

In SQL Server, how do I generate a CREATE TABLE statement for a given table?

Asked 16 years, 4 months ago Modified 4 years, 3 months ago

Viewed 134k times



88

I've spent a good amount of time coming up with solution to this problem, so in the spirit of [this post](#), I'm posting it here, since I think it might be useful to others.



If anyone has a better script, or anything to add, please post it.



Edit: Yes guys, I know how to do it in Management Studio - but I needed to be able to do it from within another application.

sql-server

scripting

dynamic-sql

Share

Improve this question

Follow

edited May 23, 2017 at 10:31



Community Bot

1 • 1

asked Aug 22, 2008 at 0:58



Blorgbeard

103k • 50 • 235 • 276

(Very) late to the party, but I found a GitHub repo for this purpose. It was a tool written by Microsoft IIRC, but unfortunately the repo was archived. And now when I try to find it again, I can't find it... utterly annoying. If anyone else finds it, please share the link here. – [Per Lundberg](#) Sep 18 at 6:31

- 1 Found it myself in the browser history. YMMV, but this could be worth a try: github.com/microsoft/mssql-scripiter
– [Per Lundberg](#) Sep 18 at 10:14

16 Answers

Sorted by:

Highest score (default)



119



I've modified the version above to run for all tables and support new SQL 2005 data types. It also retains the primary key names. Works only on SQL 2005 (using cross apply).



```
select  'create table [' + so.name + '] (' + o.list +
        tc.Constraint_Name IS NULL THEN '' ELSE 'ALTER TABLE '
CONSTRAINT ' + tc.Constraint_Name + ' PRIMARY KEY ' +
        Len(j.List)-1) + ')' END
from    sysobjects so
cross apply
        (SELECT
            ' ['+column_name+'] ' +
            data_type + case data_type
                when 'sql_variant' then ''
                when 'text' then ''
                when 'ntext' then ''
                when 'xml' then ''
                when 'decimal' then '(' + cast(numeric_pre
+ cast(numeric_scale as varchar) + ')'
                else coalesce('(' + case when character_maxi
'MAX' else cast(character_maximum_length as varchar) e
```

```

        case when exists (
        select id from syscolumns
        where object_name(id)=so.name
        and name=column_name
        and columnproperty(id,name,'IsIdentity') = 1
        ) then
        'IDENTITY(' +
        cast(ident_seed(so.name) as varchar) + ',' +
        cast(ident_incr(so.name) as varchar) + ')'
        else ''
        end + ' ' +
        (case when UPPER(IS_NULLABLE) = 'NO' then 'NO
' +
        case when information_schema.columns.COLUMN_
'DEFAULT '+ information_schema.columns.COLUMN_DEFAULT

        from information_schema.columns where table_name
        order by ordinal_position
        FOR XML PATH('')) o (list)
left join
        information_schema.table_constraints tc
on tc.Table_name      = so.Name
AND tc.Constraint_Type = 'PRIMARY KEY'
cross apply
        (select '[' + Column_Name + ']', '
        FROM    information_schema.key_column_usage kcu
        WHERE    kcu.Constraint_Name = tc.Constraint_Name
        ORDER BY
                ORDINAL_POSITION
        FOR XML PATH('')) j (list)
where xtype = 'U'
AND name NOT IN ('dtproperties')

```

Update: Added handling of the XML data type

Update 2: Fixed cases when 1) there is multiple tables with the same name but with different schemas, 2) there is multiple tables having PK constraint with the same name

Share Improve this answer

edited May 24, 2020 at 3:28

Follow



MasterOdin

7,856 ● 1 ● 22 ● 39

answered Nov 25, 2008 at 16:04



David

6 This is fantastic! I've been looking for something like this for quite some time! (trying to script table structure into our version control) – [Matt](#) May 26, 2009 at 21:21

2 Thank you for posting this! I think you should be using the `numeric_precision` column, not `numeric_precision_radix`, and in addition to the decimal datatype, it should work the same for the numeric datatype. – [Scott Whitlock](#) Jun 24, 2010 at 18:05

2 I think you have an error -- `numeric_precision_radix` should be `numeric_precision` – [BlueMonkMN](#) Jan 12, 2011 at 15:01

5 I noticed when running this that the output has a extra comma at the end when listing the columns. I was able to remove the extra comma by surrounding the `o.list` in line 1 with `STUFF(o.list, 1, 2, '')` and moving the comma from the end of the cross apply on line 23 to the start in line 5 `' ,` `['+column_name+'] ' +` – [Imingle](#) Oct 27, 2011 at 15:50



With @Imingle change I also changed `LEFT(j.List, Len(j.List)-1)` to `STUFF(j.List, 1, 2, '')` and moved the comma on line 33 to the front `select ' , [' + Column_Name + '] ' – Schalk Versteeg Mar 26, 2012 at 16:18`



