**Run same command on all SQL Server databases without cursors**

|  |  |
| --- | --- |
| http://www.mssqltips.com/images/TimFord.jpg | By: [Tim Ford](http://www.mssqltips.com/sqlserverauthor/13/tim-ford/)   |   [Read Comments (26)](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/#comments)   |   Related Tips: [1](http://www.mssqltips.com/sqlservertip/1599/sql-server-cursor-example/) | 2 | [3](http://www.mssqltips.com/sqlservertip/1905/iterate-through-sql-server-database-objects-without-cursors/) | [More](http://www.mssqltips.com/sql_server_developer_tips.asp) > [Scripts](http://www.mssqltips.com/sql-server-tip-category/26/scripts/) |

**Problem**There are times when I find myself needing to run a SQL command against each database on one of my SQL Server instances. There is a handy undocumented stored procedure that allows you to do this without needing to set up a cursor against your sysdatabases table in the master database: sp\_MSforeachdb.

**Solution**The syntax for this undocumented procedure is:

**EXEC sp\_MSforeachdb @command**

(Where @command is a variable-length string.)

**Example 1: Query Information From All Databases On A SQL Instance**

|  |
| --- |
| --This query will return a listing of all tables in all databases on a SQL instance:  DECLARE @command varchar(1000)  SELECT @command = 'USE ? SELECT name FROM sysobjects WHERE xtype = ''U'' ORDER BY name'  EXEC sp\_MSforeachdb @command |

You can alternately omit the process of declaring and setting the @command variable. The T-SQL command below behaves identically to the one above and is condensed to a single line of code:

|  |
| --- |
| --This query will return a listing of all tables in all databases on a SQL instance:  EXEC sp\_MSforeachdb 'USE ? SELECT name FROM sysobjects WHERE xtype = ''U'' ORDER BY name' |

**Example 2: Execute A DDL Query Against All User Databases On A SQL Instance**

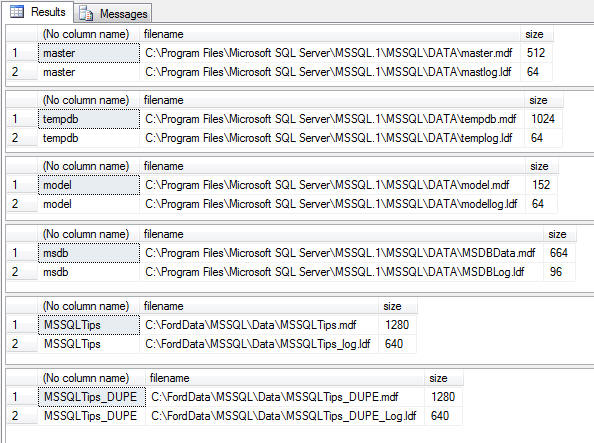
|  |
| --- |
| --This statement creates a stored procedure in each user database that will return a listing of all users in a database, sorted by their modification date  DECLARE @command varchar(1000)  SELECT @command = 'IF ''?'' NOT IN(''master'', ''model'', ''msdb'', ''tempdb'') BEGIN USE ? EXEC(''CREATE PROCEDURE pNewProcedure1 AS SELECT name, createdate, updatedate FROM sys.sysusers ORDER BY updatedate DESC'') END'  EXEC sp\_MSforeachdb @command |

As you may notice, there are additional items to take into consideration when limiting the scope of the sp\_MSforeachdb stored procedure, particularly when creating or modifying objects. You must also set the code to execute if the IF statement is true by using the T-SQL keywords BEGIN and END. You should take note that the USE ? statement is contained within the BEGIN...END block. It is important to remember key T-SQL rules and account for them. In this case the rule that when creating a procedure, the CREATE PROCEDURE phrase must be the first line of code to be executed. To accomplish this you can encapsulate the CREATE PROCEDURE code within an explicit EXEC() function.

