



### ***Windows PowerShell Get-Help on Cmdlet 'Describe'***

***PS:\>Get-HELP Describe -Full***

#### **NAME**

Describe

#### **SYNOPSIS**

Creates a logical group of tests. All Mocks and TestDrive contents defined within a Describe block are scoped to that Describe; they will no longer be present when the Describe block exits. A Describe block may contain any number of Context and It blocks.

#### **SYNTAX**

Describe [-Name] <String> [-Tags <Object>] [[-Fixture] <ScriptBlock>] [<CommonParameters>]

#### **DESCRIPTION**

#### **PARAMETERS**

-Name <String>

The name of the test group. This is often an expressive phrase describing the scenario being tested.

Required? true  
Position? 1  
Default value  
Accept pipeline input? false  
Accept wildcard characters? false

#### -Tags <Object>

Optional parameter containing an array of strings. When calling Invoke-Pester, it is possible to specify a -Tag parameter which will only execute Describe blocks containing the same Tag.

Required? false  
Position? named  
Default value @()  
Accept pipeline input? false  
Accept wildcard characters? false

#### -Fixture <ScriptBlock>

The actual test script. If you are following the AAA pattern (Arrange-Act-Assert), this typically holds the arrange and act sections. The Asserts will also lie in this block but are typically nested each in its own It block. Assertions are typically performed by the Should command within the It blocks.

Required? false  
Position? 2  
Default value \$(Throw "No test script block is provided. (Have you put the open curly brace on the next line?)")  
Accept pipeline input? false  
Accept wildcard characters? false

#### <CommonParameters>

This cmdlet supports the common parameters: Verbose, Debug, ErrorAction, ErrorVariable, WarningAction, WarningVariable,

OutBuffer, PipelineVariable, and OutVariable. For more information, see  
about\_CommonParameters (<https://go.microsoft.com/fwlink/?LinkID=113216>).

## INPUTS

## OUTPUTS

----- EXAMPLE 1 -----

```
PS C:\>function Add-Numbers($a, $b) {
```

```
    return $a + $b
```

```
}
```

```
Describe "Add-Numbers" {
```

```
    It "adds positive numbers" {
```

```
        $sum = Add-Numbers 2 3
```

```
        $sum | Should Be 5
```

```
    }
```

```
    It "adds negative numbers" {
```

```
        $sum = Add-Numbers (-2) (-2)
```

```
        $sum | Should Be (-4)
```

```
    }
```

```
    It "adds one negative number to positive number" {
```

```
        $sum = Add-Numbers (-2) 2
```

```
        $sum | Should Be 0
```

```
    }
```

```
    It "concatenates strings if given strings" {
```

```
        $sum = Add-Numbers two three
```

```
        $sum | Should Be "twothree"
```

```
}  
}
```

## RELATED LINKS

[It](#)

[Context](#)

[Invoke-Pester](#)

[about\\_Should](#)

[about\\_Mocking](#)

[about\\_TestDrive](#)