37



Here's the script that I came up with. It handles Identity columns, default values, and primary keys. It does not handle foreign keys, indexes, triggers, or any other clever stuff. It works on SQLServer 2000, 2005 and 2008.

```
declare @schema varchar(100), @table varchar(100)
set @schema = 'dbo' -- set schema name here
set @table = 'MyTable' -- set table name here
declare @sql table(s varchar(1000), id int identity)

-- create statement
insert into @sql(s) values ('create table [' + @table

-- column list
insert into @sql(s)
select
    ' [' + column_name + ']' +
    data_type + coalesce('(' + cast(character_maximum_le
+ ' ' +
    case when exists (
        select id from syscolumns
        where object_name(id)=@table
        and name=column_name
        and columnproperty(id,name,'IsIdentity') = 1
    ) then
        'IDENTITY(' +
        cast(ident_seed(@table) as varchar) + ',' +
        cast(ident_incr(@table) as varchar) + ')'
    else ''
    end + ' ' +
    ( case when IS_NULLABLE = 'No' then 'NOT ' else ''
    coalesce('DEFAULT ' + COLUMN_DEFAULT, '') + ', '

from INFORMATION_SCHEMA.COLUMNS where table_name = @t
@schema
order by ordinal_position

-- primary key
declare @pkname varchar(100)
select @pkname = constraint_name from INFORMATION_SCHE
where table_name = @table and constraint_type='PRIMARY
```

```

if ( @pkname is not null ) begin
    insert into @sql(s) values(' PRIMARY KEY (')
    insert into @sql(s)
        select ' ['+COLUMN_NAME+'], ' from INFORMATION
        where constraint_name = @pkname
        order by ordinal_position
    -- remove trailing comma
    update @sql set s=left(s,len(s)-1) where id=@@iden
    insert into @sql(s) values ( ' )' )
end
else begin
    -- remove trailing comma
    update @sql set s=left(s,len(s)-1) where id=@@iden
end

-- closing bracket
insert into @sql(s) values( ' )' )

-- result!
select s from @sql order by id

```

Share Improve this answer

edited Feb 25, 2019 at 16:20

Follow



Matt and Neil

157 ● 11

answered Aug 22, 2008 at 1:03



Blorgbeard

103k ● 50 ● 235 ● 276

Lots of systems won't be able to run this due to the dynamic sql. On your local box probably not a problem, but just wanted to point that out. – [Jason Short](#) Oct 10, 2008 at 1:15

6 What dynamic sql? This script generates sql - it doesn't execute it. – [Blorgbeard](#) Oct 15, 2008 at 23:57

Would love to see this broken down into steps so it can be performed with a series of queries that can be written in C++

– [Mike Trader](#) Apr 30, 2011 at 8:58

You're missing support for the Schema if it is not 'dbo'. Also, if a database has the same table name with different schema, problems will arise. I got out of it what I needed though, thanks. – [DavidStein](#) Jun 17, 2013 at 23:53

@TT. are you sure? `select cast(123456 as varchar)` returns `123456` for me. – [Blorgbeard](#) Feb 18, 2016 at 19:28



There is a Powershell script buried in the [msdb](#) forums that will script all the tables and related objects:

15



```
# Script all tables in a database
[System.Reflection.Assembly]::LoadWithPartialName("Microsoft.SqlServer.Management.Smo") | out-null

$Ss = new-object ('Microsoft.SqlServer.Management.Smo') $Server
$db = $Ss.Databases['<Database>']

$scrp = new-object ('Microsoft.SqlServer.Management.Smo') $Server
$scrp.Options.AppendToFile = $True
$scrp.Options.ClusteredIndexes = $True
$scrp.Options.DriAll = $True
$scrp.Options.ScriptDrops = $False
$scrp.Options.IncludeHeaders = $False
$scrp.Options.ToFileOnly = $True
$scrp.Options.Indexes = $True
$scrp.Options.WithDependencies = $True
$scrp.Options.FileName = 'C:\Temp\<Database>.SQL'

foreach($item in $db.Tables) { $tablearray+=@($item) }
$scrp.Script($tablearray)

Write-Host "Scripting complete"
```

Share Improve this answer

answered Apr 11, 2012 at 23:47

Follow



8kb

11.4k ● 7 ● 40 ● 52

SMO is great when you have permissions to use it. I've found many systems where that wasn't the case though, so a pure INFORMATION_SCHEMA approach has some advantages.

– dsz Jun 4, 2019 at 4:11



13



Support for schemas:

This is an updated version that amends the great answer from David, et al. Added is support for named schemas. It should be noted this may break if there's actually tables of the same name present within various schemas.

Another improvement is the use of the official

[QuoteName\(\)](#) function.

```
SELECT
    t.TABLE_CATALOG,
    t.TABLE_SCHEMA,
    t.TABLE_NAME,
    'create table '+QuoteName(t.TABLE_SCHEMA)+'.' + Qu
LEFT(o.List, Len(o.List)-1) + '); '
    + CASE WHEN tc.Constraint_Name IS NULL THEN ''
    ELSE
        'ALTER TABLE ' + QuoteName(t.TABLE_SCHEMA)
        + ' ADD CONSTRAINT ' + tc.Constraint_Name
+ LEFT(j.List, Len(j.List)-1) + '); '
    END as 'SQL_CREATE_TABLE'
FROM sysobjects so

CROSS APPLY (
    SELECT
        ' ['+column_name+'] '
        + data_type
```



```

+ case data_type
    when 'sql_variant' then ''
    when 'text' then ''
    when 'ntext' then ''
    when 'decimal' then '(' + cast(numeric
', ' + cast(numeric_scale as varchar) + ')'
    else
        coalesce(
            '(' + case when character_maximum_lengt
                then 'MAX'
                else cast(character_maximum_length
                    + ')', '')
        end
+ ' '
+ case when exists (
    SELECT id
    FROM syscolumns
    WHERE
        object_name(id) = so.name
        and name = column_name
        and columnproperty(id,name,'IsIdentity
) then
    'IDENTITY(' +
        cast(ident_seed(so.name) as varchar) + ','
        cast(ident_incr(so.name) as varchar) + ')'
    else ''
    end
+ ' '
+ (case when IS_NULLABLE = 'No' then 'NOT ' el
+ 'NULL '
+ case when information_schema.columns.COLUMN_
'DEFAULT ' + information_schema.columns.COLUMN_DEFAULT
    ELSE ''
    END
+ ', ' -- can't have a field name or we'll end

FROM information_schema.columns
WHERE table_name = so.name
ORDER BY ordinal_position
FOR XML PATH('')
) o (list)

LEFT JOIN information_schema.table_constraints tc on
tc.Table_name = so.Name

```

```

        AND tc.Constraint_Type = 'PRIMARY KEY'

LEFT JOIN information_schema.tables t on
    t.Table_name = so.Name

CROSS APPLY (
    SELECT QuoteName(Column_Name) + ', '
    FROM information_schema.key_column_usage kcu
    WHERE kcu.Constraint_Name = tc.Constraint_Name
    ORDER BY ORDINAL_POSITION
    FOR XML PATH('')
) j (list)

WHERE
    xtype = 'U'
    AND name NOT IN ('dtproperties')
    -- AND so.name = 'ASPStateTempSessions'
;

```

..

