas en una sola consulta. Esto es útil cuando se necesita acceder a datos que están almacenados en múltiples tablas y que se relacionan entre sí. Existen diferentes tipos de joins, cada uno de ellos con un propósito específico.



JOINS BÁSICAS

LEFT JOIN

Un JOIN de tipo **LEFT** mantiene todas las observaciones de la tabla principal (**LEFT**) e integra las observaciones que coinciden de la tabla secundaria (**RIGHT**).

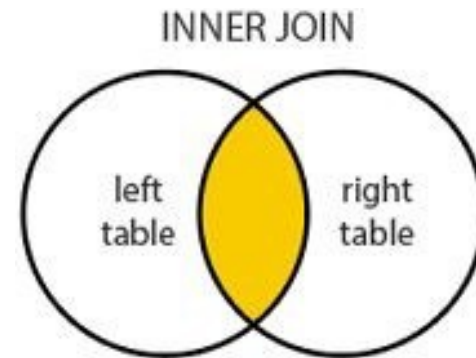


```
SELECT  
T1.`column1`, T2.`column2`  
FROM `table1` T1  
LEFT JOIN `table2` T2  
ON T1.key = T2.key
```

JOINS BÁSICAS

INNER JOIN

Un JOIN de tipo INNER mantiene solo las observaciones que coinciden en ambas tablas (LEFT y RIGHT).

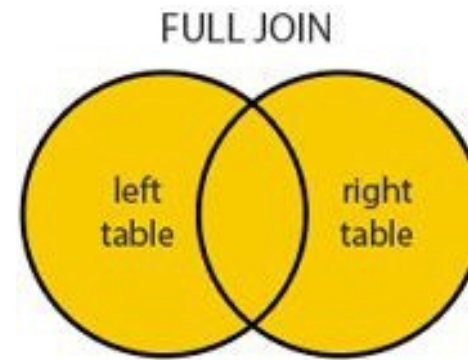


```
SELECT  
T1.`column1`,T2.`column2`  
FROM `table1` T1  
INNER JOIN `table2` T2  
ON T1.key = T2.key
```

JOINS BÁSICAS

FULL JOIN

Un JOIN de tipo FULL mantiene todas las observaciones de ambas tablas (LEFT y RIGHT), sin importar si coinciden o no.

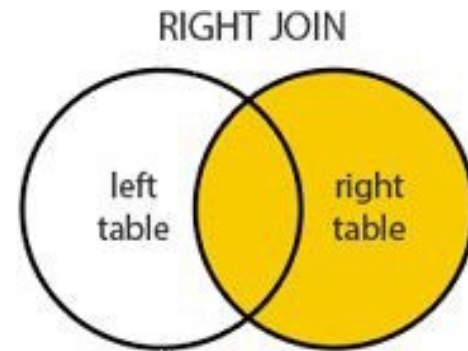


```
SELECT  
T1.`column1`, T2.`column2`  
FROM `table1` T1  
FULL JOIN `table2` T2  
ON T1.key = T2.key
```

JOINS BÁSICAS

RIGHT JOIN

Un JOIN de tipo RIGHT es el opuesto a una LEFT join. Mantiene todas las observaciones de la tabla secundaria (RIGHT) e integra aquellas observaciones de la tabla principal (LEFT) cuyos valores coinciden.



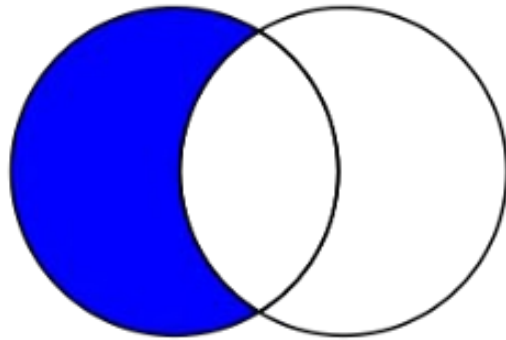
```
SELECT  
T1.`column1`,T2.`column2`  
FROM `table1` T1  
RIGHT JOIN `table2` T2  
ON T1.key = T2.key
```

JOINS AVANZADAS

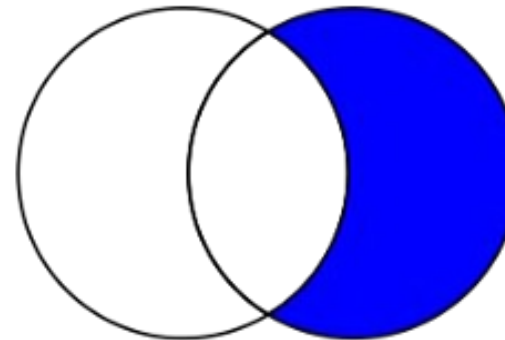
ANTI JOINS

Un anti join en SQL es un tipo de join que devuelve las filas de una tabla que no tienen coincidencias en otra tabla. Se utiliza para encontrar las filas en una tabla que no tienen correspondencia en otra tabla relacionada.

LEFT ANTI



RIGHT ANTI

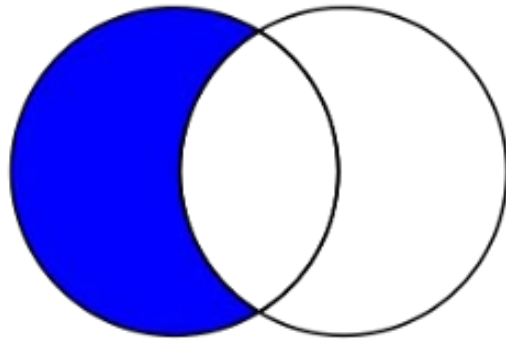


JOINS AVANZADAS

LEFT ANTI

Un LEFT ANTI JOIN devuelve las filas de la tabla de la izquierda que no tienen una correspondencia en la tabla de la derecha.

LEFT ANTI



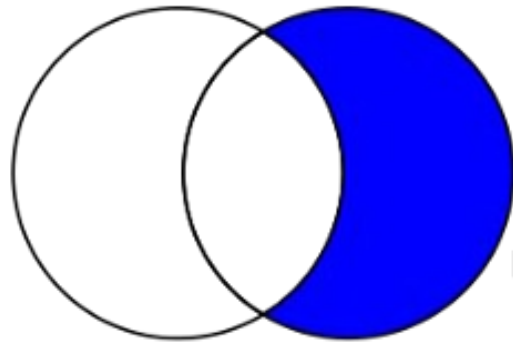
```
SELECT *  
FROM table1  
LEFT JOIN table2  
ON table1.column = table2.column  
WHERE table2.column IS NULL
```

JOINS AVANZADAS

RIGHT ANTI

Un RIGHT ANTI JOIN devuelve las filas de la tabla de la derecha que no tienen una correspondencia en la tabla de la izquierda

RIGHT ANTI



```
SELECT *  
FROM table1  
RIGHT JOIN table2  
ON table1.column = table2.column  
WHERE table1.column IS NULL
```