Shortcut	A = a	Snippet Text
BT	True	BEGIN TRAN {C} ROLLBACK
C	True	/* {C} */
CA	True	CROSS APPLY
CASE	True	CASE WHEN THEN ELSE END AS
CDD	True	COLLATE database_default
CHKJOB	True	use msdb EXECUTE sp_help_jobhistory @run_status = 0
CJ	True	CROSS JOIN
CLG	True	COLLATE Latin1_General_CI_AI
CONTEXTINFO	True	DECLARE @ContextInfo VARBINARY(128) SELECT @ContextInfo = CAST(N'shuard_SQL' AS VARBINARY(128)) SET CONTEXT_INFO @ContextInfo

COPAR	True	SELECT *
		
		FROM OPENQUERY(BD_WEB_PROD,
		        'select *
		    from uda_db.drupal_choix_copar_reer
		WHERE matricule NOT IN ("128645")
		 ')

CR	True	CREATE
CRCHK	True	ALTER TABLE ... ADD CONSTRAINT CHK_ CHECK
CRCI	True	CREATE CLUSTERED INDEX CI_ ON
CRNCI	True	CREATE NONCLUSTERED INDEX CI_ ON
CRT	True	CREATE TABLE ( )
CRTI	True	CREATE TABLE (ID INT IDENTITY(1,1),)
CRTT	True	CREATE TABLE #temp ( )
CRTU	True	CREATE TABLE (ID UNIQUEIDENTIFIER,)
CRTV	True	DECLARE @tbl TABLE ( )
CRUNQ	True	CREATE UNIQUE INDEX UNQ_ ON
		DECLARE _cursor CURSOR FOR SELECT * FROM OPEN _cursor FETCH NEXT FROM _cursor; INTO @

```
                        WHILE @@FETCH_STATUS = 0
                        BEGIN
```

CUR	True	```FETCH NEXT FROM _cursor``` ```INTO @```

```
                        END
                        CLOSE _cursor;
                        DEALLOCATE _cursor;
```

D	True	`DELETE`
DE	True	`DECLARE`
DEP	True	`EXECUTE sp_depends ''`
DI	True	`DISTINCT(`
DT	True	` IF OBJECT_ID(N'',N'U') IS NOT NULL` `DROP TABLE`
E	True	`EXECUTE`
ESH	True	`EXECUTE sp_help`

ESQL	True	`DECLARE @sql NVARCHAR(MAX) = ' '`
		
		 `EXECUTE sp_executesql @sql;`

EX	True	`EXISTS`
F	True	`FROM`
FJ	True	`FULL JOIN`
FOJ	True	`FULL OUTER JOIN`
GB	True	`GROUP BY`
H	True	`HAVING`
HA	True	`HAVING`

HH	True	`EXECUTE SP_HELP  {C}`
		
		 `-- sp_helpdb`

I	True	`INSERT`
IJ	True	`INNER JOIN`
IN	True	`INSERT`
INS	True	`INSERT INTO <>()` `VALUES (<>)`
J	True	`JOIN`
JO	True	`JOIN`

```
                        USE master
                        GO

                        declare @bd_name varchar(50) = ''

                        IF OBJECT_ID(N'#TmpWho') IS NOT NULL
                        BEGIN
                        DROP TABLE #TmpWho
                        END

                        CREATE TABLE #TmpWho (SPID INT
                         , ecid INT
                         , status VARCHAR(150)
                         , loginame VARCHAR(150)
                         , hostname VARCHAR(150)
                         , blk INT
                         , dbname VARCHAR(150)
                         , cmd VARCHAR(150)
                         , request_id INT)
```

| KI | True | ```
INSERT INTO #TmpWho
EXEC sp_who

DECLARE @spid INT
DECLARE @tString VARCHAR(15)
DECLARE @getspid CURSOR

SET @getspid =   CURSOR FOR

SELECT SPID
FROM #TmpWho
WHERE dbname =  @bd_name

OPEN @getspid

FETCH NEXT FROM @getspid INTO @spid
WHILE @@FETCH_STATUS = 0
BEGIN
SET @tString = 'KILL ' + CAST(@spid AS VARCHAR(5))

