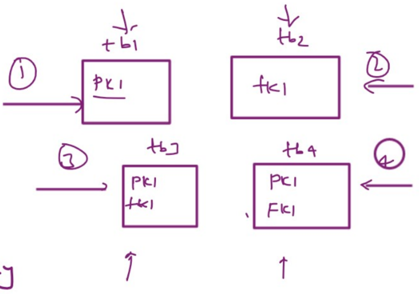
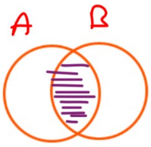
Joins

→ Joins combine rows from two or more tables based on related columns. They allow you to retrieve data from multiple tables in a single query



Types of Joins

(1) Inner join:



→ returns only the rows where there is a match in both tables

→ Most commonly used join

A - col1 → col1 B

	col1	col2
row1	A	X
row2	B	Y
row3	C	Z

	col1	col2
row1	B	2
row2	C	U
row3	D	Y

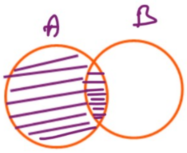
inner join

B	Y	B	Z
C	Z	C	U

Syntax: Select * from table1 INNER JOIN table 2
ON table1.column = table2.column

(2) Left JOIN (Left outer Joins)

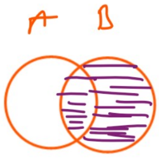
→ returns all the rows from the left table & only matching rows from the right table



→ If no match, NULL values are returned for right table columns

→ SYNTAX: SELECT * from table1 LEFT JOIN table2
ON table1.colum = table2.column;

(3) Right JOIN (right outer join)

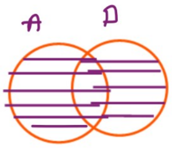


→ Returns all rows from the right table & only matching rows from the left table

→ If no match, NULL values are returned for left table columns

Syntax: SELECT * from table1 RIGHT JOIN table 2
ON table1.column = table2.column

(4) full JOIN (Full outer JOIN)



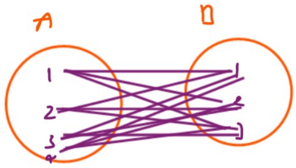
→ Returns all the rows where there is a match in either table

→ mySQL doesn't directly support full Join, but can be simulated with UNION

SYNTAX:

```
SELECT * from table1 where table1.column >18;
UNION
SELECT * from table2 where table2.column = ' ';
```

(5) CROSS JOIN



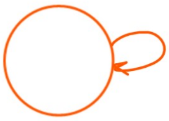
→ returns the Cartesian product (all possible combination of rows)

→ No matching condition is required

Syntax:

```
SELECT * from table1 CROSS JOIN table2;
```

(6) SELF JOIN:



→ JOIN a table to itself

→ used for any hierarchical or comparitive data within the samble table

SYNTAX:

```
SELECT * from table1 AS a JOIN table1 AS b
    ON a.column = b.related_column
```