For use in Management Studio:

One detractor to the sql code above is if you test it using SSMS, long statements aren't easy to read. So, as per [this helpful post](#), here's another version that's somewhat modified to be easier on the eyes after clicking the link of a cell in the grid. The results are more readily identifiable as nicely formatted CREATE TABLE statements for each table in the db.

```

-- settings
DECLARE @CRLF NCHAR(2)
SET @CRLF = Nchar(13) + NChar(10)
DECLARE @PLACEHOLDER NCHAR(3)
SET @PLACEHOLDER = '{:}'

-- the main query

```

```

SELECT
    t.TABLE_CATALOG,
    t.TABLE_SCHEMA,
    t.TABLE_NAME,
    CAST(
        REPLACE(
            'create table ' + QuoteName(t.TABLE_SCHEMA)
            + QuoteName(so.name) + ' (' + @CRLF
            + LEFT(o.List, Len(o.List) - (LEN(@PLACEH
            + @CRLF
            + CASE WHEN tc.Constraint_Name IS NULL THE
            ELSE
            'ALTER TABLE ' + QuoteName(t.TABLE_SCH
            QuoteName(so.Name)
            + ' ADD CONSTRAINT ' + tc.Constraint_N
            LEFT(j.List, Len(j.List) - 1) + ');' + @CRLF
            END,
            @PLACEHOLDER,
            @CRLF
        )
        AS XML) as 'SQL_CREATE_TABLE'
FROM sysobjects so

CROSS APPLY (
    SELECT
        ' '
        + '['+column_name+'] '
        + data_type
        + case data_type
            when 'sql_variant' then ''
            when 'text' then ''
            when 'ntext' then ''
            when 'decimal' then '(' + cast(numeric
            ', ' + cast(numeric_scale as varchar) + ')'
            else
            coalesce(
                '('+ case when character_maximum_lengt
                then 'MAX'
                else cast(character_maximum_length
                + ')', '')
            end
        + ' '
        + case when exists (
            SELECT id

```

```

        FROM syscolumns
        WHERE
            object_name(id) = so.name
            and name = column_name
            and columnproperty(id,name,'IsIdentity
    ) then
        'IDENTITY(' +
        cast(ident_seed(so.name) as varchar) + ','
        cast(ident_incr(so.name) as varchar) + ')'
    else ''
    end
+ ' '
+ (case when IS_NULLABLE = 'No' then 'NOT ' el
+ 'NULL '
+ case when information_schema.columns.COLUMN_
'DEFAULT ' + information_schema.columns.COLUMN_DEFAULT
    ELSE ''
    END
+ ', '
+ @PLACEHOLDER -- note, can't have a field na

```

XML

```

        FROM information_schema.columns where table_name =
        ORDER BY ordinal_position
        FOR XML PATH('')
    ) o (list)

```

```

LEFT JOIN information_schema.table_constraints tc on
    tc.Table_name = so.Name
    AND tc.Constraint_Type = 'PRIMARY KEY'

```

```

LEFT JOIN information_schema.tables t on
    t.Table_name = so.Name

```

```

CROSS APPLY (
    SELECT QUOTENAME(Column_Name) + ', '
    FROM information_schema.key_column_usage kcu
    WHERE kcu.Constraint_Name = tc.Constraint_Name
    ORDER BY ORDINAL_POSITION
    FOR XML PATH('')
) j (list)

```

```

WHERE
    xtype = 'U'

```

```

AND name NOT IN ('dtproperties')
-- AND so.name = 'ASPStateTempSessions'
;

```

Not to belabor the point, but here's the functionally equivalent example outputs for comparison:

```

-- 1 (scripting version)
create table [dbo].[ASPStateTempApplications] ( [AppI
[AppName] char(280) NOT NULL ); ALTER TABLE [dbo].[A
ADD CONSTRAINT PK__ASPState__8E2CF7F908EA5793 PRIMARY

-- 2 (SSMS version)
create table [dbo].[ASPStateTempSessions] (
    [SessionId] nvarchar(88) NOT NULL ,
    [Created] datetime NOT NULL DEFAULT (getutcdate())
    [Expires] datetime NOT NULL ,
    [LockDate] datetime NOT NULL ,
    [LockDateLocal] datetime NOT NULL ,
    [LockCookie] int NOT NULL ,
    [Timeout] int NOT NULL ,
    [Locked] bit NOT NULL ,
    [SessionItemShort] varbinary(7000) NULL ,
    [SessionItemLong] image(2147483647) NULL ,
    [Flags] int NOT NULL DEFAULT ((0))
);
ALTER TABLE [dbo].[ASPStateTempSessions] ADD CONSTRAIN
PK__ASPState__C9F4929003317E3D PRIMARY KEY ([SessionId

```

..

Detracting factors:

It should be noted that I remain relatively unhappy with this due to the lack of support for indeces other than a primary key. It remains suitable for use as a mechanism for simple data export or replication.

Share Improve this answer

edited May 23, 2017 at 10:31

Follow



Community Bot

1 • 1

answered Mar 26, 2013 at 19:31



zanlok

1,630 • 1 • 16 • 29

This actually works for the Stack Exchange Data Explorer, super useful! – [Sienna](#) Dec 24, 2016 at 23:23

In the case when statement for the datatypes, I added an additional line for 'numeric' that duplicates the same setup for 'decimal'. – [William](#) Apr 15 at 19:27



8

If the application you are generating the scripts from is a .NET application, you may want to look into using SMO (Sql Management Objects). Reference this [SQL Team link](#) on how to use SMO to script objects.