EXEC(@tString)
SELECT @tString

FETCH NEXT FROM @getspid INTO @spid
END

CLOSE @getspid
DEALLOCATE @getspid
DROP TABLE #TmpWho
GO
``` |
| KI2 | True | ```
USE [master];

declare @bd_name varchar(50) = ''

DECLARE @kill varchar(8000) = '';
SELECT @kill = @kill + 'kill ' + CONVERT(varchar(5), session_id) + ';'
FROM sys.dm_exec_sessions
WHERE database_id  = db_id(@bd_name)

EXEC(@kill);
``` |
LJ	True	LEFT JOIN
NEX	True	NOT EXISTS
OA	True	OUTER APPLY
OB	True	ORDER BY
| OpenRow | True | ```
/*

OPENROWSET
( { 'provider_name' , { 'datasource' ; 'user_id' ; 'password'
  | 'provider_string' }
 , {   [ catalog. ] [ schema. ] object
    | 'query'
   }
 | BULK 'data_file' ,
    { FORMATFILE = 'format_file_path' [  ]
    | SINGLE_BLOB | SINGLE_CLOB | SINGLE_NCLOB }
} )

 ::=
  [ , CODEPAGE = { 'ACP' | 'OEM' | 'RAW' | 'code_page' } ]
  [ , DATASOURCE = 'data_source_name' ]
  [ , ERRORFILE = 'file_name' ]
  [ , ERRORFILE_DATASOURCE = 'data_source_name' ]
  [ , FIRSTROW = first_row ]
  [ , LASTROW = last_row ]
  [ , MAXERRORS = maximum_errors ]
  [ , ROWS_PER_BATCH = rows_per_batch ]
  [ , ORDER ( { column [ ASC | DESC ] } [ ,...n ] ) [ UNIQUE ] ]

  -- bulk_options related to input file format
  [ , FORMAT = 'CSV' ]
  [ , FIELDQUOTE = 'quote_characters']
  [ , FORMATFILE = 'format_file_path' ]

 */
``` |

```
SELECT *
FROM OPENROWSET(
    'Microsoft.ACE.OLEDB.12.0',
    'Excel 12.0;
 HDR=YES;
 Database=c:\temp\',    -- nom fichier
    'select * from [Feuil1$]')  -- nom feuille + $
```

RECOMP	True	`EXEC sp_recompile N'';`

REG	True	`--#region {C}`
		
		`--#endregion`

RJ	True	`RIGHT JOIN`
RN	True	`ROW_NUMBER() OVER(PARTITION BY <> ORDER BY <> )`
S	True	`SELECT`
S*	True	`SELECT *`

SAT	True	`SELECT  -- COUNT(*)`
		`*`
		`FROM   UDA_iAcces.dbo.tblAcces AS a`
		
		`WHERE a.EstModifie = 1`

SDEG	True	`EXECUTE dbo.sp_Sondage_SDEG '{C}.xml'`
SDEG1	True	`EXECUTE dbo.sp_Sondage_SDEG '{C}.xml', NULL, 1`
SDF	True	`SET DATEFORMAT YMD`
SFO	True	`SELECT *` `FROM OPENQUERY (, ' ' )`

SIP	True	`SELECT S.[name] + N'.' + O.[name] AS ObjName`
		`    ,SM.[definition]`
		
		`FROM sys.sql_modules AS SM`
		
		`INNER JOIN  sys.objects AS O`
		`ON SM.[object_id] = O.[object_id]`
		
		`INNER JOIN  sys.schemas AS S`
		`ON O.[schema_id] = S.[schema_id]`
		
		`WHERE SM.[definition] LIKE '%%'`
		
		`ORDER BY ObjName`

SIZE	True	`WITH fs`
		`AS`
		`(`
		`    SELECT database_id, TYPE, SIZE * 8.0 / 1024 SIZE`
		`    FROM sys.master_files`
		`)`
		`SELECT`
		`    name as DbName,`
		`    (SELECT SUM`
		`(SIZE) FROM fs WHERE TYPE = 0 AND fs.database_id = db.database_id) DataFileSizeMB,`
		`    (SELECT SUM`
		`(SIZE) FROM fs WHERE TYPE = 1 AND fs.database_id = db.database_id) LogFileSizeMB`
		`FROM sys.databases db`

SPACE	True	`EXEC sp_spaceused N''`
SSC	True	`SELECT  COUNT(*)` `FROM`
SSF	True	`SELECT  *` `FROM`

SSFA	True	`SELECT  *`
		`FROM   UDA_Membership.dbo.tblIntervenants as i`
		
		`INNER JOIN UDA_Membership.dbo.tblArtistes as artiste`
		`on i.IntervenantID = artiste.ArtisteID`
		
		`CROSS APPLY Axiant_Vs_Sigma.dbo.fn_categEtatTypeByMat(i.NumMatricule ) as cet`