**What about the "?" Placeholder**Throughout the examples provided above you'll see the use of the question mark as a placeholder for the database/database name. To reference the database name as a string to be returned in a query, embed it between a double set of single quotation marks. To treat it as a reference to the database object simply use it by itself (as presented in Example 3b.) It is necessary to set the database for the query to run against, by using the USE ? statement, otherwise the code will execute in the context of the current database, for each database in your SQL instance. If you have 5 databases hosted in the current instance and you were to run the stored procedure code above while in the context of DBx it would execute the T-SQL text of the @command 5 times in DBx. This behavior is evident in the output of Example 3 below.

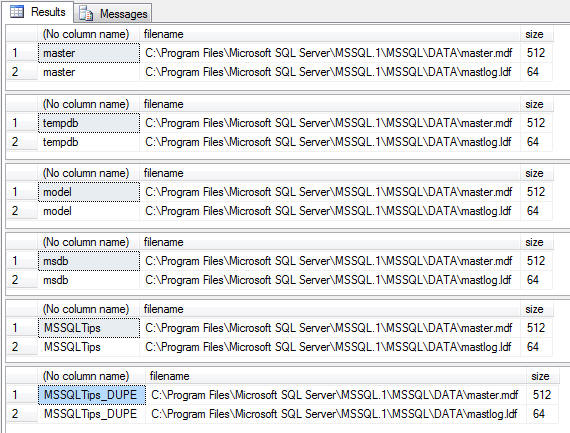
**Example 3: Query File Information From All Databases On A SQL Instance**

|  |
| --- |
| --This query will return a listing of all files in all databases on a SQL instance:  EXEC sp\_MSforeachdb 'USE ? SELECT ''?'', SF.filename, SF.size FROM sys.sysfiles SF' |



What happens though if we omit the USE ? clause, which sets the scope of the query? As you can see below, though it is apparent the code executed for each database, it never changed context. Pay particular interest to the filename column and you will see that the query executed from within the context of the master database (where I ran the query from) for each database in the SQL instance (as noted by the database name being returned via the use of the "?" placeholder).

|  |
| --- |
| --Remove the USE ? clause and you end up executing the query repetitively within the context of the current database:  EXEC sp\_MSforeachdb 'SELECT ''?'', SF.filename, SF.size FROM sys.sysfiles SF' |



**Why Not Just Use a Cursor?**Sure, a cursor can accomplish all that I've presented above, but let's look at the code required to set up a cursor to execute the command used in Example 3:

|  |
| --- |
| DECLARE @DB\_Name varchar(100)  DECLARE @Command nvarchar(200)  DECLARE database\_cursor CURSOR FOR  SELECT name  FROM MASTER.sys.sysdatabases  OPEN database\_cursor  FETCH NEXT FROM database\_cursor INTO @DB\_Name  WHILE @@FETCH\_STATUS = 0  BEGIN       SELECT @Command = 'SELECT ' + '''' + @DB\_Name + '''' + ', SF.filename, SF.size FROM sys.sysfiles SF'      EXEC sp\_executesql @Command       FETCH NEXT FROM database\_cursor INTO @DB\_Name  END  CLOSE database\_cursor  DEALLOCATE database\_cursor |

Considering the behavior is similar I'd rather type and execute a single line of T-SQL code versus sixteen.

**Next Steps**

* sp\_MSforeachdb is extremely useful for pulling together metadata about your various SQL databases. I use it quite frequently for reporting on such important metrics as database file sizes, amount of free space, and backup status.
* In future tips I will present how to collect those metrics and many more, and report on them via SQL Server Reporting Services. Stay Tuned!

Last Update: 1/17/2008

**About the author**

[](http://www.mssqltips.com/sqlserverauthor/13/tim-ford/)

Tim is a Senior Database Administrator and SQL Server MVP. He has been working with SQL Server since 1999.   
  