Share Improve this answer

answered Nov 25, 2008 at 16:12

Follow



user25623

245 • 2 • 3



8

One more variant with foreign keys support and in one statement:



```
SELECT
    obj.name
    , 'CREATE TABLE [' + obj.name + '] (' + LEFT(co
1 ) + ')'
    + ISNULL(' ' + refs.list, '')
```



```
FROM sysobjects obj
CROSS APPLY (
    SELECT
        CHAR(10)
        + ' [' + column_name + ' ] '
        + data_type
        + CASE data_type
            WHEN 'sql_variant' THEN ''
            WHEN 'text' THEN ''
            WHEN 'ntext' THEN ''
            WHEN 'xml' THEN ''
            WHEN 'decimal' THEN '(' + CAST(numeric
', ' + CAST(numeric_scale as VARCHAR) + ')'
            ELSE COALESCE('(' + CASE WHEN characte
THEN 'MAX' ELSE CAST(character_maximum_length as VARCH
END
        + ' '
        + case when exists ( -- Identity skip
select id from syscolumns
where object_name(id) = obj.name
and name = column_name
and columnproperty(id,name,'IsIdentity') =
) then
'IDENTITY(' +
cast(ident_seed(obj.name) as varchar) + ',
cast(ident_incr(obj.name) as varchar) + '
else ''
end + ' '
        + CASE WHEN IS_NULLABLE = 'No' THEN 'NOT '
        + 'NULL'
        + CASE WHEN information_schema.columns.col
THEN ' DEFAULT ' + information_schema.columns.column_d
        + ', '
    FROM
        INFORMATION_SCHEMA.COLUMNS
    WHERE table_name = obj.name
    ORDER BY ordinal_position
    FOR XML PATH('')
) cols (list)
CROSS APPLY(
    SELECT
        CHAR(10) + 'ALTER TABLE ' + obj.name + '_n
LEFT(alt, LEN(alt)-1)
    FROM(
```

```

SELECT
    CHAR(10)
    + ' CONSTRAINT ' + tc.constraint_name
    + ' ' + tc.constraint_type + ' (' + LE
+ '))'
    + COALESCE(CHAR(10) + r.list, ', ')
FROM
    information_schema.table_constraints t
CROSS APPLY(
    SELECT
        '[' + kcu.column_name + ']', '
    FROM
        information_schema.key_column_
WHERE
        kcu.constraint_name = tc.const
ORDER BY
        kcu.ordinal_position
    FOR XML PATH('')
) c (list)
OUTER APPLY(
    -- // http://stackoverflow.com/question/11111111/server-howto-get-foreign-key-reference-from-information-schema
    SELECT
        ' REFERENCES [' + kcu1.constr
+ kcu2.table_name + ']' + '(' + kcu2.column_name + '),
    FROM information_schema.referentia
        JOIN information_schema.key_co
(kcu1.constraint_catalog = rc.constraint_catalog AND k
rc.constraint_schema AND kcu1.constraint_name = rc.con
        JOIN information_schema.key_co
(kcu2.constraint_catalog = rc.unique_constraint_catalo
kcu2.constraint_schema = rc.unique_constraint_schema A
rc.unique_constraint_name AND kcu2.ordinal_position =
WHERE
        kcu1.constraint_catalog = tc.c
kcu1.constraint_schema = tc.constraint_schema AND kcu1
tc.constraint_name
    ) r (list)
WHERE tc.table_name = obj.name
FOR XML PATH('')
) a (alt)
) refs (list)
WHERE
    xtype = 'U'

```



```
AND name NOT IN ('dtproperties')
AND obj.name = 'your_table_name'
```

You could try in is sqlfiddle:

<http://sqlfiddle.com/#!6/e3b66/3/0>

Share Improve this answer

edited Nov 16, 2015 at 12:49

Follow

answered Sep 4, 2013 at 16:45



Hubbitus

5,351 ● 3 ● 43 ● 49

-
- 2 I like this one for its single statement solution and it supports foreign keys. The first 2 instances of `@tableName` can be replaced with `so.name` making it easier to generate create table scripts for multiple tables at once.

– [Christiaan Westerbeek](#) Nov 14, 2015 at 8:41

Also when there's not a single constraint, like a primary key, the query would be `NULL`. To solve this `' ' + j.list` should be `ISNULL(' ' + j.list, '')`. I took the liberty to adjust the code for the use cases described in both of my comments – [Christiaan Westerbeek](#) Nov 14, 2015 at 15:55



@ChristiaanWesterbeek suggestions looks reasonable. But I do not understand last case. When `j.list` will be null? May be example? – [Hubbitus](#) Nov 14, 2015 at 17:38

-
- 2 Try it with the table district in your fiddle. You'll see a null. Also the last column has a trailing comma that should not be there. Still like your answer though :)

– [Christiaan Westerbeek](#) Nov 14, 2015 at 23:17

- 2 Ah, you speak about tables does not have any keys! Thanks, it fixed too. Last comma is not issue because it correct SQL, but it (among with more readable names of subqueries) also fixed for estetic. – [Hubbitus](#) Nov 16, 2015 at 12:51
-