```
{C}
```

SSFI	True	`SELECT   *`
		`FROM     UDA_Membership.dbo.tblIntervenants as i`
		
		`LEFT JOIN UDA_Membership.dbo.tblArtistes as artiste`
		`on i.IntervenantID = artiste.ArtisteID`
		
		`CROSS APPLY Axiant_Vs_Sigma.dbo.fn_categEtatTypeByMat(i.NumMatricule ) as cet`
		
		`{C}`

SSFS	True	`SELECT  i.IntervenantID, i.nom, i.Prenom`
		`,aSig.NumIPN`
		`,statut.StatutSIGARTID, statut.StatutSIGART`
		
		`FROM     SIGART.dbo.tblIntervenants AS i`
		
		`INNER JOIN SIGART.dbo.tblArtistesSIGART AS aSig`
		`ON i.IntervenantID = aSig.ArtisteID`
		`and aSig.Groupe = 0`
		
		`INNER JOIN SIGART.dbo.tblStatutsSIGARTDesc AS statut`
		`ON aSig.StatutSIGARTID = statut.StatutSIGARTID`
		`AND statut.LangueID = 1`
		`AND statut.StatutSIGARTID IN (1,3,5)`

SSI	True	`SELECT *`
		`INTO`
		`FROM`

SSP	True	`SELECT ROUTINE_NAME, ROUTINE_DEFINITION`
		`FROM INFORMATION_SCHEMA.ROUTINES`
		`WHERE ROUTINE_DEFINITION LIKE '%%'`
		`AND ROUTINE_TYPE='PROCEDURE'`

SSS	True	`SELECT c.name, * FROM sys.tables AS t`
		`INNER JOIN  sys.columns AS c ON c.object_id = t.object_id`
		`WHERE c.name LIKE '%%'`

SST	True	`SELECT  TOP 10 *` `FROM`
STATS	True	`EXECUTE sp_updatestats`

SYNC	True	`USE UDA_SYNC;`
		
		`SELECT  top 10 *`
		`FROM    [dbo].[tblLogs]`
		`ORDER BY  id desc`
		
		`SELECT  TOP 10 *`
		`FROM    [dbo].[tblJobsLog]`
		`ORDER BY  jobID desc`
		
		`SELECT  top 10 *`
		`FROM    [dbo].[tblQueue]`
		`ORDER BY  QueueID desc`

TT	True	`TRUNCATE TABLE`
U	True	`UPDATE`
UP	True	`UPDATE`

UPD	True	`UPDATE <>`
		`SET    <>`
		`FROM   <>`

VERIF	True	`RESTORE VERIFYONLY FROM DISK = '\\uda-veeam\c$\BackupSQL\'` `GO`
W	True	`WHERE`

WEBD	True	`SELECT  *`
		
		`FROM OPENQUERY(BD_WEB_DEV,`
		`        'SELECT *`
		` FROM  devuda_db.`
		` ')`

```
SELECT  *
```

WEBP	True	`FROM OPENQUERY(BD_WEB_PROD,`
		`       'SELECT *`
		`FROM  uda_db.`
		`')`

WTF	True	`WTF???!??!   AAAAAARRRRRRRRRRRGGGGGGGGGHHHHHHHHHHHHHHHHHHHHH!!!!!!!!!!!!!!!!!!!`