[**View all my tips**](http://www.mssqltips.com/sqlserverauthor/13/tim-ford/)

**We Recommend**

* [SQL Server Cursor Example...](http://www.mssqltips.com/sqlservertip/1599/sql-server-cursor-example/)
* [Iterate through SQL Server database objects withou...](http://www.mssqltips.com/sqlservertip/1905/iterate-through-sql-server-database-objects-without-cursors/)
* [More Database Developer Tips...](http://www.mssqltips.com/sql_server_developer_tips.asp)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| [[http://www.mssqltips.com/images/print.gif](http://www.mssqltips.com/tipprint.asp?tip=1414)Print](http://www.mssqltips.com/tipprint.asp?tip=1414) | [in**Share**](javascript:void(0);)0 |  | [Tweet](http://twitter.com/share) |  | [Become a paid author](http://www.mssqltips.com/contribute.asp) |

***[Solving SQL Server problems for millions of DBAs and Developers.  
  
Join MSSQLTips.com and get ahead.](http://www.mssqltips.com/freesqlservertips.asp)***

**Comments and Feedback:**

|  |  |
| --- | --- |
| **Thursday, January 17, 2008 - 5:23:02 PM - probaby** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| Thanks for your wonderful scripts!  You save my time and money!  Have a great day! | |

|  |  |
| --- | --- |
| **Thursday, April 03, 2008 - 11:53:47 AM - prasanthg74** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| nice script ! thanks a bunch! | |

|  |  |
| --- | --- |
| **Friday, April 04, 2008 - 7:07:46 AM - admin** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| Thanks so much.  Tim Ford did a great job with this tip.  Thank you, The MSSQLTips.com Team | |

|  |  |
| --- | --- |
| **Wednesday, December 10, 2008 - 3:23:23 PM - robvon** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| You may need some "[" and "]" around the '?' to deal with dodgy database names.......   as in:  EXEC sp\_MSforeachdb 'USE [?] SELECT ''?'', SF.filename, SF.size FROM sys.sysfiles SF'  R | |

|  |  |
| --- | --- |
| **Sunday, December 14, 2008 - 9:04:32 PM - timmer26** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| You are absolutely correct.  This is particularly important when dealing with spaces in database names. | |

|  |  |
| --- | --- |
| **Friday, November 20, 2009 - 3:49:19 AM - bobmihada** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| --// Just another flavor to offer IF OBJECT\_ID('tempdb..##TempDBInfo') IS NOT NULL BEGIN PRINT 'Previous ##TmpDBInfo found and Dropped.' DROP TABLE ##TempDBInfo END CREATE TABLE ##TempDBInfo(DBName nVarChar(255), FileName nvarchar(255), fileSize INT) EXEC sp\_MSforeachdb 'USE [?] INSERT INTO ##TempDBInfo SELECT ''?'', SF.filename, SF.size FROM sys.sysfiles SF' SELECT \* FROM ##TempDBInfo --// DROP TABLE ##TempDBInfo | |

|  |  |
| --- | --- |
| **Wednesday, January 27, 2010 - 11:19:23 AM - Akbar** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| Hi All, Is there are way to grant Execute permission to a particular Stored procedure or Function in all database of SQL Server 2008 Instance? The below one use to Grant Execute permission to any procedure.   EXEC sp\_msforeachdb 'USE ?; GRANT EXECUTE TO UserOrRoleName'  But when trying the below one I'm getting error  EXEC sp\_msforeachdb 'USE ?; GRANT EXECUTE ON [?].[dbo].[sp\_StoredProc1] TO UserOrRoleName'  Error Message: Cannot find the object 'sp\_StoredProc1', because it does not exist or you do not have permission.  I have the 'sp\_StroedProc1' exist in all my database in the sql server 2008 instance. Any help will be highly appreciated. | |

|  |  |
| --- | --- |
| **Friday, January 29, 2010 - 12:02:55 PM - aprato** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| EXEC sp\_msforeachdb 'USE ?; if object\_id(''sp\_StoredProc1'') is not null begin print ''?'' GRANT EXECUTE ON [dbo].[sp\_StoredProc1] TO public end' | |

|  |  |
| --- | --- |
| **Monday, April 26, 2010 - 3:02:40 PM - thomashmaine** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| is there a way to have a variabled declared  withing the statement?    select @command1 = 'use [?]; exec sp\_grantlogin [@user2]'  print @user2  exec sp\_MSforeachdb @command1 | |

