

```

ALTER PROCEDURE [dbo].[spLoadImages]
(
  @directory  VARCHAR(256),
  @files      VARCHAR(128)
)

AS

BEGIN

-- Declarations
DECLARE @cmd          VARCHAR(512)      -- DOS command string to list files
DECLARE @filename     NVARCHAR(128)    -- File name
DECLARE @filecount    INT              -- File count
DECLARE @statement    NVARCHAR(512)    -- SQL command using OPENROWSET
DECLARE @params       NVARCHAR(512)    -- Parameters for SQL command
DECLARE @table        VARCHAR(128)     -- Table containing images
DECLARE @imageid      VARCHAR(128)     -- Column containing image id
DECLARE @imagename    VARCHAR(128)     -- Column containing image name
DECLARE @imagebinary  VARCHAR(128)     -- Column containing image
DECLARE @imagecount   INT              -- Current number of images

-- Insure directory ends with \
IF SUBSTRING(@directory,LEN(@directory),1) <> '\'
    SET @directory = @directory + '\'

-- Hard-wired table name and columns
-- If you want to use different table/columns, change these settings (must also
change spSaveImages)
SET @table = 'Images'
SET @imageid = 'ImageId'
SET @imagename = 'ImageName'
SET @imagebinary = 'ImageBinary'

-- Build temporary table to list files that need to be loaded
IF Object_id('tempdb..#files2load') IS NOT NULL
    DROP TABLE #files2load

CREATE TABLE #files2load
(
  rownum INT IDENTITY,
  fname  VARCHAR(512) NULL
)

-- Populate temporary table with names of files that need to be loaded (list may
contain NULL entry)
SET @cmd = 'dir ' + Quotename(@directory, '''') + @files + ' /a:-d /b' -- file
names only, and no summary info

INSERT INTO #files2load
([fname])
EXEC [master].[dbo].[xp_cmdshell] @cmd

DELETE FROM #files2load WHERE fname IS NULL

-- Initialize image count
SELECT @imagecount = MAX(ImageId) FROM images
IF @imagecount IS NULL
    SET @imagecount = 0

```

```

-- Load files into image table (return if no files found)
SELECT @filename = ''

SELECT @filecount = COUNT(*) FROM #files2load

IF @filecount = 1 AND (SELECT fname FROM #files2load WHERE fname = 'File Not
Found') IS NOT NULL
    RETURN

SET @params = N'@filename nvarchar(128)'

WHILE @filecount > 0
    BEGIN
        SELECT @filename = fname FROM #files2load WHERE rownum = @filecount

        SET @imagecount = @imagecount + 1

        SET @statement =
            N'INSERT INTO ' + @table + '(' + @imageid + ', ' + @imagename + ', ' +
@imagebinary + ') SELECT ' + CAST(@imagecount AS varchar(128)) + ' AS ' + @imageid
+ ', '' ' +
            REPLACE(@filename, '', ' ') +
            N'' AS ' + @imagename + ', * FROM OPENROWSET(BULK N'' ' +
            REPLACE(@directory + @filename, '', ' ') + N'', SINGLE_BLOB) AS ' +
@imagebinary

        EXEC sp_executesql @statement, @params, @filename=@filename

        SET @filecount = @filecount - 1
    END

-- Drop temporary table
DROP TABLE #files2load

END

```