

- [Skip to navigation](#)
- [Skip to main content](#)
- [Skip to secondary content](#)
- [Skip to comments](#)
- [Skip to footer](#)

(<https://sqlstudies.com>)

# Best practice recommendations for writing Dynamic SQL

(<https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/>)

June 12, 2013 (<https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/>) by Kenneth Fisher

Note this is not “Best Practices when USING Dynamic SQL”. These are just good habits I’ve come up with over the years to make sure that when writing my dynamic SQL it breaks less often. Note I don’t say “doesn’t break ever”, dynamic SQL is tricky stuff. The best you are likely to get is “less often”. Some people may manage 100%, I’m not one of them, but I’ll bet that if they are out there then they use every one of these recommendations.

Generally the problem with dynamic SQL revolves around the fact that, you guessed it, it’s dynamic. It changes based on the data. This means that in order to keep from having problems the statements you create need to be as robust as possible. Every single one of these recommendations are things that you can and probably should use in regular SQL, but most people frequently don’t.

## **Always use []s around object names.**

```
1 | EXEC sp_msforeachdb 'USE [?];'
```

I can hear someone in the back saying “I can ignore this one, all of my names work without the brackets.” Maybe true, maybe not, but I’ll almost guarantee that at some point one of your database names, object names etc will end up starting with a number, have a space or dash in it, a symbol, etc and your dynamic SQL is going to break.

## **Always put EXEC in front of stored procedure calls**

```
1 | SELECT 'EXEC sp_spaceused ['+TABLE_SCHEMA+'.'+TABLE_NAME+'];'  
2 | FROM INFORMATION_SCHEMA.TABLES
```

I wrote about this one time [here](https://sqlstudies.com/2013/02/06/execute-not-required-but-advisable/) (<https://sqlstudies.com/2013/02/06/execute-not-required-but-advisable/>). This one is a good habit anyway, but particularly with dynamic SQL. SQL Server only runs a stored procedure without EXEC if it’s the first thing in the batch. If you are writing dynamic SQL to say generate a group of commands that run a given stored procedure for every table in the database it would be nice if you could run the whole thing at once, not one line at a time.

## **Always include the Schema where appropriate**

```

1 | SELECT 'SELECT COUNT(1) FROM [' + TABLE_SCHEMA + '].[' + TABLE_NAME + ']'
2 | FROM INFORMATION_SCHEMA.TABLES

```

First of all I say “where appropriate” because databases, logins etc don’t have schemas to use. Now in response to that same heckler in the back is now saying “All of my objects are in the dbo schema, I don’t need to include it.” Really? Are you sure? Someone didn’t create an object while you weren’t looking? The vendor software you just bought only uses dbo? You would be amazed how often systems that “only use the dbo schema” don’t really.

### End statements with semicolons;

```

1 | sp_msforeachdb 'USE [?]; PRINT ''?'';'

```

Again something that is a good habit regardless. But since in dynamic SQL you may not have the normal line feeds/carriage returns that let SQL know that this is the end of a statement it’s a good idea include these as a habit. You probably won’t need them, but if you always put them in then on the odd occasion when you do, they will already be there.

### Double, Triple and Quadruple check your quotes

```

1 | SET @stringvar = 'It's'

```

This one is hard enough I wrote an article “[Single Quotation Marks in SQL](http://www.sqlservercentral.com/articles/T-SQL/95670/) (http://www.sqlservercentral.com/articles/T-SQL/95670/)” to try to help people with it. Getting the quotes right is possibly the hardest part of dynamic SQL. Double the quotes when you write dynamic SQL, but then you have dynamic SQL creating dynamic SQL, now it’s quadruple the quotes. Add another layer and it’s 8 times the quotes. It can get ugly really really quickly.

### Format your code so that the output is formatted the way you want to read it

```

1 | DECLARE @SQL nvarchar(max)
2 | DECLARE @ProductID int
3 | DECLARE @MaxQuantity int
4 | DECLARE @MinQuantity int
5 | SET @ProductID = 4
6 | SET @MinQuantity = 1
7 | SET @MaxQuantity = 100
8 |
9 | SET @SQL =
10 | 'SELECT Product.Name, Product.ProductNumber, ' + CHAR(13) +
11 | ' ProductInventory.LocationID, ' + CHAR(13) +
12 | ' ProductInventory.Quantity ' + CHAR(13) +
13 | 'FROM [Production].[Product] ' + CHAR(13) +
14 | 'JOIN [Production].[ProductInventory] ' + CHAR(13) +
15 | ' ON Product.ProductID = ProductInventory.ProductID ' + CHAR(13) +
16 | 'WHERE 1=1 ' + CHAR(13) +
17 | ISNULL(' AND Product.ProductID = '+CAST(@ProductID AS varchar) + CHAR(13)
18 | ISNULL(' AND ProductInventory.Quantity >= '+CAST(@MinQuantity AS varchar)
19 | ISNULL(' AND ProductInventory.Quantity <= '+CAST(@MaxQuantity AS varchar)
20 |
21 | PRINT @SQL