7

I modified the accepted answer and now it can get the command including primary key and foreign key in a certain schema.



```
declare @table varchar(100)
declare @schema varchar(100)
set @table = 'Persons' -- set table name here
set @schema = 'OT' -- set SCHEMA name here
declare @sql table(s varchar(1000), id int identity)

-- create statement
insert into @sql(s) values ('create table ' + @table

-- column list
insert into @sql(s)
select
    ' ' + column_name + ' ' +
    data_type + coalesce('(' + cast(character_maximum_le
+ ' ' +
    case when exists (
        select id from syscolumns
        where object_name(id)=@table
        and name=column_name
        and columnproperty(id,name,'IsIdentity') = 1
    ) then
        'IDENTITY(' +
        cast(ident_seed(@table) as varchar) + ',' +
        cast(ident_incr(@table) as varchar) + ')'
    else ''
end + ' ' +
    ( case when IS_NULLABLE = 'No' then 'NOT ' else ''
    coalesce('DEFAULT '+COLUMN_DEFAULT, '') + ','
from information_schema.columns where table_name = @t
```

```

@schema
order by ordinal_position

-- primary key
declare @pkname varchar(100)
select @pkname = constraint_name from information_sche
where table_name = @table and constraint_type='PRIMARY'

if ( @pkname is not null ) begin
    insert into @sql(s) values(' PRIMARY KEY (')
    insert into @sql(s)
        select ' '+COLUMN_NAME+', ' from information_
        where constraint_name = @pkname
        order by ordinal_position
    -- remove trailing comma
    update @sql set s=left(s,len(s)-1) where id=@@iden
    insert into @sql(s) values (' )')
end
else begin
    -- remove trailing comma
    update @sql set s=left(s,len(s)-1) where id=@@iden
end

-- foreign key
declare @fkname varchar(100)
select @fkname = constraint_name from information_sche
where table_name = @table and constraint_type='FOREIGN'

if ( @fkname is not null ) begin
    insert into @sql(s) values(',')
    insert into @sql(s) values(' FOREIGN KEY (')
    insert into @sql(s)
        select ' '+COLUMN_NAME+', ' from information_
        where constraint_name = @fkname
        order by ordinal_position
    -- remove trailing comma
    update @sql set s=left(s,len(s)-1) where id=@@iden
    insert into @sql(s) values (' ) REFERENCES ')
    insert into @sql(s)
        SELECT
            OBJECT_NAME(fk.referenced_object_id)
        FROM
            sys.foreign_keys fk

```

```

INNER JOIN
    sys.foreign_key_columns fkc ON fkc.constra
fk.object_id
INNER JOIN
    sys.columns c1 ON fkc.parent_column_id = c
fk.parent_object_id = c1.object_id
INNER JOIN
    sys.columns c2 ON fkc.referenced_column_id
fk.referenced_object_id = c2.object_id
where fk.name = @fkname
insert into @sql(s)
SELECT
    '('+c2.name+ ')'
FROM
    sys.foreign_keys fk
INNER JOIN
    sys.foreign_key_columns fkc ON fkc.constra
fk.object_id
INNER JOIN
    sys.columns c1 ON fkc.parent_column_id = c
fk.parent_object_id = c1.object_id
INNER JOIN
    sys.columns c2 ON fkc.referenced_column_id
fk.referenced_object_id = c2.object_id
where fk.name = @fkname
end

-- closing bracket
insert into @sql(s) values( ' )' )

-- result!
select s from @sql order by id

```

Share Improve this answer

answered Aug 15, 2014 at 10:06

Follow



JasmineOT

2,068 ● 2 ● 22 ● 30

-
- 1 Might be nice to republish this (edit) into the accepted answer. That seems adequately in the spirit of SO. Schemas and newlines from my answer would be nice as well for a



I'm going to improve the answer by supporting partitioned tables:

6

find partition scheme and partition key using below scripts:



```
declare @partition_scheme varchar(100) = (  
select distinct ps.Name AS PartitionScheme  
from sys.indexes i  
join sys.partitions p ON i.object_id=p.object_id AND i  
join sys.partition_schemes ps on ps.data_space_id = i.  
where i.object_id = object_id('your table name')  
)  
print @partition_scheme  
  
declare @partition_column varchar(100) = (  
select c.name  
from sys.tables t  
join sys.indexes i  
on(i.object_id = t.object_id  
and i.index_id < 2)  
join sys.index_columns ic  
on(ic.partition_ordinal > 0  
and ic.index_id = i.index_id and ic.object_id = t.ob  
join sys.columns c  
on(c.object_id = ic.object_id  
and c.column_id = ic.column_id)  
where t.object_id = object_id('your table name')  
)  
print @partition_column
```

then change the generation query by adding below line at the right place:

```
+ IIF(@partition_scheme is null, '', 'ON [' + @partiti  
@partition_column + ']))')
```

Share Improve this answer

answered Jan 29, 2016 at 19:08

Follow



FLICKER

6,652 ● 4 ● 51 ● 80



Credit due to @Blorgbeard for sharing his script. I'll certainly bookmark it in case I need it.

4



Yes, you can "right click" on the table and script the

`CREATE TABLE` script, but:

- The a script will contain **loads** of cruft (interested in the extended properties anyone?)
- If you have 200+ tables in your schema, it's going to take you half a day to script the lot by hand.



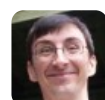
With this script converted into a stored procedure, and combined with a wrapper script you would have a nice automated way to dump your table design into source control etc.

The rest of your DB code (SP's, FK indexes, Triggers etc) would be under source control anyway ;)

Share Improve this answer


answered Aug 22, 2008 at 11:45

Follow



Guy

9,826 ● 7 ● 39 ● 43

I think a dbproject can import the schema directly from the server, so you dont need to checkin the create table statements, and in SSMS you can script out multiple tables, either by using "Object Explorer Details" or right clicking on the database in object explorer and selecting Tasks->Generate scripts – [Akash](#) Apr 15, 2015 at 18:17 



4



Something I've noticed - in the INFORMATION_SCHEMA.COLUMNS view, CHARACTER_MAXIMUM_LENGTH gives a size of 2147483647 ($2^{31}-1$) for field types such as image and text. ntext is $2^{30}-1$ (being double-byte unicode and all).



