1. Excel folder in D drive
2. ProductMapExcelTemplate folder in D drive and Template.xlsx file in same folder

sp\_configure 'show advanced options', 1

reconfigure

GO

sp\_configure 'Ad Hoc Distributed Queries', 1

reconfigure

GO

2)

install the **AccessDatabaseEngine\_x64.exe** from microsoft:  
  
[**http://www.microsoft.com/en-us/download/details.aspx?id=13255**](http://www.microsoft.com/en-us/download/details.aspx?id=13255)  
  
make sure you open an Administrative command prompt window, and run it with the c:\Downloads\AccessDatabaseEngine\_x64.exe **/passive**  
command line flag;  
  
this will force the install of the drivers, even if you have 32 bit office installed;  
otherwise you get some error about 32 bit Office preventing the install.  
After that is isntalled:

3)

--Required settings for the provider to work correctly as a linked server

EXEC master.dbo.sp\_MSset\_oledb\_prop N'Microsoft.ACE.OLEDB.12.0', N'AllowInProcess', 1

GO

EXEC master.dbo.sp\_MSset\_oledb\_prop N'Microsoft.ACE.OLEDB.12.0', N'DynamicParameters', 1

4)

1. How to write openrowset query

SELECT \* FROM OPENROWSET('Microsoft.ACE.OLEDB.12.0',

'Excel 12.0;HDR=NO;Database=D:\Filename.xlsx;',

[SheetName$])

5) and then the code for the linked server:

--#################################################################################################

--Linked server Syntax for Excel with ACE 64 driver

--#################################################################################################

GO

DECLARE

@srvproduct nvarchar(256),

@provider nvarchar(256),

@location nvarchar(100),

@provstr nvarchar(100),

@catalog sysname,

@sql varchar(1000)

--add an excel spreadsheet as a linked server.

SET @server = 'MyExcelACE'

SET @srvproduct = ''

SET @provider = 'Microsoft.ACE.OLEDB.12.0'

set @provstr = 'Excel 12.0'

SET @datasrc ='C:\Data\BlockGroups\_2010\AKblockgroup.xls'

EXEC sp\_addlinkedserver @server,@srvproduct,@provider,@datasrc,NULL,@provstr

EXEC dbo.sp\_AddLinkedSrvLogin @server, FALSE, NULL, Admin, NULL

--what spreadsheets (table equivilents are available?

EXEC sp\_tables\_ex 'MyExcelACE'

--you MUST know the name of the spreadsheet;

--spreadsheet name has a dollar sign at the end of it!

--I've personally never gor a spreadsheet that has a space in it

--for example "Activity Data" =ActivityData$ never got [Activity Data$] or [Activity Data]$ to work

--to work, so I end up editing the spreadsheet to remove spaces if that happens.

SELECT \* FROM MyExcelACE...ActivityData$;

SELECT \* FROM MyExcelACE...Sheet1$;

5)

Alter procedure [dbo].[Save\_Area\_SP]

(

@Submoduleid int,

@Areaename as nvarchar(max) ,

@selectedModule as nvarchar(max),

@selectedsubmodule as nvarchar(max)

)

AS

Begin

--DECLARE @cmd sysname ='copy D:\ProductMapExcelTemplate\Template.xlsx D:\Excelfolder\'+@selectedModule +'\'+ @selectedsubmodule + '\'+ @Areaename +'\' +'.xlsx'

DECLARE @cmd sysname ='copy D:\ProductMapExcelTemplate\Template.xlsx D:\Excelfolder\'+@selectedModule +'\'+ @selectedsubmodule + '\'+ @Areaename +'.xlsx'

INSERT INTO [tblArea](Subrootid,Areaname)VALUES(@Submoduleid,@Areaename+'.xlsx')

EXEC master..xp\_cmdshell @cmd

END

select \* from [tblArea]