SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

IF EXISTS (SELECT \* FROM sys.objects WHERE object\_id = OBJECT\_ID(N'[Tools\_Custom\_eSecuritel\_Store\_Load]') AND type in (N'P', N'PC'))

DROP PROCEDURE [Tools\_Custom\_eSecuritel\_Store\_Load]

GO

CREATE PROCEDURE [Tools\_Custom\_eSecuritel\_Store\_Load] (

@storesCsv varchar(4000))

AS

begin

declare

@orgId int,

@midba1Id int,

@firstOfNextMonth\_YYYYMMDD varchar(10)

set @midba1Id = 1

print 'Start Tools\_Custom\_eSecuritel\_Store\_Load'

-- Hard coded for eSecuritel

if not exists (select \* from organization where orgname = 'eSecuritel')

begin

RAISERROR ('Tenant Does Not Exist', 16, 1);

end

select @orgId = orgid from organization where orgname = 'eSecuritel'

if OBJECT\_ID('TEMPDB..#storeList') IS NOT NULL Drop table #storeList

create table #storeList (

storeNumber varchar(50),

storeName varchar(100),

storevisitrulenew int

)

declare @SqlStatement nvarchar(4000)

set @SqlStatement = N'bulk insert #storeList from ' + @storesCsv + CHAR(13) + CHAR(10) +

N'with ( firstrow = 2,' + CHAR(13) + CHAR(10) +

N'fieldterminator = ''^'',' + CHAR(13) + CHAR(10) +

N'rowterminator = ''\n''' + CHAR(13) + CHAR(10) + ')'

print @SqlStatement

execute sp\_executesql @SqlStatement

alter table #storeList ADD storeId int

alter table #storeList ADD comments varchar(255)

update #storeList

set storeNumber = rtrim(ltrim(storeNumber)),

storeName = rtrim(ltrim(storeName)),

storevisitrulenew = rtrim(ltrim(storevisitrulenew)),

storeId = null,

comments = null

where 1=1

update sl

set sl.storeId = s.storeId

from #storeList sl

join store s on s.storeNumber = sl.storeNumber

where s.orgId = @orgId and s.isactive = 1

update sl

set comments = 'storevisitrule must be between 0 and 5'

from #storeList sl

where storevisitrulenew < 0 or storevisitrulenew > 5

select storeNumber, storevisitrulenew, comments from #storeList where storeid is null

select storeNumber, storevisitrulenew, comments from #storeList where comments is not null

select @firstOfNextMonth\_YYYYMMDD =

convert( varchar(10), DATEADD(MONTH, DATEDIFF(MONTH, 0, GETDATE()) + 1, 0), 111);

-- Put first of next month there

update s

set s.storeName = sl.storeName,

s.storevisitrulenew = sl.storevisitrulenew,

s.storevisiteffectivedatenew = @firstOfNextMonth\_YYYYMMDD

from store s

join #storeList sl on s.storeid = sl.storeId

print 'End Tools\_Custom\_eSecuritel\_Store\_Load'

end