|  |  |
| --- | --- |
| **Wednesday, November 03, 2010 - 10:11:28 AM - Pradip Patel** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| You need to replace       SELECT @Command = 'SELECT ' + '''' + @DB\_Name + '''' + ', SF.filename, SF.size FROM sys.sysfiles SF' with       SELECT @Command = 'USE '+ @DB\_NAME +  ' SELECT ' + '''' + @DB\_Name + '''' + ', SF.filename, SF.size FROM sys.sysfiles SF'  when you want to execute by cursor otherwise it gives you all file name as master.mdf | |

|  |  |
| --- | --- |
| **Tuesday, February 01, 2011 - 11:49:50 AM - Larrybud** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| There are times when you might need a cursor:  1) running a script against certain databases, instead of all  2) wanting your script results in one table (for a select, for example) | |

|  |  |
| --- | --- |
| **Thursday, September 08, 2011 - 4:16:37 PM - Jason** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| When you need a special condition of list of DBs such as not offline, not read-only, use temp table or CTE or table variable, for example, the code below  Regards,  Jason  <http://usa.redirectme.net>    -----------------  SET NOCOUNT ON  DECLARE @dbName varchar(255), @rc int  SELECT name INTO #db FROM sys.databases WHERE      name NOT IN ('master', 'model', 'msdb', 'tempdb', 'admindb') AND     name NOT LIKE '%ReportServer%' AND     DATABASEPROPERTYEX([name], 'IsInStandBy') = 0 AND     DATABASEPROPERTYEX([name], 'Status') = 'ONLINE'  SELECT @rc = 1, @dbName = MIN(name) FROM #db  WHILE @rc <> 0 BEGIN      EXEC DB.schema.DO\_SOMETHING      SELECT TOP 1 @dbName = name     FROM #db     WHERE name > @dbName     ORDER BY name      SET @rc = @@ROWCOUNT END  DROP TABLE #db | |

|  |  |
| --- | --- |
| **Tuesday, February 07, 2012 - 8:49:37 AM - Fernando Paez Becker** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| There is anothers parameters to comments... EXEC sp\_msforeachdb sintax accepts  EXEC sp\_msforeachdb @precommand='---YOUR TEXT FOR GENERAL PURPUSE HERE-TEMP TABLE CREATION IE...  CREATE TABLE ##TEMP1 (ID INT,NAME VARCHAR(15);SELECT \* INTO ##TEMP2 FROM ##TEMP1',  @COMMAND1='USE [?];SELECT \* INTO ##TEMP1 WHERE ...',  @COMMAND2='USE [?];SELECT \* INTO ##TEMP2 WHERE ...',   @COMMAND3='USE [?];SELECT \* INTO ##TEMP1 WHERE ...',  @POSTCOMMAND='SELECT \* FROM ##TEMP1 UNION SELECT \* FROM ##TEMP2;DROP TABLE ##TEMP1;DROP TABLE ##TEMP2;' | |

|  |  |
| --- | --- |
| **Tuesday, February 28, 2012 - 4:31:02 AM - akash** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| DECLARE @DB\_Name varchar(100)  DECLARE @Command nvarchar(200)  DECLARE database\_cursor CURSOR FOR  SELECT name FROM master..sysdatabases where name like '%\_Se'  OPEN database\_cursor  FETCH NEXT FROM database\_cursor INTO @DB\_Name  WHILE @@FETCH\_STATUS = 0  BEGIN  SELECT @Command = 'USE ['+ @DB\_NAME +  '] INSERT INTO  genMenuMaster                     (MenuId, ModuleId, MenuName, MenuType, ParentId, MenuURL, [Position]) VALUES     (1163, 32, ''SMS Mobile DND Status Report'', 0, ''General/SMSMobileDNDStatus.aspx'', 60)'     EXEC sp\_executesql @Command       FETCH NEXT FROM database\_cursor INTO @DB\_Name  END  CLOSE database\_cursor  DEALLOCATE database\_cursor | |

|  |  |
| --- | --- |
| **Tuesday, February 28, 2012 - 4:31:43 AM - akash** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| Server: Msg 105, Level 15, State 1, Line 3 Unclosed quotation mark before the character string '\General/S'. | |

