

---

Subject: removing substring from string array  
Posted by [rchughes](#) on Mon, 02 Apr 2007 21:21:42 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi All,

Does anyone know a quick way to remove the same prefix from all elements in a string array?

Here is an example:

```
start with string_array = ['dog.spot', 'dog.rover', 'dog.lucky']
```

```
and get ['spot', 'rover', 'lucky']
```

The only way I know I can do this is with a FOR loop and the STRSPLIT(string\_array[i], '.', /EXTRACT) command. I am hoping someone could help me out with a much more efficient way if there is one.

Thanks in advance,  
Ryan.

---

---

Subject: Re: removing substring from string array  
Posted by [JD Smith](#) on Mon, 02 Apr 2007 21:38:43 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

On Mon, 02 Apr 2007 14:21:42 -0700, rchughes wrote:

> Hi All,

>

> Does anyone know a quick way to remove the same prefix from all  
> elements in a string array?

>

> Here is an example:

```
> start with string_array = ['dog.spot', 'dog.rover', 'dog.lucky']
```

>

```
> and get ['spot', 'rover', 'lucky']
```

>

> The only way I know I can do this is with a FOR loop and the  
> STRSPLIT(string\_array[i], '.', /EXTRACT) command. I am hoping someone  
> could help me out with a much more efficient way if there is one.

For a fixed length string like this, it's easy:

```
print, strmid(string_array, 4)
```

But, suppose it's a non-uniform length string:

```
string_array = ['dog.spot','cat.betsy','doggy.rover','parrot.bill','dog.luc ky']
p=strpos(string_array, '.')
print, strmid(string_array, 1#p+1)
```

Notice that silly use of 1#p. This is to ensure that only 1 sub-string is extracted from each string in the array (try it without the 1# and you'll see what I mean; see STRMID docs).

Or you could even be more general and use a regular expression:

```
string_array = ['dog.spot','cat*betsy','doggy#rover','parrot&bill','dog @lucky']
p=stregex(string_array, '^.[*#&@]', LENGTH=len)
print, strmid(string_array, 1#len)
```

JD

---

---

Subject: Re: removing substring from string array  
Posted by [Jean H.](#) on Mon, 02 Apr 2007 21:40:55 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

rchughes@gmail.com wrote:

```
> Hi All,
>
> Does anyone know a quick way to remove the same prefix from all
> elements in a string array?
>
> Here is an example:
> start with string_array = ['dog.spot', 'dog.rover', 'dog.lucky']
>
> and get ['spot', 'rover', 'lucky']
>
> The only way I know I can do this is with a FOR loop and the
> STRSPLIT(string_array[i], '.', /EXTRACT) command. I am hoping someone
> could help me out with a much more efficient way if there is one.
>
> Thanks in advance,
> Ryan.
>
```

Hi,

If you know that every element in the array starts with the prefix, you can do something like

```
prefixLength = strlen(prefix)
stringWithoutPrefix = strmid(stringArray, prefixLength)
```

If some entries may NOT start with the prefix, you can extract the

prefix, verify it and if it is the good one, remove it

```
prefix_array = strmid(stringArray, 0, prefixLength)
entriesWithThePrefix = where(prefix_array eq rebin(prefix,
n_elements(stringArray))
```

```
stringWithoutPrefix = strmid(stringArray[entriesWithThePrefix],
prefixLength)
```

You might want to do something to keep the entries that do not start with the prefix

Hope that helps,  
Jean

---

Subject: Re: removing substring from string array  
Posted by [Jean H.](#) on Mon, 02 Apr 2007 21:44:16 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

```
> entriesWithThePrefix = where(prefix_array eq rebin(prefix,
> n_elements(stringArray))
```

Please read entriesWithThePrefix = where(prefix\_array eq prefix)

---

Subject: Re: removing substring from string array  
Posted by [yp](#) on Mon, 02 Apr 2007 21:54:53 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

On Apr 2, 10:21 pm, rchug...@gmail.com wrote:

```
> Hi All,
>
> Does anyone know a quick way to remove the same prefix from all
> elements in a string array?
>
> Here is an example:
> start with string_array = ['dog.spot', 'dog.rover', 'dog.lucky']
>
> and get ['spot', 'rover', 'lucky']
>
> The only way I know I can do this is with a FOR loop and the
> STRSPLIT(string_array[i], '.', /EXTRACT) command. I am hoping someone
> could help me out with a much more efficient way if there is one.
>
> Thanks in advance,
```

> Ryan.

```
string_array = ['dog.spot', 'dog.rover', 'dog.lucky']
tmp=stregex(string_array,'dog.',length=L)
n = 100 ;set max number of characters to be extracted
new_array=transpose((strmid(string_array, L, n))[0,*])
```

Be cautious when you have a huge string\_array...

---

---

Subject: Re: removing substring from string array  
Posted by [Ryan](#). on Tue, 03 Apr 2007 13:21:11 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

> print,strmid(string\_array,4)

My current need is only for the same prefix so the above line works  
but if that changes I will come back and look at all the responses =)

Thanks everyone for all your help.

Ryan.

---

---

Subject: Re: removing substring from string array  
Posted by [Ryan](#). on Tue, 03 Apr 2007 13:50:06 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

But on second thought, I will use the following to be more robust:

```
string_array =  
['dog.spot','cat.betsy','doggy.rover','parrot.bill','dog.luc ky']  
p=strpos(string_array, '.')  
print,strmid(string_array,1#p+1)
```

Ryan.

---

---

Subject: Re: removing substring from string array  
Posted by [Ryan](#). on Tue, 03 Apr 2007 14:51:24 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

> Hooray! But, be careful. You don't want to over-design  
> the thing. :-)

I like the second method because it also applies to strings that do not have the prefix. Also, I am only using one prefix but the data that we have has many different prefixes so it will be easier to apply if it already in place.

Ryan.

---

---

Subject: Re: removing substring from string array  
Posted by [David Fanning](#) on Tue, 03 Apr 2007 15:28:29 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Ryan. writes:

```
> But on second thought, I will use the following to be more robust:  
>  
> string_array =  
> ['dog.spot','cat.betsy','doggy.rover','parrot.bill','dog.luc ky']  
> p=strpos(string_array, '.')  
> print, strmid(string_array, 1#p+1)
```

Hooray! But, be careful. You don't want to over-design the thing. :-)

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Sepore ma de ni thui. ("Perhaps thou speakest truth.")

---