```

SQL could care less about formatting, and its dynamic code that no one is going to read, so why format it? Because at some point someone may read it. Maybe to debug a problem, maybe just to try to figure out what you are doing. But somewhere along the line someone is going to want to

read it and they will very much appreciate you having formatted it ahead of time. And to be fair I'm a fanatic about formatting my code. Everyone has to have their quirks right?

Well, that's it. Those are my suggestions. Stick to them and hopefully you will run into fewer issues with your Dynamic SQL. I should note that I play with Dynamic SQL because I find it interesting and a bit of a challenge at times. As a general case dynamic SQL should be one of your last resorts. Unless you have a good reason to use dynamic SQL or don't have another method that will work, then you should probably skip it and use another method.

NOTE: Some of you may notice I didn't include using `sp_executesql` as one of my recommendations even though it's a good idea and a very important way to avoid SQL injection. I will be talking about it later, I just didn't feel it fit in with the rest of these ideas.

Category: [Dynamic SQL](#), [Microsoft SQL Server](#), [SQLServerPedia Syndication](#), [T-SQL](#) | Tags: [best practices](#), [code language](#), [dynamic sql](#), [language sql](#), [microsoft sql server](#), [sql statements](#), [T-SQL](#)

ORDER BY the numbers (<https://sqlstudies.com/2013/06/17/order-by-the-numbers/>). »  
« [Using the OVER clause \(https://sqlstudies.com/2013/06/10/using-the-over-clause/\)](https://sqlstudies.com/2013/06/10/using-the-over-clause/).

## 16 thoughts on “Best practice recommendations for writing Dynamic SQL”

*Filipe says:*

[June 13, 2013 at 7:42 AM \(https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/#comment-614\)](https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/#comment-614).

I find it interesting that you use only to format you dynamic SQL. I have had problems with that practice, when I want to display the code and then copy and paste to execute, so I got into the practice of always using the CR+LF combination and never had problems after that.

[Reply \(https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=614#respond\)](https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=614#respond).

*Kenneth Fisher (https://sqlstudiesdotcom.wordpress.com) says:*

[June 13, 2013 at 11:01 AM \(https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/#comment-619\)](https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/#comment-619).

I used to use both CR & LF but found over time that CHR(13) was enough when I was cutting and pasting into a query window. Also when I used both I occasionally would get double line breaks if I was putting it into other editors.

[Reply \(https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=619#respond\)](https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=619#respond).

*TSQL hack, decent pilot says:*

[June 13, 2013 at 8:12 AM \(https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/#comment-615\)](https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/#comment-615).

All excellent points for writing dynamic SQL. I would just add that for production support of dynamic SQL nothing beats logging the actual statements being run to a file or table. The only thing that's worse than debugging dynamic sql errors when you aren't seeing what it's actually running is debugging bitwise coding. Great stuff!

[Reply \(https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=615#respond\)](https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=615#respond)

*Kenneth Fisher (https://sqlstudiesdotcom.wordpress.com)* says:

June 13, 2013 at 11:02 AM (<https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/#comment-620>)

I think that is a great suggestion. I wouldn't want to do it all the time because it would add a bit of a delay to the process. Also if I'm running the code frequently enough, 100's of times a minute for example, then I might run into some concurrency issues. Usually if the error is recent enough I can pull what I need out of the DMVs.

[Reply \(https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=620#respond\)](https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=620#respond)

*Charles Kincaid* says:

June 13, 2013 at 8:24 AM (<https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/#comment-616>)

Wonderful article.

I have taken to putting brackets around all identifiers. I upper case SQL reserved words that are not used as identifiers (we have tables [Order] and [User]).

The SELECT is on a line by itself and each column is on its own line

FROM is in the left column. Each join is on a line by itself and indented a couple of spaces.

Same thing for WHERE, GROUP, HAVING and ORDER.

These practices make changes easier.

