W09 Paper

Left and Right Joins behave in a similar way to Inner Joins, in that they can bring two tables together. However, the sets that are returned are very different. Inner Joins will only connect values that are connected to each other. If the table on the left has a value, but the table on the right has no value that is connected to the same ID, it will get rid of it entirely in the result set. Left Joins will retain all the values on the left side of the join and display NULL for any values that aren’t matched on the right side of the join. Right Joins do the same thing but in reverse.

With three-way joins, you can connect more than just two tables. Once again, if you are relying on left outer joins to connect the tables, then the id of the left table will be displayed along with any values that are found for the corresponding id. If no value is found with the corresponding id, then it will display NULL. If table 3 has a much but table 2 does not, it will display the value for table 3 and display NULL for table 2.

Since Cross Joins create a Cartesian product, you can fabricate result sets depending on what you are trying to display. By using a combination of UNIONS and a cross join, you can iterate through a series of values with one fabricated table and multiply it by another table. It saves time so that you don’t have to perform UNIONs for each connection made.

Natural joins are joins that are performed without specifying the join condition. Therefore, the ON operator isn’t necessary with this type of join. It will look for identical column names and use that as the join condition without you telling it to do so. This can be risky because sometimes two tables can have the same column name without meaning the same thing. It will still perform the join, but maybe not in the way you want it to.