

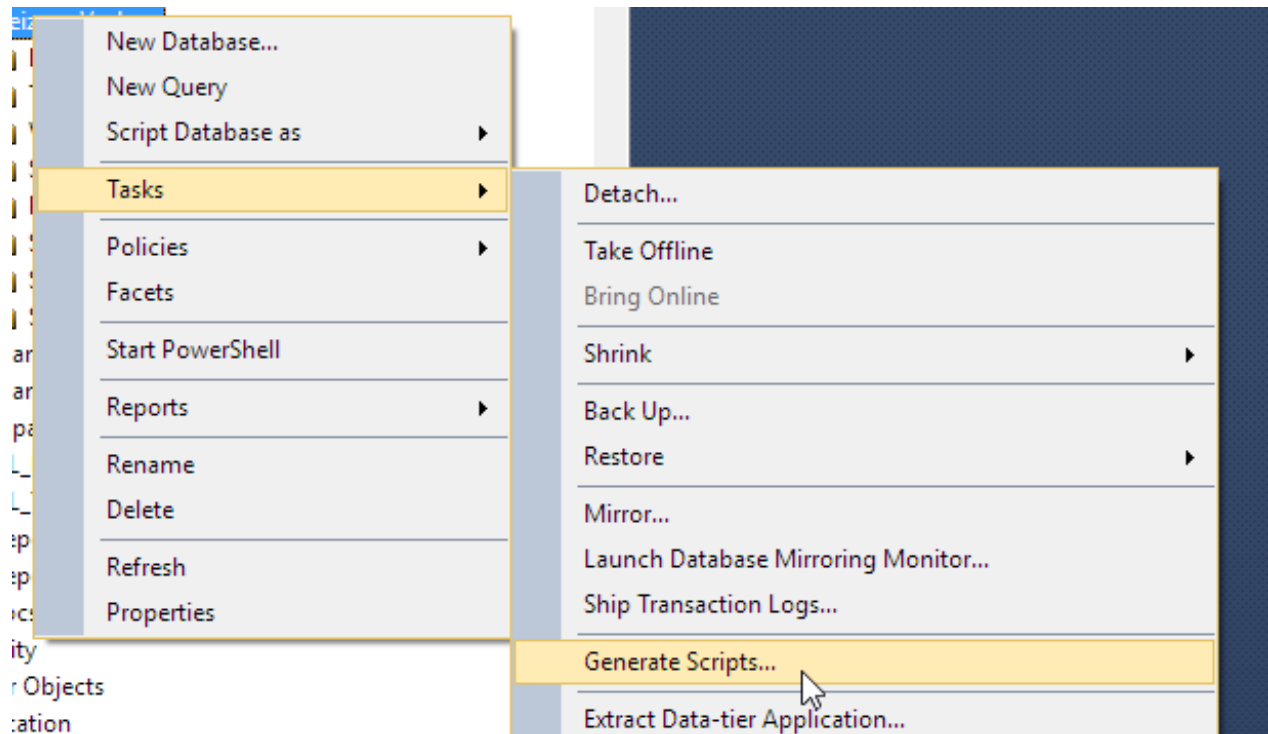
# Powershell: SQL Server Management Studio Script Generator