[Reply \(https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=616#respond\)](https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=616#respond)

*Kenneth Fisher (https://sqlstudiesdotcom.wordpress.com)* says:

June 13, 2013 at 11:04 AM (<https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/#comment-621>)

Thanks :). I'm a big proponent of formatting although when I push for it I try to tell people be reasonably consistent and be readable. Beyond that I don't really care what formatting they use, as long as they use something.

[Reply \(https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=621#respond\)](https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=621#respond)

*beeporama23Brian J. Parker (http://gravatar.com/beeporama23)* says:

June 13, 2013 at 9:17 AM (<https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/#comment-617>)

These are great recommendations, thank you for sharing!

Although you will probably never need it, for the most robust code, consider using the QUOTENAME() function instead of hard coding brackets. For example:

— Instead of:

```
SELECT 'EXEC sp_spaceused ['+TABLE_SCHEMA+'.'+TABLE_NAME+'];'
FROM INFORMATION_SCHEMA.TABLES
```

— Consider:

```
SELECT 'EXEC sp_spaceused '+QUOTENAME(TABLE_SCHEMA+'.'+TABLE_NAME)+';'
FROM INFORMATION_SCHEMA.TABLES
```

This will handle the very rare cases when a table name has a square bracket in it.

When I write a stored procedure that uses complex dynamic SQL, I usually include a “@debug” bit parameter that, when set to 1, prints dynamic SQL before executing it. This has made debugging easier for me. For example:

```
SET @sql = '...'
```

```
IF @debug = 1 print @sql  
EXEC (@sql)
```

[Reply \(https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=617#respond\)](https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=617#respond)

*Kenneth Fisher (https://sqlstudiesdotcom.wordpress.com)* says:

June 13, 2013 at 3:01 PM (<https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/#comment-624>)

I really like the QUOTENAME suggestion. I haven't run into cases with []s in the name but I could see how I could. I don't usually add in a @debug variable just because it seems to add to the complexity. I keep a commented out print statement and if I need to debug I uncomment it and comment out the EXEC as needed. But again, that's just personal preference 😊

[Reply \(https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=624#respond\)](https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=624#respond)

*George* says:

June 13, 2013 at 9:30 AM (<https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/#comment-618>)

```
DECLARE @SQL nvarchar(max) = "  
DECLARE @CrLf char(1) = CHAR(13)
```

```
SET @SQL = @SQL + 'SELECT Product.Name, Product.ProductNumber, ' + @CrLf  
SET @SQL = @SQL + 'ProductInventory.LocationID, ' + @CrLf
```

[Reply \(https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=618#respond\)](https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=618#respond)

*Kenneth Fisher (https://sqlstudiesdotcom.wordpress.com)* says:

June 13, 2013 at 11:06 AM (<https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/#comment-622>)

I've done that before, however, I decided over time that it just added to the complexity. Of course that's just me and everyone has to decide for themselves. 😊

[Reply \(https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=622#respond\)](https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=622#respond)

*Dan Lavigueur* says:

June 13, 2013 at 7:15 PM (<https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/#comment-625>)

I have written some pretty intense dynamic T-SQL. It doesn't break often.

I use all of the practices that you recommend, especially the formatting. It is a great aid in debugging since you can just capture the dynamically generated SQL to debug it; and it actually makes sense.

I also wrap everything with sp\_executesql. It's just safer.

[Reply \(https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=625#respond\)](https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=625#respond)

[Writing Dynamic SQL \(A how to\) | SQL Studies \(https://sqlstudies.com/2013/07/01/writing-dynamic-sql-a-how-to/\)](https://sqlstudies.com/2013/07/01/writing-dynamic-sql-a-how-to/) says:

July 1, 2013 at 11:34 AM (<https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/#comment-711>)

[...] little while back I wrote Best practice recommendations for writing Dynamic SQL where I made a number of suggestions for good habits when writing dynamic SQL. Along the same [...]

[Reply \(https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=711#respond\)](https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=711#respond)

*Rajeshkumar Ramamoorthy* says:

November 11, 2016 at 9:20 PM (<https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/#comment-22105>)

How can i generate procedure script with ALTER but not using any replace key word ???

[Reply \(https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=22105#respond\)](https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=22105#respond)

[Queries with optional parameters | SQL Studies \(https://sqlstudies.com/2017/02/27/queries-with-optional-parameters/\)](https://sqlstudies.com/2017/02/27/queries-with-optional-parameters/) says:

February 27, 2017 at 8:00 AM (<https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/#comment-22937>)

[...] Writing Dynamic SQL (A how to) Best practice recommendations for writing Dynamic SQL [...]

