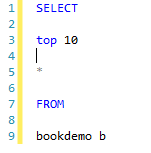
A big part of starting to learn SQL is typing out the queries. Its best to learn how to code efficiently. I don’t mean taking any shortcuts that will be detrimental but actually efficient so you can focus less on typing out long column names and more on what the query is doing. Pinal Dave has written a fantastic post on exactly this topic and [it can be found at (1) his blog.](Dave,%20P.%20(2009,%20April%2006).%20SQL%20SERVER%20-%20Logical%20Query%20Processing%20Phases%20-%20Order%20of%20Statement%20Execution%20-%20Journey%20to%20SQL%20Authority%20with%20Pinal%20Dave.%20Retrieved%20February%2027,%202016,%20from%20http:/blog.sqlauthority.com/2009/04/06/sql-server-logical-query-processing-phases-order-of-statement-execution/)  It’s a great read and I highly recommend it.

I am going to focus on how you use this information. The order from his blog is:

1. FROM  
2. ON  
3. OUTER  
4. WHERE  
5. GROUP BY  
6. CUBE | ROLLUP  
7. HAVING  
8. SELECT  
9. DISTINCT  
10 ORDER BY  
11. TOP

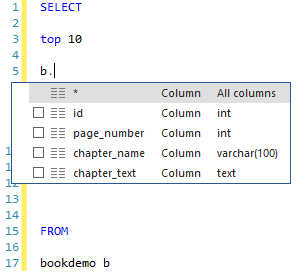
You’re asking why that matters for someone new to SQL. Well when starting to code if you follow these operations you can take advantage of the IntelliSense to auto complete your typing and make things much faster.

I start my queries with the following sequence:



The bookdemo is just a random table that I’ve created for demonstration purposes. The b after the table is now the table alias. It allows me to refer to it using a much shorter name that I’ve assigned it. When aliasing be sure the alias isn’t just random. Please don’t alias a, b, c, d. No one will want to troubleshoot that and when you come back to your code in 6 months you’ll have no idea what you did. The \* is known as a wildcard which just means every column that joins will give you.

Now that I started with a syntactically correct query I can go back and fill in my columns. Using the IntelliSense. That looks like this.



Notice how it knows that b is referencing the bookdemo table and it’s giving me the option to choose the \*(wildcard) or any of the relevant columns without having to type them.

1. Dave, P. (2009, April 06). SQL SERVER - Logical Query Processing Phases - Order of Statement Execution - Journey to SQL Authority with Pinal Dave. Retrieved February 27, 2016, from http://blog.sqlauthority.com/2009/04/06/sql-server-logical-query-processing-phases-order-of-statement-execution/