Crear tabla a partir de un texto separado por comas (CSV)

Por **FREDDY LEANDRO ANGARITA C.  
SqlServer MVP**   
[Perfil MVP](https://mvp.support.microsoft.com/es-es/mvp/Freddy%20Leandro%20Angarita%20Castellanos-4028407)   
  
[freddy\_angarita@hotmail.com](mailto:freddy_angarita@hotmail.com)   
<http://geeks.ms/blogs/fangarita/default.aspx>

En un artículo anterior, [[Code] Crear una tabla a partir de un texto separado por comas (CSV)](http://geeks.ms/blogs/fangarita/archive/2011/01/03/code-crear-una-tabla-a-apartir-de-un-texto-se.aspx), se presentó cómo realizar ésta tarea de la manera más simple, ahora, se complementará ésta solución presentando una alternativa que usa CTEs para realizar el trabajo recursivo de analizar las palabras en la sentencia

Usando la capacidad de las expresiones CTEs de crear iteraciones se permite la creación de la iteración consigo mismo, es decir en la primera parte, se pone la semilla y en la segunda parte (luego de union all) se implementa la iteración, la cual devuelve las posiciones inicial y final del texto al que se le va a aplicar substring:

* 1. CREATE FUNCTION dbo.Split (@sep char(1), @s varchar(512))
  2. RETURNS table
  3. AS
  4. RETURN (
  5. WITH Pieces(pn, start, stop) AS (
  6. SELECT 1, 1, CHARINDEX(@sep, @s)
  7. UNION ALL
  8. SELECT pn + 1, stop + 1, CHARINDEX(@sep, @s, stop + 1)
  9. FROM Pieces
  10. WHERE stop > 0
  11. )
  12. SELECT pn,
  13. SUBSTRING(@s, start, CASE WHEN stop > 0 THEN stop-start ELSE 512 END) AS s
  14. FROM Pieces
  15. )

La solución anterior presenta una forma interesante de aproximarse al problema pero aún usa recursión, esto se puede eliminar usando XML como base de análisis

* 1. CREATE FUNCTION [dbo].[Split] (@sep VARCHAR(32), @s VARCHAR(MAX))
  3. RETURNS @t TABLE
  4. (
  5. val VARCHAR(MAX)
  6. )
  7. AS
  8. BEGIN
  9. DECLARE @xml XML
  10. SET @XML = N'' + REPLACE(@s, @sep, '') + ''
  12. INSERT INTO @t(val)
  13. SELECT r.value('.','VARCHAR(255)') as Item
  14. FROM @xml.nodes('//root/r') AS RECORDS(r)
  16. RETURN
  17. END

El truco consiste en convertir el texto, en un XML y luego usando el tipo de datos especial (y sus funciones) leemos la información como una tabla desde XML, el problema con esta solución es que tiene problemas de desempeño y se presenta solo como referencia

También se presenta uno de mis trucos favoritos el uso de la función [dm\_fts\_parser](http://msdn.microsoft.com/es-co/library/cc280463.aspx), la cual permite realizar éste análisis rápidamente, con la ventaja de:

Ser una función del sistema,

Permitir configurar el idioma

Ser sensible al acentro y mayúsculas

Poder usar una lista de palabras a ignorar

Más funcionalidades

* 1. Select display\_term from sys.dm\_fts\_parser('"' + 'Mi texto separado por espacios' + '"', 1033, 0,0)

Es ejemplo está para idioma inglés con el código 1033, para un listado completo se puede consultar el listado [LCID](http://msdn.microsoft.com/en-us/library/microsoft.sharepoint.splocale.lcid.aspx)

Espero sea de ayuda,

**FREDY LEANDRO ANGARITA CASTELLANOS  
SQL Server MVP**