This size is included in the output from this query, but it is invalid for these data types in a CREATE statement (they should not have a maximum size value at all). So unless the results from this are manually corrected, the CREATE script won't work given these data types.

I imagine it's possible to fix the script to account for this, but that's beyond my SQL capabilities.

Share Improve this answer

[edited Sep 23, 2008 at 8:21](#)

Follow

answered Sep 23, 2008 at 8:14



[Jivlain](#)

3,648 ● 1 ● 28 ● 32



-- or you could create a stored procedure ... first with Id creation

4



```
USE [db]
GO

/***** Object:  StoredProcedure [dbo].[procUtils_Inse
Script Date: 06/13/2009 22:18:11 *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

create PROC [dbo].[procUtils_InsertGeneratorWithId]
(
    @domain_user varchar(50),
    @tableName varchar(100)
)

as

--Declare a cursor to retrieve column specific informa
table
DECLARE cursCol CURSOR FAST_FORWARD FOR
SELECT column_name,data_type FROM information_schema.c
@tableName
OPEN cursCol
DECLARE @string nvarchar(3000) --for storing the first
DECLARE @stringData nvarchar(3000) --for storing the d
statement
DECLARE @dataType nvarchar(1000) --data types returned
DECLARE @IDENTITY_STRING nvarchar ( 100 )
SET @IDENTITY_STRING = ' '
select  @IDENTITY_STRING
SET @string='INSERT '+@tableName+'('
SET @stringData=''

DECLARE @colName nvarchar(50)
```



```

FETCH NEXT FROM cursCol INTO @colName,@dataType

IF @@fetch_status<>0
begin
print 'Table '+@tableName+' not found, processing ski
close curscol
deallocate curscol
return
END

WHILE @@FETCH_STATUS=0
BEGIN
IF @dataType in ('varchar','char','nchar','nvarchar')
BEGIN
--SET @stringData=@stringData+'''+isnull('+@col
SET
@stringData=@stringData+'''+'''+isnull(''+'''+'''+
END
ELSE
if @dataType in ('text','ntext') --if the datatype is
BEGIN
SET @stringData=@stringData+'''+isnull(cast('+@
varchar(2000)),''')+''','''+
END
ELSE
IF @dataType = 'money' --because money doesn't get con
implicitly
BEGIN
SET @stringData=@stringData+'''+convert(money, '''+i
as varchar(200)), '0.0000')+''','''+
END
ELSE
IF @dataType='datetime'
BEGIN
--SET
@stringData=@stringData+'''+convert(datetime, '''+isn
varchar(200)), '0')+''','''+
--SELECT 'INSERT Authorizations(StatusDate)
VALUES('+convert(datetime, '+isnull(''+'+convert(varch
FROM Authorizations
--SET @stringData=@stringData+'''+convert(money, '''+
as varchar(200)), '0.0000')+''','''+
SET

```

```

@stringData=@stringData+''convert(datetime,'+''+isnu
--
'convert(datetime,'+isnull(''+convert(varchar(200),S
FROM Authorizations
END
ELSE
IF @dataType='image'
BEGIN
SET
@stringData=@stringData+'''+isnull(cast(convert(
as varchar(6)), '0'))+''', ''+
END
ELSE --presuming the data type is int,bit,numeric,deci
BEGIN
--SET @stringData=@stringData+'''+isnull(cast('
varchar(200)), '0'))+''', ''+
--SET
@stringData=@stringData+''convert(datetime,'+''+isnu
SET
@stringData=@stringData+'''+'''+isnull(''+'+co
END

SET @string=@string+@colName+', '

FETCH NEXT FROM cursCol INTO @colName,@dataType
END
DECLARE @Query nvarchar(4000)

SET @query ='SELECT '''+substring(@string,0,len(@strin
substring(@stringData,0,len(@stringData)-2)+''+''')''
exec sp_executesql @query
--select @query

CLOSE cursCol
DEALLOCATE cursCol

/*
USAGE

*/

GO

```

-- and second without iD INSERTION

```
USE [db]
GO

/***** Object:  StoredProcedure [dbo].[procUtils_Inse
Date: 06/13/2009 22:20:52 *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE PROC [dbo].[procUtils_InsertGenerator]
(
    @domain_user varchar(50),
    @tableName varchar(100)
)

as

--Declare a cursor to retrieve column specific informa
table
DECLARE cursCol CURSOR FAST_FORWARD FOR

-- SELECT column_name,data_type FROM information_schem
table_name = @tableName
/* NEW
SELECT c.name , sc.data_type FROM sys.extended_proper
INNER JOIN sys.tables AS t ON ep.major_id = t.object_i
INNER JOIN sys.columns AS c ON ep.major_id = c.object_
= c.column_id
INNER JOIN INFORMATION_SCHEMA.COLUMNS sc ON t.name = s
c.name = sc.column_name
WHERE t.name = @tableName and c.is_identity=0
*/

select object_name(c.object_id) "TABLE_NAME", c.name "
"DATA_TYPE"
    from sys.columns c
    join sys.systypes s on (s.xtype = c.system_type_id)
```

```

where object_name(c.object_id) in (select name from
like 'sysdiagrams')
AND object_name(c.object_id) in (select name from s
[name]=@tableName ) and c.is_identity=0 and s.name no

```

```

OPEN cursCol
DECLARE @string nvarchar(3000) --for storing the first
DECLARE @stringData nvarchar(3000) --for storing the d
statement
DECLARE @dataType nvarchar(1000) --data types returned
DECLARE @IDENTITY_STRING nvarchar ( 100 )
SET @IDENTITY_STRING = ' '
select @IDENTITY_STRING
SET @string='INSERT '+@tableName+'('
SET @stringData=''

```

```

DECLARE @colName nvarchar(50)

```

```

FETCH NEXT FROM cursCol INTO @tableName , @colName,@da

```

```

IF @@fetch_status<>0
begin
print 'Table '+@tableName+' not found, processing ski
close curscol
deallocate curscol
return
END

