
Subject: Re: splitting strings

Posted by [Benjamin Hornberger](#) on Fri, 04 Feb 2005 20:06:35 GMT

[View Forum Message](#) <> [Reply to Message](#)

Benjamin Hornberger wrote:

> Hi all,
>
> I would like to split a string by whitespace characters, while anything
> between quotes should be recognized as one elements (even if it contains
> whitespace). Let's say I have the string
>
> 'cat dog "ground hog" bird'
>
> I want to split it into ['cat', 'dog', 'ground hog', 'bird'].

Ok, here we go. Found on <http://php.net/split> and translated to IDL with some modifications. Any comments are welcome.

```
;+
; NAME:
;   QUOTESPLIT
;
;
; PURPOSE:
;   This function splits a scalar string into an array of substrings,
;   similar to STRSPLIT(). However, substrings enclosed in single or
;   double quotes are not split. Delimiters can be specified.
;
;
; AUTHOR:
;   Benjamin Hornberger
;   benjamin.hornberger@stonybrook.edu
;
;
; CATEGORY:
;   Utilities, string processing
;
;
; CALLING SEQUENCE:
;   Result = QUOTESPLIT(String [, Delimiters])
;
;
; RETURN VALUE:
;   A string array holding all the substrings of the String
;   argument. Each substring is trimmed of leading or trailing blanks
;   unless the blanks are within quotes.
;
;
;
```

```

; INPUTS:
;   String: String to be split. Must be a scalar string or one
;           element string array.
;
;
; OPTIONAL INPUTS:
;   Delimiters: A string holding all characters which serve as
;               delimiters. Only single characters can be delimiters. The
;               String argument will be split on each occurrence of one or more
;               delimiters except within quotes (single or double). If a
;               string array is given, it will be joined by STRJOIN()
;               internally. Example: if Delimiters is " ,;", the string will
;               be split on each occurrence of a space, comma or semicolon, or
;               any combination thereof (except within quotes). Default:
;               Comma, Semicolon, Space, Tab, Carriage Return and Line Feed.
;
;
; SIDE EFFECTS:
;   If the Delimiters argument is passed as string array, it will be
;   joined by QUOTESPLIT.
;
;
; EXAMPLES:
;   (quotes in the output show space characters)
;
;   IDL> petstring = 'cat dog "ground hog" "bird"'
;   IDL> pets = QUOTESPLIT(petstring)
;   IDL> FOR i = 0, n_elements(pets)-1 DO print, pets[i]
;   'cat'
;   'dog'
;   'ground hog'
;   'bird'
;
;   IDL> petstring = " cat , dog ; 'ground hog' : ' bird ' ,"
;   IDL> pets = QUOTESPLIT(petstring, " ,;:")
;   IDL> FOR i = 0, n_elements(pets)-1 DO print, pets[i]
;   'cat'
;   'dog'
;   'ground hog'
;   ' bird '
;
;
; MODIFICATION HISTORY:
;   Written: BH 2005-02-04, translated to IDL with some modifications
;   from http://php.net/split (User Contributed Note from "moritz").
;
;-

```

FUNCTION quotesplit, string, delimiters

on_error, 2

IF n_params() EQ 0 THEN \$

message, 'STRING argument required in function QUOTESPLIT'

IF n_elements(string) NE 1 THEN \$

message, 'STRING argument must be scalar in function QUOTESPLIT'

count = 0 ;; walk through all characters in string

length = strlen(string)

;; check delimiters

IF n_elements(delimiters) EQ 0 THEN \$

delimiters = ';;'+string(9B)+string(10B)+string(13B)

IF n_elements(delimiters) GT 1 THEN delimiters = strjoin(delimiters)

WHILE count LT length DO BEGIN

;; pass over all delimiters

WHILE (count LT length && strpos(delimiters, strmid(string,
count, 1)) NE -1) DO count++

;; double quotes

IF strmid(string, count, 1) EQ '"' THEN BEGIN

count++

start = count

WHILE (count LT length && strmid(string, count, 1) NE '"') DO
count++

IF n_elements(array) EQ 0 THEN \$

array = [strmid(string, start, count-start)] ELSE \$

array = [array, strmid(string, start, count-start)]

count += 2 ;; jump over 2nd quote

ENDIF ELSE IF strmid(string, count, 1) EQ "'" THEN BEGIN ;;

single quotes

count++

start = count

WHILE (count LT length && strmid(string, count, 1) NE '"') DO
count++

IF n_elements(array) EQ 0 THEN \$

array = [strmid(string, start, count-start)] ELSE \$

array = [array, strmid(string, start, count-start)]

count += 2 ;; jump over 2nd quote

ENDIF ELSE BEGIN ;; all other characters

start = count

WHILE (count LT length && strpos(delimiters, strmid(string,
count, 1)) EQ -1) DO count++

IF count GT start THEN \$

IF n_elements(array) EQ 0 THEN \$

array = [strmid(string, start, count-start)] ELSE \$

array = [array, strmid(string, start, count-start)]

count++

```
ENDELSE  
ENDWHILE
```

```
;; array could still be undefined here if string had only delimiters  
;; in it  
IF size(array, /type) EQ 0 THEN array = "
```

```
return, array
```

```
END
```