[Reply \(https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=22937#respond\)](https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=22937#respond)

[Dynamic SQL and the joys of QUOTENAME | SQL Studies](https://sqlstudies.com/2017/05/11/dynamic-sql-and-the-joys-of-quotename/)

(<https://sqlstudies.com/2017/05/11/dynamic-sql-and-the-joys-of-quotename/>) says:

May 11, 2017 at 8:01 AM (<https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/#comment-23779>)

[...] a big fan of dynamic SQL. In the past I've written a How to, a Best Practices and even a Generic Dynamic SP although that last one was a bit so so in my opinion. Depending on [...]

[Reply \(https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=23779#respond\)](https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=23779#respond)

[Dynamic SQL and the joys of QUOTENAME – SQL Server Best Practices](https://sqlbp.com/sqlserver/dynamicsqlandthejoysofquotename/)

(<https://sqlbp.com/sqlserver/dynamicsqlandthejoysofquotename/>) says:

August 9, 2017 at 9:36 PM (<https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/#comment-25698>)

[...] Best Practices I'm a big fan of dynamic SQL. In the past I've written a How to, a Best Practices and even a Generic Dynamic SP although that last one was a bit so so in my opinion. Depending on [...]

[Reply \(https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=25698#respond\)](https://sqlstudies.com/2013/06/12/best-practice-recommendations-for-writing-dynamic-sql/?replytocom=25698#respond)

This site uses Akismet to reduce spam. [Learn how your comment data is processed \(https://akismet.com/privacy/\)](https://akismet.com/privacy/).

## Follow me via RSS

(<https://sqlstudies.com/feed/>) [RSS - Posts](https://sqlstudies.com/feed/) (<https://sqlstudies.com/feed/>)

(<https://sqlstudies.com/comments/feed/>) [RSS - Comments](https://sqlstudies.com/comments/feed/) (<https://sqlstudies.com/comments/feed/>)

## Follow me via Email

## Follow me on Twitter

[@sqlstudent144](https://twitter.com/sqlstudent144) (<https://twitter.com/sqlstudent144>)

([https://sqlstudies.com/mvp\\_logo\\_horizontal\\_preferred\\_cyan300\\_rgb\\_300ppi/](https://sqlstudies.com/mvp_logo_horizontal_preferred_cyan300_rgb_300ppi/))

## Archives

- [December 2019](https://sqlstudies.com/2019/12/) (<https://sqlstudies.com/2019/12/>)
- [November 2019](https://sqlstudies.com/2019/11/) (<https://sqlstudies.com/2019/11/>)
- [October 2019](https://sqlstudies.com/2019/10/) (<https://sqlstudies.com/2019/10/>)
- [September 2019](https://sqlstudies.com/2019/09/) (<https://sqlstudies.com/2019/09/>)
- [August 2019](https://sqlstudies.com/2019/08/) (<https://sqlstudies.com/2019/08/>)
- [July 2019](https://sqlstudies.com/2019/07/) (<https://sqlstudies.com/2019/07/>)
- [June 2019](https://sqlstudies.com/2019/06/) (<https://sqlstudies.com/2019/06/>)
- [May 2019](https://sqlstudies.com/2019/05/) (<https://sqlstudies.com/2019/05/>)
- [April 2019](https://sqlstudies.com/2019/04/) (<https://sqlstudies.com/2019/04/>)
- [March 2019](https://sqlstudies.com/2019/03/) (<https://sqlstudies.com/2019/03/>)
- [February 2019](https://sqlstudies.com/2019/02/) (<https://sqlstudies.com/2019/02/>)
- [January 2019](https://sqlstudies.com/2019/01/) (<https://sqlstudies.com/2019/01/>)
- [December 2018](https://sqlstudies.com/2018/12/) (<https://sqlstudies.com/2018/12/>)
- [November 2018](https://sqlstudies.com/2018/11/) (<https://sqlstudies.com/2018/11/>)
- [October 2018](https://sqlstudies.com/2018/10/) (<https://sqlstudies.com/2018/10/>)
- [September 2018](https://sqlstudies.com/2018/09/) (<https://sqlstudies.com/2018/09/>)
- [August 2018](https://sqlstudies.com/2018/08/) (<https://sqlstudies.com/2018/08/>)
- [July 2018](https://sqlstudies.com/2018/07/) (<https://sqlstudies.com/2018/07/>)
- [June 2018](https://sqlstudies.com/2018/06/) (<https://sqlstudies.com/2018/06/>)
- [May 2018](https://sqlstudies.com/2018/05/) (<https://sqlstudies.com/2018/05/>)
- [April 2018](https://sqlstudies.com/2018/04/) (<https://sqlstudies.com/2018/04/>)
- [March 2018](https://sqlstudies.com/2018/03/) (<https://sqlstudies.com/2018/03/>)
- [February 2018](https://sqlstudies.com/2018/02/) (<https://sqlstudies.com/2018/02/>)
- [January 2018](https://sqlstudies.com/2018/01/) (<https://sqlstudies.com/2018/01/>)
- [December 2017](https://sqlstudies.com/2017/12/) (<https://sqlstudies.com/2017/12/>)