|  |  |
| --- | --- |
| **Tuesday, March 13, 2012 - 11:35:27 PM - surajkuamwat** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| Respect sir               i am facing some problem in sql query  using cursor declaration t was try to do with declare variable                    declare      cmain cursorforselect\*from gstock                    open    cmain               how to use cursor in fetch status... | |

|  |  |
| --- | --- |
| **Tuesday, March 13, 2012 - 11:54:24 PM - SURAJ** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| Dear sir        I AM FACING SOME PROBLEM IN SQL QUERY CURSOR DECLARATION I WANT TO USE CURSOR WITHOUT DECLARE VARIABLE      PLEASE GIVE ME SOLUCTION     MENTON MY QUERY      DECLARE @ABCTABLE AS TABLE  (STKCODE INT,STOCKNO VARCHAR(20))      declare abc\_cursor cursor for select  stkcode ,stockno from stock     OPEN abc\_curosr                \*\* THIS QUERY EXECUTE  SUCESSFUL BUT HOW TO USE IN FETCH NEXT FROM CURSOR      FETCH NEXT FROM abc\_curosr  INTO @stkcode,@stockno      \*\* ERROR CURSOR VARIABLE NOT READ           WHILE @@FETCH\_STATUS = 0             BEGIN                INSET INTO  @ABCTABLE (STKCODE,STOCKNO) VALUES (@stkcode,@stockno)              FETCH NEXT FROM abc\_curosr       INTO  @DB\_Name  END  CLOSE abc\_curosr       DEALLOCATE abc\_curosr | |

|  |  |
| --- | --- |
| **Tuesday, May 01, 2012 - 6:27:28 PM - corey\_lawson** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| This, and sp\_msforeachtable, are internally not magic or undocumented, in that they don't invoke any CLR functionality, extended stored procs, obfuscated TSQL, etc.  They're just plain ol' TSQL procs that executes the code specified in the parameters inside a CURSOR that loops over the desired objects... | |

|  |  |
| --- | --- |
| **Wednesday, May 16, 2012 - 6:40:50 AM - Sandeep Maher** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| Hi,  I got your email id in forum. I am having issues in SQL,  Issues is that I am having more than 50 customers and there is more than 50 Database in SQl for every customers. Now All SP's And Functions are same but now problem is in maintenance, If I modify any SP then I need to modify SP manually in all database. So is there any way to avoid such maintenance problem  So I can modify one SP and that will get update in all remaining Database?  Or Can I get script in c# code using the can i modify SP in all database? **EXEC sp\_MSforeachdb can not work for me because some of my SP text length is more than 10,000.**  **Kindly help me.. Thanks in Advance.** | |

|  |  |
| --- | --- |
| **Monday, June 18, 2012 - 5:32:37 PM - Ludwig Guevara** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| Hi,  I use this script to fix all Orphan users in a Server  (Use a cursor, because requires to check for all the users in the database)  **exec sp\_msforeachdb '**         **USE [?];**                  DECLARE              @DB sysname                      **SELECT DB\_NAME()**                  BEGIN             BEGIN                 DECLARE                      @username varchar(25)                                  DECLARE fixusers CURSOR FOR                     SELECT UserName = name                      FROM sysusers                     WHERE issqluser = 1                          AND (sid is not null and sid <> 0x0)                         AND name <>''dallas''                         AND suser\_sname(sid) is null                     ORDER BY name                                          OPEN fixusers                         FETCH NEXT FROM fixusers                              INTO @username                                                      WHILE @@FETCH\_STATUS = 0                             BEGIN                                 EXEC sp\_change\_users\_login ''update\_one'', @username, @username                                 FETCH NEXT FROM fixusers                                     INTO @username                             END                         CLOSE fixusers                 DEALLOCATE fixusers             END                                  END'  Thanks for the tip  Regards | |

