Full credit is given to the above companies including the OS that this PDF file was generated!

PowerShell Get-Help on command 'Tee-Object'

PS C:\Users\wahid> Get-Help Tee-Object

NAME

Tee-Object

SYNOPSIS

Saves command output in a file or variable and also sends it down the pipeline.

SYNTAX

Tee-Object [-FilePath] <System.String> [-Append] [-InputObject <System.Management.Automation.PSObject>] [<CommonParameters>]

Tee-Object [-InputObject <System.Management.Automation.PSObject>] -LiteralPath <System.String> [<CommonParameters>]

Tee-Object [-InputObject <System.Management.Automation.PSObject>] -Variable <System.String> [<CommonParameters>]

DESCRIPTION

The `Tee-Object` cmdlet redirects output, that is, it sends the output of a command in two directions (like the letter T). It stores the output in a file

or variable and also sends it down the pipeline. If `Tee-Object` is the last command in the pipeline, the command output is displayed at the prompt.

PARAMETERS

-Append <System.Management.Automation.SwitchParameter>

Indicates that the cmdlet appends the output to the specified file. Without this parameter, the new content replaces any existing content in the file without warning.

This parameter was introduced in Windows PowerShell 3.0.

-FilePath <System.String>

Specifies a file that this cmdlet saves the object to Wildcard characters are permitted, but must resolve to a single file.

-InputObject <System.Management.Automation.PSObject>

Specifies the object to be saved and displayed. Enter a variable that contains the objects or type a command or expression that gets the objects. You can also pipe an object to `Tee-Object`.

When you use the InputObject parameter with `Tee-Object`, instead of piping command results to `Tee-Object`, the InputObject value is treated as a single object even if the value is a collection.

-LiteralPath <System.String>

Specifies a file that this cmdlet saves the object to. Unlike FilePath , the value of the LiteralPath parameter is used exactly as it is typed. No characters are interpreted as wildcards. If the path includes escape characters, enclose it in single quotation marks. Single quotation marks tell PowerShell not to interpret any characters as escape sequences.

-Variable <System.String>

Specifies a variable that the cmdlet saves the object to. Enter a variable name without the preceding dollar sign (`$`).

<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).

--- Example 1: Output processes to a file and to the console ---

```
Get-Process | Tee-Object -FilePath "C:\Test1\testfile2.txt"
```

```
Handles  NPM(K)    PM(K)      WS(K) VM(M)   CPU(s)    Id ProcessName
-------  ------    -----      ----- -----   ------    -- -----------
83       4     2300       4520    39     0.30    4032 00THotkey
272      6     1400       3944    34     0.06    3088 alg
81       3      804       3284    21     2.45     148 ApntEx
81       4     2008       5808    38     0.75    3684 Apoint
...
```

Example 2: Output processes to a variable and `Select-Object`

```
Get-Process notepad | Tee-Object -Variable proc | Select-Object
processname,handles
```

```
ProcessName                  Handles
-----------                  -------
notepad                         43
notepad                         37
notepad                         38
notepad                         38
```

The `Select-Object` cmdlet selects the ProcessName and Handles properties.

Note that the `$proc` variable includes the default information returned by

`Get-Process`.

------- Example 3: Output system files to two log files -------

```
Get-ChildItem -Path D: -File -System -Recurse |
  Tee-Object -FilePath "c:\test\AllSystemFiles.txt" -Append |
    Out-File c:\test\NewSystemFiles.txt
```

The command uses the `Get-ChildItem` cmdlet to do a recursive search for system files on the D: drive. A pipeline operator (`|`) sends the list to `Tee-Object`, which appends the list to the AllSystemFiles.txt file and passes the list down the pipeline to the `Out-File` cmdlet, which saves the list in the `NewSystemFiles.txt file`.

REMARKS

To see the examples, type: "get-help Tee-Object -examples".

For more information, type: "get-help Tee-Object -detailed".

For technical information, type: "get-help Tee-Object -full".

For online help, type: "get-help Tee-Object -online"