```

```

WHILE @@FETCH_STATUS=0
BEGIN
IF @dataType in ('varchar','char','nchar','nvarchar')
BEGIN
--SET @stringData=@stringData+'''+isnull('+@col
SET
@stringData=@stringData+'''+'''+isnull(''+'+'''+'+
END
ELSE
if @dataType in ('text','ntext') --if the datatype is
BEGIN
SET @stringData=@stringData+'''+isnull(cast('+@
varchar(2000)),''')+''','''+

```

```

END
ELSE
IF @dataType = 'money' --because money doesn't get con
implicitly
BEGIN
    SET @stringData=@stringData+''convert(money, '''''+i
as varchar(200)), '0.0000')+'''''), ''+'
END
ELSE
IF @dataType='datetime'
BEGIN
    --SET
    @stringData=@stringData+''convert(datetime, '''''+isn
varchar(200)), '0')+'''''), ''+'
    --SELECT 'INSERT Authorizations(StatusDate)
VALUES('+convert(datetime, '+isnull(''+'+convert(varch
FROM Authorizations
    --SET @stringData=@stringData+''convert(money, '''''+
as varchar(200)), '0.0000')+'''''), ''+'
    SET
    @stringData=@stringData+''convert(datetime, '+'''+isnu
    --
    'convert(datetime, '+isnull(''+'+convert(varchar(200), S
FROM Authorizations
END
ELSE
IF @dataType='image'
BEGIN
    SET
    @stringData=@stringData+'''''''''+isnull(cast(convert(
as varchar(6)), '0')+''''', ''+'
END
ELSE --presuming the data type is int,bit,numeric,deci
BEGIN
    --SET @stringData=@stringData+'''''''''+isnull(cast('
varchar(200)), '0')+''''', ''+'
    --SET
    @stringData=@stringData+''convert(datetime, '+'''+isnu
    SET
    @stringData=@stringData+'''+'''+isnull(''+'+'''+co
END

SET @string=@string+@colName+', '

```

```

FETCH NEXT FROM cursCol INTO @tableName , @colName,@da
END
DECLARE @Query nvarchar(4000)

SET @query ='SELECT '''+substring(@string,0,len(@stringData)-2)+'''+''''
exec sp_executesql @query
--select @query

CLOSE cursCol
DEALLOCATE cursCol

/*

use poc
go

DECLARE @RC int
DECLARE @domain_user varchar(50)
DECLARE @tableName varchar(100)

-- TODO: Set parameter values here.
set @domain_user='yorgeorg'
set @tableName = 'tbGui_WizardTabButtonAreas'

EXECUTE @RC = [POC].[dbo].[procUtils_InsertGenerator]
    @domain_user
    ,@tableName

*/
GO

```

Share Improve this answer

edited Jun 6, 2016 at 10:29

Follow

answered Jun 13, 2009 at 19:19



Yordan Georgiev

5,420 ● 2 ● 59 ● 56



3



[Show create table in classic asp](#) (handles constraints, primary keys, copying the table structure and/or data ...)

Sql server Show create table Mysql-style "Show create table" and "show create database" commands from Microsoft sql server. The script is written in Microsoft asp-language and is quite easy to port to another language.*

Share Improve this answer

answered Jan 14, 2011 at 3:53

Follow



[viljun](#)

370 ● 3 ● 12



3



I include definitions for computed columns

```

select 'CREATE TABLE [' + so.name + '] (' + o.list
tc.Constraint_Name IS NULL THEN ' ELSE 'ALTER TABLE '
CONSTRAINT ' + tc.Constraint_Name + ' PRIMARY KEY ' +
Len(j.List)-1) + ')' END, name
from sysobjects so
cross apply
(SELECT

case when comps.definition is not null then ' ['+colu
comps.definition
else
    ' ['+column_name+'] ' + data_type +
case
when data_type like '%text' or data_type in ('
, 'xml')
then ''
when data_type in ('float')
then '(' + cast(coalesce(numeric_precision
')
```

```

        when data_type in ('datetime2', 'datetimeoffset')
        then '(' + cast(coalesce(datetime_precision, 0) as varchar(11))
        ')'
        when data_type in ('decimal', 'numeric')
        then '(' + cast(coalesce(numeric_precision, 0) as varchar(11))
        + ',' + cast(coalesce(numeric_scale, 0) as varchar(11))
        when (data_type like '%binary' or data_type like '%character')
        and character_maximum_length = -1
        then '(max)'
        when character_maximum_length is not null
        then '(' + cast(character_maximum_length as varchar(11))
        else ''
        end + ' ' +
        case when exists (
            select id from syscolumns
            where object_name(id)=so.name
            and name=column_name
            and columnproperty(id,name,'IsIdentity') = 1
        ) then
            'IDENTITY(' +
            cast(ident_seed(so.name) as varchar) + ',' +
            cast(ident_incr(so.name) as varchar) + ')'
        else ''
        end + ' ' +
        (case when information_schema.columns.IS_NULLABLE = 'NO'
        else 'NULL' end ) + ' ' +
        case when information_schema.columns.COLUMN_DEFAULT is not null
        then 'DEFAULT ' + information_schema.columns.COLUMN_DEFAULT
        else ''
        end + ', '

from information_schema.columns
left join sys.computed_columns comps
on OBJECT_ID(information_schema.columns.TABLE_NAME)=comps.object_id
and information_schema.columns.COLUMN_NAME=comps.name

where table_name = so.name
order by ordinal_position
FOR XML PATH('')) o (list)
left join
information_schema.table_constraints tc
on tc.Table_name = so.Name
AND tc.Constraint_Type = 'PRIMARY KEY'
cross apply
(select '[' + Column_Name + ']', '