|  |  |
| --- | --- |
| **Monday, June 18, 2012 - 5:34:53 PM - Ludwig Guevara** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| Hi,  I also use a simple script (Callit from a Job) to check for the status of all the databases.  Runs after a Failover or a Server Reboot.    CREATE PROC [dbo].[usp\_CheckDB] AS BEGIN     BEGIN TRY         BEGIN TRANSACTION             exec sp\_msforeachdb '                 use [?];                 exec sp\_helpdb ?';              END TRY     ---     BEGIN CATCH         DECLARE @ErrorMessage NVARCHAR(4000);         DECLARE @ErrorSeverity INT;         DECLARE @ErrorState INT;          SELECT              @ErrorMessage = ERROR\_MESSAGE(),             @ErrorSeverity = ERROR\_SEVERITY(),             @ErrorState = ERROR\_STATE();          -- Use RAISERROR inside the CATCH block to return error         -- information about the original error that caused         -- execution to jump to the CATCH block.         RAISERROR (@ErrorMessage, @ErrorSeverity, @ErrorState );     END CATCH     COMMIT END | |

|  |  |
| --- | --- |
| **Tuesday, June 19, 2012 - 1:55:45 PM - Kevin Archibald** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| Nice!  I use this to shrink db and truncate logs of EVERY DB.    DECLARE @cmd1 nvarchar(2000)  SET @cmd1 = 'IF ''?'' NOT IN(''master'', ''model'', ''tempdb'', ''msdb'')' + 'BEGIN '  + 'USE ?  DBCC SHRINKFILE( ?\_log, 1 )  BACKUP LOG ? WITH TRUNCATE\_ONLY  DBCC SHRINKFILE( ?\_log, 1 )  END  '    EXEC sp\_MSForEachdb  @command1 = @cmd1  GO | |

|  |  |
| --- | --- |
| **Friday, August 03, 2012 - 7:14:31 AM - Amol Birar** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| Very useful indeed | |

|  |  |
| --- | --- |
| **Tuesday, December 04, 2012 - 6:28:32 AM - Jordan** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| Thanks a lot man ! You save my day.  Regards,  Jordan | |

|  |  |
| --- | --- |
| **Wednesday, December 26, 2012 - 3:13:45 PM - Christina** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| I need to loop the following script through all databases.  I have tried almost everything I have researched online without any success.  Any suggestions?    DECLARE @table\_name VARCHAR(500)    DECLARE @schema\_name VARCHAR(500)  DECLARE @tab1 TABLE(          tablename VARCHAR (500) collate database\_default         ,schemaname VARCHAR(500) collate database\_default  )  CREATE TABLE #temp\_Table (         [tablename] [sysname] NOT NULL,  [row\_count] [int] NULL,  [reserved] [varchar](50) NULL,  [data] [varchar](50) NULL,  [index\_size] [varchar](50) NULL,  [unused] [varchar](50) NULL)    INSERT INTO @tab1  SELECT Table\_Name, Table\_Schema  FROM information\_schema.tables  WHERE TABLE\_TYPE = 'BASE TABLE'        DECLARE c1 CURSOR FOR  SELECT Table\_Schema + '.' + Table\_Name  FROM information\_schema.tables t1  WHERE TABLE\_TYPE = 'BASE TABLE'    OPEN c1  FETCH NEXT FROM c1 INTO @table\_name  WHILE @@FETCH\_STATUS = 0  BEGIN          SET @table\_name = REPLACE(@table\_name, '[','');          SET @table\_name = REPLACE(@table\_name, ']','');            -- make sure the object exists before calling sp\_spacedused          IF EXISTS(SELECT id FROM sysobjects WHERE id = OBJECT\_ID(@table\_name))          BEGIN                 INSERT INTO #temp\_Table EXEC sp\_spaceused @table\_name, false;                       END            FETCH NEXT FROM c1 INTO @table\_name  END  CLOSE c1  DEALLOCATE c1    SELECT  t1.\*         ,t2.schemaname  FROM #temp\_Table t1  INNER JOIN @tab1 t2 ON (t1.tablename = t2.tablename )  ORDER BY schemaname,t1.tablename;    ALTER TABLE #temp\_Table ADD [DATE] DATE  UPDATE #temp\_Table SET [DATE] =  CURRENT\_TIMESTAMP    AlteR TABLE #temp\_Table ADD [DATABASE] varchar (100)  UPDATE #temp\_Table SET [DATABASE] = DB\_NAME()    CREATE TABLE #temp\_Table\_test (name varchar(100),  data\_compression tinyint,  data\_compression\_desc varchar(100))    insert into #temp\_Table\_test SELECT T.name, P.data\_compression, P.data\_compression\_desc   FROM SYS.TABLES T, SYS.partitions P  WHERE T.OBJECT\_ID = P.OBJECT\_ID  GROUP BY T.name, P.data\_compression, P.data\_compression\_desc      ALTER TABLE #temp\_Table ADD [COMPRESSED] VARCHAR (50)  UPDATE #temp\_Table  set [COMPRESSED] = b.data\_compression\_desc  from #temp\_Table a,  #temp\_Table\_test b  where a.tablename = b.name            ----------------------------------------------  insert into bic\_reference.dbo.test\_database\_stats  select \* from #temp\_Table    DROP TABLE #temp\_Table  Drop Table #temp\_Table\_test | |