- [November 2017 \(https://sqlstudies.com/2017/11/\)](https://sqlstudies.com/2017/11/)
- [October 2017 \(https://sqlstudies.com/2017/10/\)](https://sqlstudies.com/2017/10/)
- [September 2017 \(https://sqlstudies.com/2017/09/\)](https://sqlstudies.com/2017/09/)
- [August 2017 \(https://sqlstudies.com/2017/08/\)](https://sqlstudies.com/2017/08/)
- [July 2017 \(https://sqlstudies.com/2017/07/\)](https://sqlstudies.com/2017/07/)
- [June 2017 \(https://sqlstudies.com/2017/06/\)](https://sqlstudies.com/2017/06/)
- [May 2017 \(https://sqlstudies.com/2017/05/\)](https://sqlstudies.com/2017/05/)
- [April 2017 \(https://sqlstudies.com/2017/04/\)](https://sqlstudies.com/2017/04/)
- [March 2017 \(https://sqlstudies.com/2017/03/\)](https://sqlstudies.com/2017/03/)
- [February 2017 \(https://sqlstudies.com/2017/02/\)](https://sqlstudies.com/2017/02/)
- [January 2017 \(https://sqlstudies.com/2017/01/\)](https://sqlstudies.com/2017/01/)
- [December 2016 \(https://sqlstudies.com/2016/12/\)](https://sqlstudies.com/2016/12/)
- [November 2016 \(https://sqlstudies.com/2016/11/\)](https://sqlstudies.com/2016/11/)
- [October 2016 \(https://sqlstudies.com/2016/10/\)](https://sqlstudies.com/2016/10/)
- [September 2016 \(https://sqlstudies.com/2016/09/\)](https://sqlstudies.com/2016/09/)
- [August 2016 \(https://sqlstudies.com/2016/08/\)](https://sqlstudies.com/2016/08/)
- [July 2016 \(https://sqlstudies.com/2016/07/\)](https://sqlstudies.com/2016/07/)
- [June 2016 \(https://sqlstudies.com/2016/06/\)](https://sqlstudies.com/2016/06/)
- [May 2016 \(https://sqlstudies.com/2016/05/\)](https://sqlstudies.com/2016/05/)
- [April 2016 \(https://sqlstudies.com/2016/04/\)](https://sqlstudies.com/2016/04/)
- [March 2016 \(https://sqlstudies.com/2016/03/\)](https://sqlstudies.com/2016/03/)
- [February 2016 \(https://sqlstudies.com/2016/02/\)](https://sqlstudies.com/2016/02/)
- [January 2016 \(https://sqlstudies.com/2016/01/\)](https://sqlstudies.com/2016/01/)
- [December 2015 \(https://sqlstudies.com/2015/12/\)](https://sqlstudies.com/2015/12/)
- [November 2015 \(https://sqlstudies.com/2015/11/\)](https://sqlstudies.com/2015/11/)
- [October 2015 \(https://sqlstudies.com/2015/10/\)](https://sqlstudies.com/2015/10/)
- [September 2015 \(https://sqlstudies.com/2015/09/\)](https://sqlstudies.com/2015/09/)
- [August 2015 \(https://sqlstudies.com/2015/08/\)](https://sqlstudies.com/2015/08/)
- [July 2015 \(https://sqlstudies.com/2015/07/\)](https://sqlstudies.com/2015/07/)
- [June 2015 \(https://sqlstudies.com/2015/06/\)](https://sqlstudies.com/2015/06/)
- [May 2015 \(https://sqlstudies.com/2015/05/\)](https://sqlstudies.com/2015/05/)
- [April 2015 \(https://sqlstudies.com/2015/04/\)](https://sqlstudies.com/2015/04/)
- [March 2015 \(https://sqlstudies.com/2015/03/\)](https://sqlstudies.com/2015/03/)
- [February 2015 \(https://sqlstudies.com/2015/02/\)](https://sqlstudies.com/2015/02/)
- [January 2015 \(https://sqlstudies.com/2015/01/\)](https://sqlstudies.com/2015/01/)
- [December 2014 \(https://sqlstudies.com/2014/12/\)](https://sqlstudies.com/2014/12/)
- [November 2014 \(https://sqlstudies.com/2014/11/\)](https://sqlstudies.com/2014/11/)
- [October 2014 \(https://sqlstudies.com/2014/10/\)](https://sqlstudies.com/2014/10/)
- [September 2014 \(https://sqlstudies.com/2014/09/\)](https://sqlstudies.com/2014/09/)
- [August 2014 \(https://sqlstudies.com/2014/08/\)](https://sqlstudies.com/2014/08/)
- [July 2014 \(https://sqlstudies.com/2014/07/\)](https://sqlstudies.com/2014/07/)
- [June 2014 \(https://sqlstudies.com/2014/06/\)](https://sqlstudies.com/2014/06/)
- [May 2014 \(https://sqlstudies.com/2014/05/\)](https://sqlstudies.com/2014/05/)
- [April 2014 \(https://sqlstudies.com/2014/04/\)](https://sqlstudies.com/2014/04/)
- [March 2014 \(https://sqlstudies.com/2014/03/\)](https://sqlstudies.com/2014/03/)
- [February 2014 \(https://sqlstudies.com/2014/02/\)](https://sqlstudies.com/2014/02/)
- [January 2014 \(https://sqlstudies.com/2014/01/\)](https://sqlstudies.com/2014/01/)
- [December 2013 \(https://sqlstudies.com/2013/12/\)](https://sqlstudies.com/2013/12/)
- [November 2013 \(https://sqlstudies.com/2013/11/\)](https://sqlstudies.com/2013/11/)
- [October 2013 \(https://sqlstudies.com/2013/10/\)](https://sqlstudies.com/2013/10/)
- [September 2013 \(https://sqlstudies.com/2013/09/\)](https://sqlstudies.com/2013/09/)



