How to Trim a string in PowerShell

lazyadmin.nl/powershell/powershell-trim

November 28, 2023

Do you need to remove the trailing spaces of a string? Or a character for the beginning or the end of a string in PowerShell? Then the Trim function in PowerShell is what you are looking for.

With the Trim method in PowerShell, we can remove unwanted characters or spaces from the beginning or end of a string. This allows you to easily clean up a string when needed.

In this article, we are going to take a look at how to use the Trim method and I will give you some useful examples.
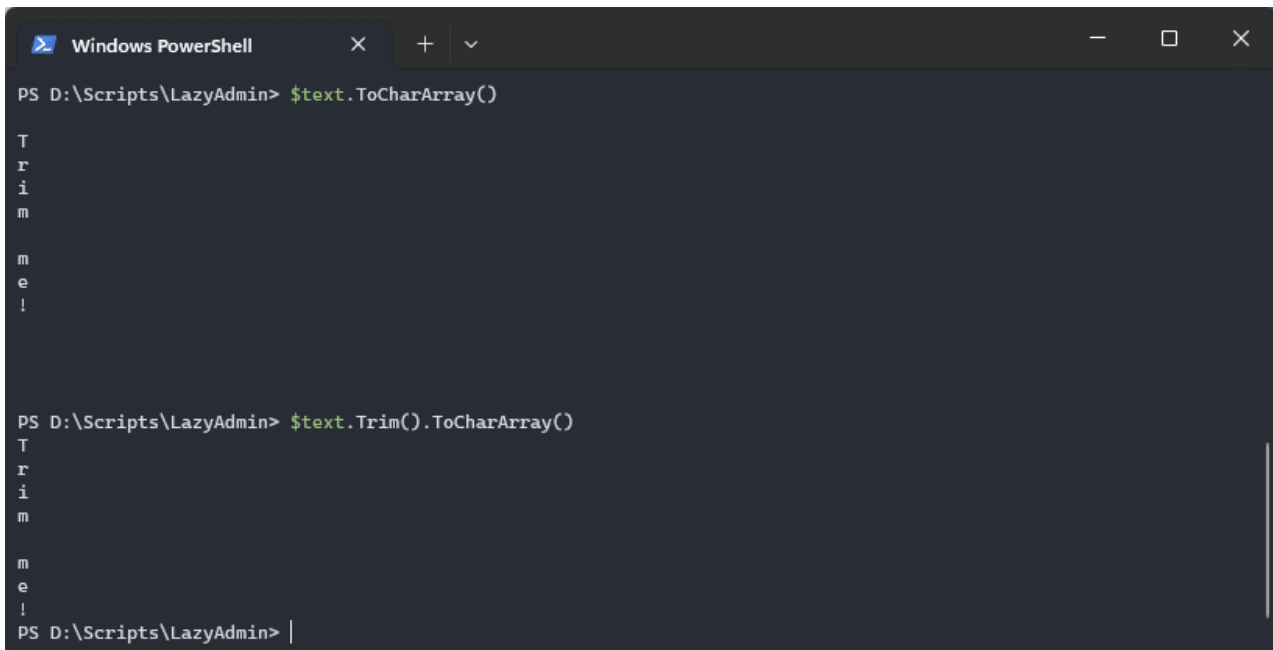
PowerShell Trim a String

In PowerShell, we can use the `Trim()` method on any string or variable that is a string. If we just use the Trim method without any characters, then it will remove the whitespace from both the beginning and the end of a string.

Let's take a quick look. In the example below, we have a string that has a space at the beginning of it, and multiple spaces at the end. When we use the Trim method on the string, all the spaces will be removed:

```
$text = " Trim me! "
Write-Host $text.Trim()
# Result
Trim me!
```

You can also see the difference in the screenshot below. We have the string, stored in the `$text` variable. By using the `ToCharArray()` method, we can clearly see each character of the string, including the spaces. But when we use the `Trim()` method first, the spaces are gone:

```
PS D:\Scripts\LazyAdmin> $text.ToCharArray()

T
r
i
m

m
e
!



PS D:\Scripts\LazyAdmin> $text.Trim().ToCharArray()
T
r
i
m

m
e
!
PS D:\Scripts\LazyAdmin> |
```

Removing Specific Characters

We can also use the Trim method in Powershell to remove other unwanted characters from the beginning or end of the string. To do this, we will need to specify the character(s) that we want to remove.

Let's start with a simple example, we are going to remove the # from the beginning of the string:

$string = "###La-srv-dc01"
Write-Host $string.Trim('#')
Result
La-srv-dc01
As you can see in the result, all the hashtags from the beginning of the string are removed, even though there are three of them. Now you might think that we can also use Trim to move the hyphen - from the string, but that doesn't work.

To remove or replace character(s) in the middle of the string, you will need to use the PowerShell Replace method.

This won't work:
$string = "La-srv-dc01"
Write-Host $string.Trim('-')
Result
La-srv-dc01
We can also define multiple characters that need to be removed from the beginning or end of a string. For example, if you want to remove !, ? or a . from the end of the string, you can use an array of characters that you want to remove:

$string = "Is this a string?"
Write-Host $string.Trim('!', '?', '.')

```
# Result
Is this a string
```

Removing complete Words

Even though we can't remove characters from the middle of the string, it's possible to remove complete words from the beginning or end of the string with the Trim method in PowerShell. We can simply specify the word that we want to remove.

```
$string = "2 days"
Write-Host $string.Trim(' days')
# Result
2
```

Trim only the Beginning or End

If you only need to remove a character from either the beginning or the end of a string, then you can also use the `TimStart()` or `TrimEnd()` methods. These methods work exactly the same as Trim(), but they only remove the character or whitespace from one side of the string.

Take the example below, we have a path to a folder but want to remove the `\` from the beginning of the path or the one from the end of the path:

```
$string = "\share\budgets\"
Write-Host $string.TrimEnd('\')
# Result
\share\budgets
# Or remove from the beginning
Write-Host $string.TrimStart('\')
# Result
share\budgets\
```

Wrapping Up

If you need to remove whitespace or other characters from either the beginning or end of a string, then the PowerShell Trim method is a great option. But when also need to remove characters from other parts of the string, then you can better use the replace method.

When important data is from a CSV fil, for example, it's always a good idea to run the strings through a trim method. This way you can be sure that the string doesn't contain any trailing spaces.

I hope you found this article helpful, if you have any questions, just drop a comment below.

Did you **Liked** this **Article**?
Get the latest articles like this **in your mailbox**
or share this article

I hate spam to, so you can unsubscribe at any time.