|  |  |
| --- | --- |
| **Wednesday, March 06, 2013 - 4:40:16 PM - newbie** | [Read The Tip](http://www.mssqltips.com/sqlservertip/1414/run-same-command-on-all-sql-server-databases-without-cursors/) |
| I need assisance, please.  I am using the sp\_msForcEachDB and it is having difficulities.  1) although, I have the filter on the databases excluding master, model, msdb, tempdb, it is still executing the script in them.  2) when the script executes, it only seems to get a few of the databases and not all of them.  3) I have also tried putting the script into a variable and running the exec sp\_msforEachDB @command  What am I doing wrong?  I am testing the script as a print to be sure it is working before I run the execute.  Yes, this is a cursor, I realize you are all going to 'smack' me on that also.    The purpose of the script is to reorg indexes that are fragmented between 5 and 30 percent.  Here is my script... I have included the output at the bottom.    EXEC      sp\_MSforeachdb    'IF ''[?]'' NOT IN (''master'',''tempDB'',''model'',''msdb'')  BEGIN    use [?]    DECLARE @dbName sysname,@table varchar(50), @index varchar(50), @frag int, @msg varchar(500),    @sql nvarchar(500), @dbsql nvarchar(500)    declare index\_cur cursor for    SELECT OBJECT\_NAME(object\_id) AS tblName    ,[?].dbo.index\_name(object\_id, index\_id) AS ixName    ,avg\_fragmentation\_in\_percent    FROM [?].sys.dm\_db\_index\_physical\_stats(DB\_ID(), NULL, NULL, NULL, NULL)    WHERE avg\_fragmentation\_in\_percent between 5 and 30    and index\_type\_desc IN(''CLUSTERED INDEX'', ''NONCLUSTERED INDEX'')        open index\_cur    FETCH NEXT FROM index\_cur into @table, @index, @frag    WHILE @@FETCH\_STATUS = 0    BEGIN    if objectproperty (object\_id (@table), ''OwnerId'') = 1    set @sql = N''ALTER INDEX '' + quotename(@index) + '' ON '' + ''[dbo].''+ quotename(@table)    + '' REORGANIZE''    print @sql    FETCH NEXT FROM index\_cur into @table, @index, @frag        END        CLOSE index\_cur    DEALLOCATE index\_cur    END'          Msg 4121, Level 16, State 1, Line 9    Cannot find either column "tempdb" or the user-defined function or aggregate "tempdb.dbo.index\_name", or the name is ambiguous.    Msg 4121, Level 16, State 1, Line 9    Cannot find either column "model" or the user-defined function or aggregate "model.dbo.index\_name", or the name is ambiguous.    Msg 4121, Level 16, State 1, Line 9    Cannot find either column "msdb" or the user-defined function or aggregate "msdb.dbo.index\_name", or the name is ambiguous.                                                ALTER INDEX [IX\_bs\_unitanks\_FloatTable\_DateAndTime] ON [dbo].[BS\_UNITANKS\_FloatTable] REORGANIZE    ALTER INDEX [IX\_beerfilter\_FloatTable\_DateAndTime] ON [dbo].[BeerFilter\_FloatTable] REORGANIZE    ALTER INDEX [XPKItem] ON [dbo].[Item] REORGANIZE | |