- [August 2013 \(https://sqlstudies.com/2013/08/\)](https://sqlstudies.com/2013/08/)
- [July 2013 \(https://sqlstudies.com/2013/07/\)](https://sqlstudies.com/2013/07/)
- [June 2013 \(https://sqlstudies.com/2013/06/\)](https://sqlstudies.com/2013/06/)
- [May 2013 \(https://sqlstudies.com/2013/05/\)](https://sqlstudies.com/2013/05/)
- [April 2013 \(https://sqlstudies.com/2013/04/\)](https://sqlstudies.com/2013/04/)
- [March 2013 \(https://sqlstudies.com/2013/03/\)](https://sqlstudies.com/2013/03/)
- [February 2013 \(https://sqlstudies.com/2013/02/\)](https://sqlstudies.com/2013/02/)
- [January 2013 \(https://sqlstudies.com/2013/01/\)](https://sqlstudies.com/2013/01/)
- [December 2012 \(https://sqlstudies.com/2012/12/\)](https://sqlstudies.com/2012/12/)
- [November 2012 \(https://sqlstudies.com/2012/11/\)](https://sqlstudies.com/2012/11/)
- [October 2012 \(https://sqlstudies.com/2012/10/\)](https://sqlstudies.com/2012/10/)
- [September 2012 \(https://sqlstudies.com/2012/09/\)](https://sqlstudies.com/2012/09/)
- [August 2012 \(https://sqlstudies.com/2012/08/\)](https://sqlstudies.com/2012/08/)

