

# How-to: Assersions in PowerShell

In some cases it can be useful to *assert* certain prerequisite conditions.

For example, you might want to ensure that a given folder exists, and use code like this:

```
# path to download files to
$OutPath = "$env:temp\SampleData"

# does it already exist?
$exists = Test-Path -Path $OutPath -PathType Container

# if not, create it
if (!$exists)
{
    $null = New-Item -Path $OutPath -ItemType Directory
}
```

Writing code like this every time it is needed, gets repetitive.

Here is a library function (actually a **filter**) to ensure that a given folder exists:

```
filter Assert-FolderExists
{
    $exists = Test-Path -Path $_ -PathType Container
    if (!$exists) {
        Write-Warning "$_ did not exist. Folder created."
        $null = New-Item -Path $_ -ItemType Directory
    }
}
```

With this filter available, the code becomes much cleaner.

The examples below make sure (*Assert*) that the folders exist and can also assign the folder paths to variables:

```
# make sure a bunch of folders exist
'C:\test1', 'C:\test2' | Assert-FolderExists

# make sure the path assigned to a variable exists
($Path = 'c:\test3') | Assert-FolderExists
```

Code samples above via [powershell.one](https://powershell.one) (CC by 4.0).

*“Remind yourself that all men assert that wisdom is the greatest good, but that there are few who strenuously seek out that greatest good” ~ Pythagoras*

### **Related PowerShell Cmdlets:**

**Functions and Filters** - Write a named block of code.

**Operators** - Grouping Expressions ( ), Format strings \$ ( ), and arrays @ ( )