

Anatomy of a Join

Objects		
<u>id</u>	<u>name</u>	<u>color_id</u>
1	pen	10
2	highlighter	20
3	mug	10

Colors	
<u>id</u>	<u>name</u>
10	red
20	green

```
SELECT *  
FROM Objects, Colors
```

Listing tables creates an unnamed table with the "cross product" of the rows. You combined each row in the left table with each in the right table. This multiplies the number of rows (here $3 \times 2 = 6$)

(unnamed join table)					
(row #)	<u>Objects.id</u>	<u>Objects.name</u>	<u>Objects.color_id</u>	<u>Colors.id</u>	<u>Colors.name</u>
1	1	pen	10	10	red
2	1	pen	10	20	green
3	2	highlighter	20	10	red
4	2	highlighter	20	20	green
5	3	mug	10	10	red
6	3	mug	10	20	green

But ... that's odd, because it creates rows that don't reflect what we know: the 2nd row talks about a green pen, the 3rd talks about a red highlighter and the 6th talks about a green mug, none of which actually exist!

What condition might be use to reduce this join table to just the items that make sense in the database?

When we join we use a condition, just like in SELECT, to reduce the rows to just the combinations that match the objects in the data base. We do this by reducing the cross product with the conditions that the foreign keys (Objects.color_id) equals the primary key that it refers to (Color.id)

```
SELECT *
FROM Objects, Colors
WHERE Objects.color_id = Colors.id
```

(unnamed join table)

(row #)	<u>Objects.id</u>	<u>Objects.name</u>	<u>Objects.color_id</u>	<u>Colors.id</u>	<u>Colors.name</u>
1	1	pen	10	10	red
2	1	pen	10	20	green
3	2	highlighter	20	10	red
4	2	highlighter	20	20	green
5	3	mug	10	10	red
6	3	mug	10	20	green

Results

<u>Objects.id</u>	<u>Objects.name</u>	<u>Objects.color_id</u>	<u>Colors.id</u>	<u>Colors.name</u>
1	pen	10	10	red
2	highlighter	20	20	green
3	mug	10	10	red

Joins of three or more tables work the same, except with more conditions that reduce the resulting table.

You always have 1 less join condition than you have tables.

phpmyadmin doesn't show the table names (just the column names) in the result of a join.