## Categories

- [Auditing \(https://sqlstudies.com/category/microsoft-sql-server/auditing/\)](https://sqlstudies.com/category/microsoft-sql-server/auditing/)
- [Backups \(https://sqlstudies.com/category/microsoft-sql-server/backups/\)](https://sqlstudies.com/category/microsoft-sql-server/backups/)
- [Bad idea Cowboy hat \(https://sqlstudies.com/category/microsoft-sql-server/bad-idea-cowboy-hat/\)](https://sqlstudies.com/category/microsoft-sql-server/bad-idea-cowboy-hat/)
- [Batch Files \(https://sqlstudies.com/category/batch-files/\)](https://sqlstudies.com/category/batch-files/)
- [Blogging \(https://sqlstudies.com/category/blogging-2/\)](https://sqlstudies.com/category/blogging-2/)
- [Book Review \(https://sqlstudies.com/category/microsoft-sql-server/book-review/\)](https://sqlstudies.com/category/microsoft-sql-server/book-review/)
- [Central Management Server \(https://sqlstudies.com/category/microsoft-sql-server/central-management-server/\)](https://sqlstudies.com/category/microsoft-sql-server/central-management-server/)
- [Certification \(https://sqlstudies.com/category/microsoft-sql-server/certification/\)](https://sqlstudies.com/category/microsoft-sql-server/certification/)
- [Cloud \(https://sqlstudies.com/category/cloud/\)](https://sqlstudies.com/category/cloud/)
- [Collation \(https://sqlstudies.com/category/microsoft-sql-server/collation/\)](https://sqlstudies.com/category/microsoft-sql-server/collation/)
- [Connection Strings \(https://sqlstudies.com/category/connection-strings/\)](https://sqlstudies.com/category/connection-strings/)
- [Database Mail \(https://sqlstudies.com/category/microsoft-sql-server/database-mail-microsoft-sql-server/\)](https://sqlstudies.com/category/microsoft-sql-server/database-mail-microsoft-sql-server/)
- [DBA Humor \(https://sqlstudies.com/category/dba-humor/\)](https://sqlstudies.com/category/dba-humor/)
- [DBCC \(https://sqlstudies.com/category/microsoft-sql-server/dbcc/\)](https://sqlstudies.com/category/microsoft-sql-server/dbcc/)
- [Distributed Transaction Coordinator \(https://sqlstudies.com/category/microsoft-sql-server/distributed-transaction-coordinator/\)](https://sqlstudies.com/category/microsoft-sql-server/distributed-transaction-coordinator/)
- [DMV \(https://sqlstudies.com/category/microsoft-sql-server/dmv/\)](https://sqlstudies.com/category/microsoft-sql-server/dmv/)
- [Documentation \(https://sqlstudies.com/category/documentation/\)](https://sqlstudies.com/category/documentation/)
- [DTS \(https://sqlstudies.com/category/microsoft-sql-server/dts/\)](https://sqlstudies.com/category/microsoft-sql-server/dts/)
- [Dynamic SQL \(https://sqlstudies.com/category/microsoft-sql-server/dynamic-sql/\)](https://sqlstudies.com/category/microsoft-sql-server/dynamic-sql/)
- [Encryption \(https://sqlstudies.com/category/microsoft-sql-server/encryption-microsoft-sql-server/\)](https://sqlstudies.com/category/microsoft-sql-server/encryption-microsoft-sql-server/)
- [Extended Events \(https://sqlstudies.com/category/microsoft-sql-server/extended-events/\)](https://sqlstudies.com/category/microsoft-sql-server/extended-events/)
- [Impersonation \(https://sqlstudies.com/category/microsoft-sql-server/impersonation/\)](https://sqlstudies.com/category/microsoft-sql-server/impersonation/)
- [Index \(https://sqlstudies.com/category/microsoft-sql-server/index/\)](https://sqlstudies.com/category/microsoft-sql-server/index/)
- [Interview \(https://sqlstudies.com/category/interview/\)](https://sqlstudies.com/category/interview/)
- [Keywords \(https://sqlstudies.com/category/microsoft-sql-server/keywords/\)](https://sqlstudies.com/category/microsoft-sql-server/keywords/)
- [Log File \(https://sqlstudies.com/category/microsoft-sql-server/log-file/\)](https://sqlstudies.com/category/microsoft-sql-server/log-file/)
- [Microsoft SQL Server \(https://sqlstudies.com/category/microsoft-sql-server/\)](https://sqlstudies.com/category/microsoft-sql-server/)