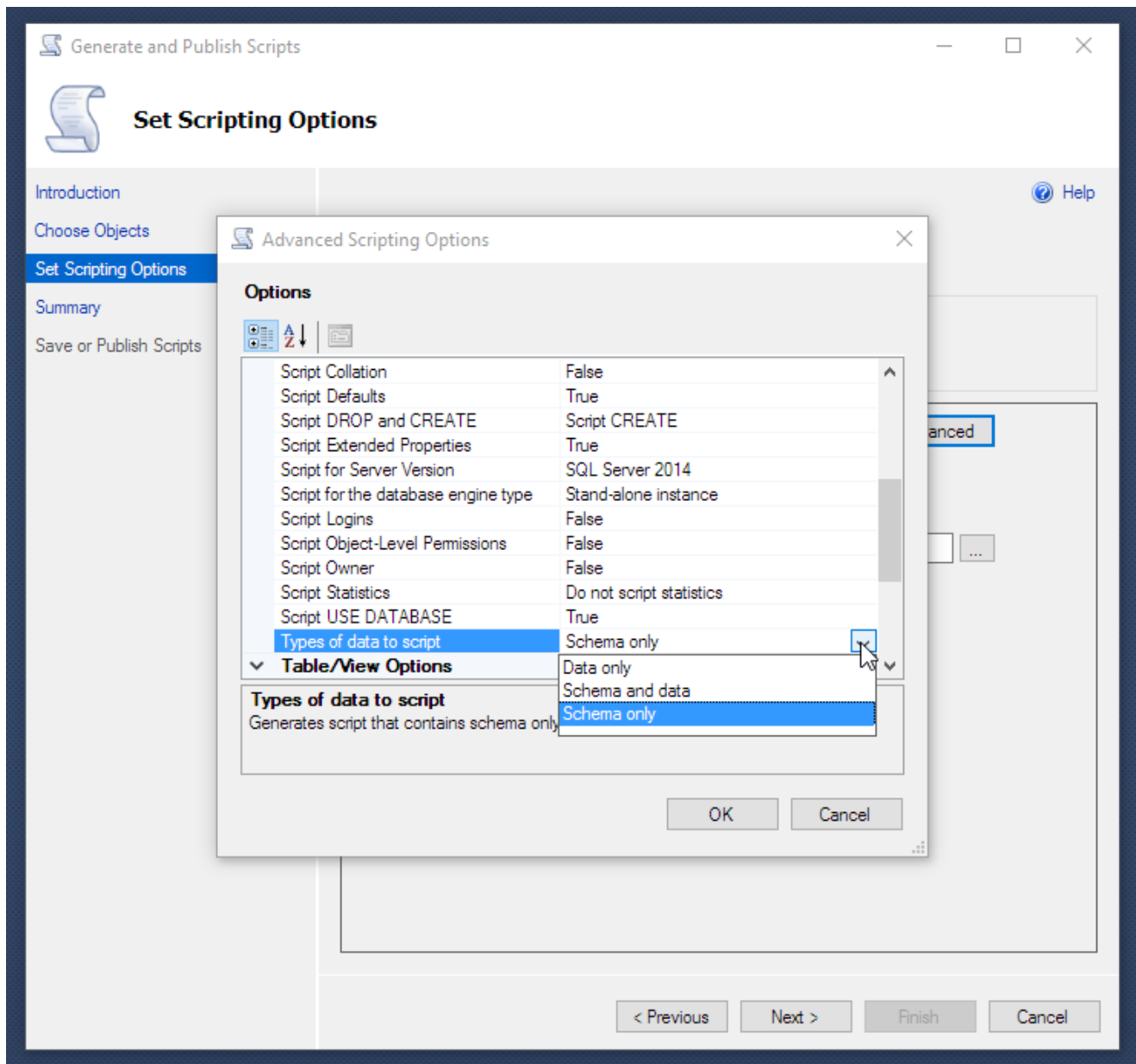
 [stackoverflow.com/questions/38633990/powershell-sql-server-management-studio-script-generator](https://stackoverflow.com/questions/38633990/powershell-sql-server-management-studio-script-generator)

2



I use the Script Generator which is integrated in the Microsoft SQL Server Management Studio to generate an import script for a whole database.





I have to do some replacements in the script which I do with Powershell. Now I want to automate the generation. Is there a way to execute exactly this Script Generator Tool (and setting some options as on the screenshot - in my case 'Data only')? Or (if this isn't possible) can I open this tool window automatically from a ps script so I don't have to open the Management Studio, selecting the DB, ...?

I found some scripts which 'manually' build the script file in Powershell but that's not exactly what I'm looking for.

Thanks!

asked Jul 28 '16 at 10:39



C.M.

61366 silver badges1313 bronze badges

3



This question's been here awhile and you've probably found your answer by now, but for those looking for a simple way to do this, the current versions of SQL server Powershell modules have native commands and methods that support this functionality from SMO.

You can use **Get-SqlDatabase** and methods such as **.Script()** and **.EnumScript()**.

For example, this will generate CREATE scripts for user defined functions and save it to file:

```
$Database = Get-SqlDatabase -ServerInstance $YourSqlServer -Name $YourDatabaseName
```

```
$MyFuncs = $Database.UserDefinedFunctions | Where Schema -eq "dbo"  
$MyFuncs.Script() | Out-File -FilePath ".\SqlScripts\MyFunctions.sql"
```

If you want to **script data** and elements like indexes, keys, triggers, etc. you will have to specify the scripting options, like this:

```
$scriptOptions = New-Object -TypeName Microsoft.SqlServer.Management.Smo.ScriptingOptions
```

```
$scriptOptions.NoCollation = $True  
$scriptOptions.Indexes = $True  
$scriptOptions.Triggers = $True  
$scriptOptions.DriAll = $True  
$scriptOptions.ScriptData = $True
```

```
$Database.Tables.EnumScript($scriptOptions) | Out-File -FilePath ".\AllMyTables.sql"
```

Note that the Script() method doesn't support scripting data. Use EnumScript() for tables.

If you want to script data only, as asked, you can try `$scriptOptions.ScriptData = $True` and `$scriptOptions.ScriptSchema = $False` .

answered Nov 15 '18 at 16:53



Victor Lopes

3133 bronze badges

## Your Answer

---

Sign up or [log in](#)



Sign up using Google



Sign up using Facebook



Sign up using Email and Password

---

Post as a guest

---

Required, but never shown

Not the answer you're looking for? Browse other questions tagged [sql](#) [sql-server](#) [powershell](#) [ssms](#) or [ask your own question](#).

---