

PowerShell ABC's - Y is for Yellow



Joe Pruitt, 2009-06-02

Welcome to this addition of the PowerShell ABC's where you'll find 26 posts detailing a component of the PowerShell scripting language, one letter at a time. For today's letter of "Y", I'll discuss PowerShell's host interface and, most importantly, it's ability to specify one of my favorite colors **Yellow**.



PowerShell is really an engine that operates within a hosting application, the default being the PowerShell.exe command line application. The host serves to expose a command line and a host interface to communicate with the commands invoked by the command line.

The Cmdlets

There are several builtin CmdLets for working with the containing host.

Out-Host - Send the output to the command line.

```
Out-Host [-InputObject ] [-Paging] []
```

Get-Host - Gets an object that represents the current host program and displays PowerShell version and regional information.

```
Get-Host []
```

Read-Host - Reads a line of input from the console

```
Read-Host  
[-Caption ]  
[-DefaultOption ]  
[-ErrorMessage ]  
[-Gui]  
[-HelpTexts ]  
[-Options ]  
[-Prompt] [-Title ]  
[-ValidatePattern ]  
[]
```

```
Read-Host  
[-AsSecureString]  
[-ErrorMessage ]  
[-Gui]  
[-Title ]  
[-ValidatePattern ]  
[]
```

Write-Host - Writes customized output to a host.

```
Write-Host  
[-BackgroundColor  
{Black|DarkBlue|DarkGreen|DarkCyan|DarkRed|DarkMagenta|DarkYellow|Gray|DarkGray|Blue|Green|Cyan|Red|Magenta|Yellow|White}]  
[-ForegroundColor  
{Black|DarkBlue|DarkGreen|DarkCyan|DarkRed|DarkMagenta|DarkYellow|Gray|DarkGray|Blue|Green|Cyan|Red|Magenta|Yellow|White}]  
[-NoNewline]  
[-Separator ]  
[[[-Object] ]  
[]
```

These read and write Cmdlets are just wrappers around the built-in host object retrieved with the \$host variable or the Get-Host Cmdlet.

```
PS C:\> (Get-Host)
```

```

Name       : ConsoleHost
Version    : 2.0
Instanceld : ac35cdb0-36c9-45a9-b692-a9443e35bd70
UI         : System.Management.Automation.Internal.Host.InternalHostUserInterface
CurrentCulture : en-US
CurrentUICulture : en-US
PrivateData : Microsoft.PowerShell.ConsoleHost+ConsoleColorProxy
IsRunspacePushed : False
Runspace   : System.Management.Automation.Runspace.LocalRunspace

```

The interesting object here is the UI member of type InternalHostUserInterface most likely derived from the [PSHostUserInterface](#) interface. A call to Get-Member on that object will display the supported commands:

```

PS C:\> (Get-Host).UI | Get-Member
TypeName: System.Management.Automation.Internal.Host.InternalHostUserInterface

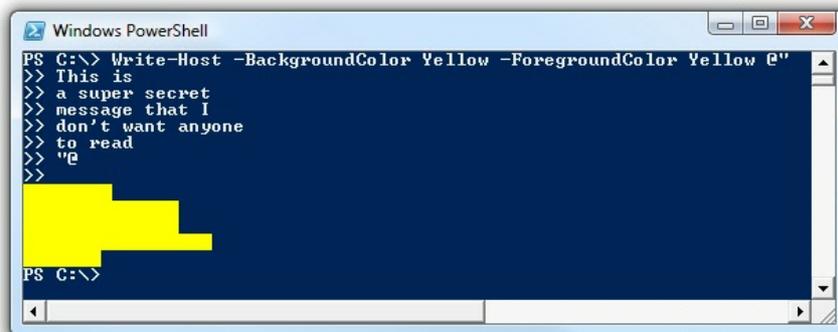
Name           MemberType Definition
-----
Equals         Method      System.Boolean Equals(Object obj)
GetHashCode    Method      System.Int32 GetHashCode()
GetType       Method      System.Type GetType()
Prompt        Method      System.Collections.Generic.Dictionary`2[System.String, mscorlib, Version=2.0.0.0,...
PromptForChoice Method      System.Int32 PromptForChoice(String caption, String message, Collection`1 choi
PromptForCredential Method      System.Management.Automation.PSCredential PromptForCredential(String capti
ReadLine      Method      System.String ReadLine()
ReadLineAsSecureString Method      System.Security.SecureString ReadLineAsSecureString()
ToString      Method      System.String ToString()
Write         Method      System.Void Write(String value), System.Void Write(ConsoleColor foregroundColor, C...
WriteDebugLine Method      System.Void WriteDebugLine(String message)
WriteErrorLine Method      System.Void WriteErrorLine(String value)
WriteLine     Method      System.Void WriteLine(), System.Void WriteLine(String value), System.Void WriteLin...
WriteProgress Method      System.Void WriteProgress(Int64 sourceId, ProgressRecord record)
WriteVerboseLine Method      System.Void WriteVerboseLine(String message)
WriteWarningLine Method      System.Void WriteWarningLine(String message)
RawUI        Property    System.Management.Automation.Host.PSHostRawUserInterface RawUI {get;}

```

You'll see some obvious relationships here to the other Cmdlets dealing with I/O such as Write-Progress, Write-Debug, Write-Warning, Write-Error and so forth.

Securing Your Output

I hinted on Yellow above in that you can control the foreground and background color of console output. Here's a practical application of using color. Consider the case where you are running a long-lived process that prints sensitive information to the console and you don't want to have others peek over your shoulder and read it. This is a perfect example of using the output color schemes. In this case, you can make the foreground and background color the same and it will look just like a color block. When you want to retrieve the output text, you can always cut and paste it from the console, but you won't be able to read it directly!



F5 Networks, Inc. | 401 Elliot Avenue West, Seattle, WA 98119 | 888-882-4447 | f5.com

F5 Networks, Inc.
Corporate Headquarters
info@f5.com

F5 Networks
Asia-Pacific
apacinfo@f5.com

F5 Networks Ltd.
Europe/Middle-East/Africa
emeainfo@f5.com

F5 Networks
Japan K.K.
f5j-info@f5.com

©2016 F5 Networks, Inc. All rights reserved. F5, F5 Networks, and the F5 logo are trademarks of F5 Networks, Inc. in the U.S. and in certain other countries. Other F5 trademarks are identified at f5.com. Any other products, services, or company names referenced herein may be trademarks of their respective owners with no endorsement or affiliation, express or implied, claimed by F5. CS04-00015 0113