- [Partitioning](https://sqlstudies.com/category/microsoft-sql-server/partitioning/) (<https://sqlstudies.com/category/microsoft-sql-server/partitioning/>).
- [Pass](https://sqlstudies.com/category/microsoft-sql-server/pass/) (<https://sqlstudies.com/category/microsoft-sql-server/pass/>).
- [Performance](https://sqlstudies.com/category/microsoft-sql-server/performance/) (<https://sqlstudies.com/category/microsoft-sql-server/performance/>).
- [Policy Based Management](https://sqlstudies.com/category/microsoft-sql-server/policy-based-management/) (<https://sqlstudies.com/category/microsoft-sql-server/policy-based-management/>).
- [Powershell](https://sqlstudies.com/category/powershell/) (<https://sqlstudies.com/category/powershell/>).
- [Problem Resolution](https://sqlstudies.com/category/microsoft-sql-server/problem-resolution/) (<https://sqlstudies.com/category/microsoft-sql-server/problem-resolution/>).
- [Query Plans](https://sqlstudies.com/category/microsoft-sql-server/query-plans/) (<https://sqlstudies.com/category/microsoft-sql-server/query-plans/>).
- [Referential Integrity](https://sqlstudies.com/category/microsoft-sql-server/referential-integrity/) (<https://sqlstudies.com/category/microsoft-sql-server/referential-integrity/>).
- [Reporting](https://sqlstudies.com/category/microsoft-sql-server/reporting/) (<https://sqlstudies.com/category/microsoft-sql-server/reporting/>).
- [Security](https://sqlstudies.com/category/microsoft-sql-server/security-microsoft-sql-server/) (<https://sqlstudies.com/category/microsoft-sql-server/security-microsoft-sql-server/>).
- [Settings](https://sqlstudies.com/category/microsoft-sql-server/settings/) (<https://sqlstudies.com/category/microsoft-sql-server/settings/>).
- [Speaking](https://sqlstudies.com/category/speaking/) (<https://sqlstudies.com/category/speaking/>).
- [SQL Agent Jobs](https://sqlstudies.com/category/microsoft-sql-server/sql-agent-jobs/) (<https://sqlstudies.com/category/microsoft-sql-server/sql-agent-jobs/>).
- [SQL Database](https://sqlstudies.com/category/cloud/sql-database/) (<https://sqlstudies.com/category/cloud/sql-database/>).
- [SQL Hangout](https://sqlstudies.com/category/microsoft-sql-server/sql-hangout/) (<https://sqlstudies.com/category/microsoft-sql-server/sql-hangout/>).
- [SQL Homework](https://sqlstudies.com/category/microsoft-sql-server/sql-homework/) (<https://sqlstudies.com/category/microsoft-sql-server/sql-homework/>).
- [SQL Judo's Monthly DBA Challenge](https://sqlstudies.com/category/microsoft-sql-server/sql-judos-monthly-dba-challenge/) (<https://sqlstudies.com/category/microsoft-sql-server/sql-judos-monthly-dba-challenge/>).
- [SQL Services](https://sqlstudies.com/category/microsoft-sql-server/sql-services/) (<https://sqlstudies.com/category/microsoft-sql-server/sql-services/>).
- [SQLServerPedia Syndication](https://sqlstudies.com/category/sqlserverpedia-syndication/) (<https://sqlstudies.com/category/sqlserverpedia-syndication/>).
- [SSIS](https://sqlstudies.com/category/microsoft-sql-server/ssis/) (<https://sqlstudies.com/category/microsoft-sql-server/ssis/>).
- [SSMS](https://sqlstudies.com/category/microsoft-sql-server/ssms/) (<https://sqlstudies.com/category/microsoft-sql-server/ssms/>).
- [System Databases](https://sqlstudies.com/category/microsoft-sql-server/system-databases/) (<https://sqlstudies.com/category/microsoft-sql-server/system-databases/>).
- [System Functions and Stored Procedures](https://sqlstudies.com/category/microsoft-sql-server/system-functions-and-stored-procedures/) (<https://sqlstudies.com/category/microsoft-sql-server/system-functions-and-stored-procedures/>).
- [T-SQL](https://sqlstudies.com/category/microsoft-sql-server/t-sql/) (<https://sqlstudies.com/category/microsoft-sql-server/t-sql/>).
- [T-SQL Tuesday](https://sqlstudies.com/category/microsoft-sql-server/t-sql-tuesday/) (<https://sqlstudies.com/category/microsoft-sql-server/t-sql-tuesday/>).
- [Transactions](https://sqlstudies.com/category/microsoft-sql-server/transactions/) (<https://sqlstudies.com/category/microsoft-sql-server/transactions/>).
- [Uncategorized](https://sqlstudies.com/category/uncategorized/) (<https://sqlstudies.com/category/uncategorized/>).
- [Upgrades](https://sqlstudies.com/category/microsoft-sql-server/upgrades/) (<https://sqlstudies.com/category/microsoft-sql-server/upgrades/>).
- [VLF](https://sqlstudies.com/category/microsoft-sql-server/vlf/) (<https://sqlstudies.com/category/microsoft-sql-server/vlf/>).

## Meta

- [Register](#)
- [Log in](#)
- [Entries feed](#)
- [Comments feed](#)
- [WordPress.com](#)

(<http://www.toadworld.com/platforms/sql-server/default.aspx>).

[Blog at WordPress.com.](https://wordpress.com/?ref=footer_blog) ([https://wordpress.com/?ref=footer\\_blog](https://wordpress.com/?ref=footer_blog)).