PowerShell Get-Help on command 'Update-FormatData'

PS C:\Users\wahid> Get-Help Update-FormatData

NAME

Update-FormatData

SYNOPSIS

Updates the formatting data in the current session.

SYNTAX

```
    Update-FormatData [[-AppendPath] <System.String[]>] [-PrependPath
    <System.String[]>] [-Confirm] [-WhatIf] [<CommonParameters>]
```

DESCRIPTION

The `Update-FormatData` cmdlet reloads the formatting data from formatting files into the current session. This cmdlet lets you update the formatting data without restarting PowerShell.

Without parameters, `Update-FormatData` reloads the formatting files that it loaded previously. You can use the parameters of `Update-FormatData` to add new formatting files to the session.

Formatting files are text files in XML format with the `format.ps1xml` file name extension. The formatting data in the files defines the display of Microsoft .NET Framework objects in the session.

When Windows PowerShell starts, it loads the format data from the formatting files in the PowerShell installation directory (`$pshome`) into the session. You can use `Update-FormatData` to reload the formatting data into the current session without restarting PowerShell. This is useful when you have added or changed a formatting file, but do not want to interrupt the session.

For more information about formatting files in PowerShell, see about_Format.ps1xml (../Microsoft.PowerShell.Core/About/about_Format.ps1xml.md).

PARAMETERS

-AppendPath <System.String[]>

Specifies formatting files that this cmdlet adds to the session. The files are loaded after PowerShell loads the built-in formatting files.

When formatting .NET objects,Windows PowerShell uses the first formatting definition that it finds for each .NET type. If you use the AppendPath parameter, Windows PowerShell searches the data from the built-in files before it encounters the formatting data that you are adding.

Use this parameter to add a file that formats a .NET object that is not referenced in the built-in formatting files.

-PrependPath <System.String[]>

Specifies formatting files that this cmdlet adds to the session. The files are loaded before PowerShell loads the built-in formatting files.

When formatting .NET objects, Windows PowerShell uses the first formatting definition that it finds for each .NET type. If you use the PrependPath parameter, Windows PowerShell searches the data from the files that you are adding before it encounters the formatting data from the built-in files.

Use this parameter to add a file that formats a .NET object that is also referenced in the built-in formatting files.

-Confirm <System.Management.Automation.SwitchParameter>

Prompts you for confirmation before running the cmdlet.

-WhatIf <System.Management.Automation.SwitchParameter>

Shows what would happen if the cmdlet runs. The cmdlet is not run.

<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: Reload previously loaded formatting files -----

```
Update-FormatData
```

This command reloads the formatting files that it loaded previously.

Example 2: Reload formatting files and trace and log formatting files

```
Update-FormatData -AppendPath "trace.format.ps1xml, log.format.ps1xml"
```

This command reloads the formatting files into the session, including two new files, Trace.format.ps1xml and Log.format.ps1xml.

Because the command uses the AppendPath parameter, the formatting data in the new files is loaded after the formatting data from the built-in files.

The AppendPath parameter is used because the new files contain formatting data for objects that are not referenced in the built-in files.

------- Example 3: Edit a formatting file and reload it -------

```
Update-FormatData -PrependPath "c:\test\NewFiles.format.ps1xml"

# Edit the NewFiles.format.ps1 file.

Update-FormatData
```

This example shows how to reload a formatting file after you have edited it.

The first command adds the NewFiles.format.ps1xml file to the session. It uses the PrependPath parameter because the file contains formatting data for objects that are referenced in the built-in files.

After adding the NewFiles.format.ps1xml file and testing it in these sessions, the author edits the file.

The second command uses the `Update-FormatData` cmdlet to reload the formatting files. Because the NewFiles.format.ps1xml file was previously loaded, `Update-FormatData` automatically reloads it without using parameters.

REMARKS

To see the examples, type: "get-help Update-FormatData -examples".

For more information, type: "get-help Update-FormatData -detailed".

For technical information, type: "get-help Update-FormatData -full".

For online help, type: "get-help Update-FormatData -online"