```



```
FROM    information_schema.key_column_usage kcu
WHERE    kcu.Constraint_Name = tc.Constraint_Name
ORDER BY
        ORDINAL_POSITION
FOR XML PATH('')) j (list)
where    xtype = 'U'
AND name NOT IN ('dtproperties')
```

Share Improve this answer

answered Jan 24, 2017 at 17:31

Follow



[Erick Lanford Xenos](#)

1,562 ● 2 ● 23 ● 37



2

I realise that it's been a very long time but thought I'd add anyway. If you just want the table, and not the create table statement you could use



```
select into x from db.schema.y where 1=0
```



to copy the table to a new DB



Share Improve this answer

answered Oct 6, 2017 at 8:09

Follow



[Stu](#)

2,436 ● 2 ● 26 ● 44

One note on this method is that it won't copy extended information like constraints or default values but it's still a simple solution for a straight table copy. – [Stu](#) Jan 27, 2018 at 9:57



A query based on Hubbitus answer.

1



- includes schema names
- fixes foreign keys with more than one field
- includes CASCADE UPDATE & DELETE
- includes a conditioned DROP TABLE

```
SELECT
    Schema_Name = SCHEMA_NAME(obj.uid)
    , Table_Name = name
    , Drop_Table = 'IF (EXISTS (SELECT * FROM INFORMATION_
TABLE_SCHEMA = ''' + SCHEMA_NAME(obj.uid) + ''' AND
obj.name + '''))
DROP TABLE [' + SCHEMA_NAME(obj.uid) + '].[' + obj.nam
    , Create_Table = '
CREATE TABLE [' + SCHEMA_NAME(obj.uid) + '].[' + obj.n
LEFT(cols.list, LEN(cols.list) - 1 ) + ']' + ISNULL('
FROM sysobjects obj
CROSS APPLY (
    SELECT
        CHAR(10)
        + ' [' + column_name + ']'
        + data_type
        + CASE data_type
            WHEN 'sql_variant' THEN ''
            WHEN 'text' THEN ''
            WHEN 'ntext' THEN ''
            WHEN 'xml' THEN ''
            WHEN 'decimal' THEN '(' + CAST(numeric
', ' + CAST(numeric_scale as VARCHAR) + ')'
            ELSE COALESCE('(' + CASE WHEN characte
THEN 'MAX' ELSE CAST(character_maximum_length as VARCH
        END
        + ' '
        + case when exists ( -- Identity skip
            select id from syscolu
            where id = obj.id
            and name = column_name
            and columnproperty(id,
            ) then
                'IDENTITY(' +
                cast(ident_seed(obj.name)
```

```

                                cast(ident_incr(obj.name)
else ''
end + ' '
+ CASE WHEN IS_NULLABLE = 'No' THEN 'NOT '
+ 'NULL'
+ CASE WHEN IC.column_default IS NOT NULL
IC.column_default ELSE '' END
+ ','
FROM INFORMATION_SCHEMA.COLUMNS IC
WHERE IC.table_name = obj.name
AND IC.TABLE_SCHEMA = SCHEMA_NAME(obj.uid)
ORDER BY ordinal_position
FOR XML PATH('')
) cols (list)
CROSS APPLY(
SELECT
CHAR(10) + 'ALTER TABLE [' + SCHEMA_NAME(o
obj.name + '] ADD ' + LEFT(alt, LEN(alt)-1)
FROM(
SELECT
CHAR(10)
+ ' CONSTRAINT ' + tc.constraint_name
+ ' ' + tc.constraint_type + ' (' + LE
+ '))'
+ COALESCE(CHAR(10) + r.list, ', ')
FROM information_schema.table_constraints
CROSS APPLY(
SELECT '[' + kcu.column_name + '
FROM information_schema.key_co
WHERE kcu.constraint_name = tc.
ORDER BY kcu.ordinal_position
FOR XML PATH('')
) c (list)
OUTER APPLY(
-- // http://stackoverflow.com/que
server-howto-get-foreign-key-reference-from-informatio
SELECT LEFT(f.list, LEN(f.list)-1)
IIF(rc.DELETE_RULE = 'NO ACTION', '', ' ON DELETE ' +
IIF(rc.UPDATE_RULE = 'NO ACTION', '', ' ON UPDATE ' +
FROM information_schema.referentia
CROSS APPLY(
SELECT IIF(kcu.ordinal_positio
kcu.table_schema + '].[' + kcu.table_name + ']' ('', ''
+ '[' + kcu.column_nam

```

```

FROM information_schema.key_co
WHERE kcu.constraint_catalog =
rc.unique_constraint_catalog AND kcu.constraint_schema
rc.unique_constraint_schema AND kcu.constraint_name =
ORDER BY kcu.ordinal_position
FOR XML PATH('')
) f (list)
WHERE rc.constraint_catalog = tc.c
AND rc.constraint_schema = tc.c
AND rc.constraint_name = tc.c
) r (list)
WHERE tc.table_name = obj.name
FOR XML PATH('')
) a (alt)
) refs (list)
WHERE xtype = 'U'

```

To combine drop table (if exists) with create use like this:

```

SELECT Drop_Table + CHAR(10) + Create_Table FROM SysCr

```

Share Improve this answer

edited Sep 13, 2020 at 3:40

Follow

answered Sep 13, 2020 at 3:26



AMieres

5,004 ● 1 ● 15 ● 20



0



If you are using management studio and have the query analyzer window open you can drag the table name to the query analyzer window and ... bingo! you get the table script. I've not tried this in SQL2008

Share Improve this answer

answered Sep 23, 2008 at 8:26



Follow



RJ



That's a neat shortcut, but again: I needed to do it programmatically. – [Blorgbeard](#) Sep 23, 2008 at 9:41



Highly active question. Earn 10 reputation (not counting the [association bonus](#)) in order to answer this question. The reputation requirement helps protect this question from spam and non-answer activity.