My Environment (/my-lab)   VMWare, PowerCLI (/vmware)

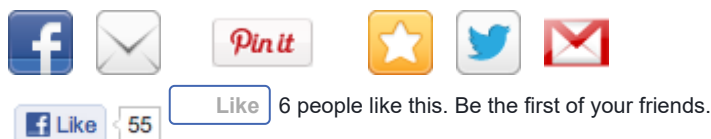Microsoft, Azure, PowerShell (/Microsoft)   About Me (/about)

Virtual Geek (/)

Tales from real IT system administrators world and non-production environment

Powershell Trick: Convert text to ASCII art (/entries/powershell/powershell-trick-convert-text-to-ascii-art)

October 4, 2017 01:40PM

(/files/images/Powershell-text-convert-to-Ascii-art-demo-abcdefghijklmnopqrstuvwxyz0123456789.png)

As PowerShell has plain command based console if you want to show different font or different bigger font on the same console as output, you will have to use ASCII Text art, I had got a request to write a script to convert any text to ASCII art. Only weirdness in the request I found was, whatever output generated as ascii art, should not be captured to file or by coping it. Someone might argue using Write-Host cmdlet, but Here I have found perfect use case for my need of using Write-Host, I have used it with backgroundcolor parameter to show font, keeping few spaces no color.

Below is the example of one of the letter A. (There are only capital alphabets, small letters are shown as capital alphabets only), You can create your own font and it is easily replaceable, I am planning to create external file in future with different fonts.

```
1 Write-Host $(" " * 6) -BackgroundColor 'Yellow' -NoNewline; Write-Host " "; Write-Host "  " -B
```

You can save script in **.\Convertto-TextASCIIArt.ps1**. There is a parameter **-Text** where your provide your custom text message. and **-FontColor** has choice (Black, DarkBlue, DarkGreen, DarkCyan, DarkRed, DarkMagenta, DarkYellow, Gray, DarkGray, Blue, Green, Cyan, Red, Magenta, Yellow, White). If you need spacial character send me comment on the blog so I can add it if required.

Difficulty running script go thru my earlier blog.
Different ways to bypass Powershell execution policy :.ps1 cannot be loaded because running scripts is disabled (http://vcloud-lab.com/entries/powershell/different-ways-to-bypass-powershell-execution-policy-ps1-cannot-be-loaded-because-running-scripts-is-disabled)
Installing, importing and using any module in powershell (http://vcloud-lab.com/entries/powershell/installing-importing-and-using-any-module-in-powershell)

If you need more font choices, you can use **-Online** parameter, which will use api and fetch selected text from http://artii.herokuapp.com using **Invoke-WebRequest** using **Get** method. For complete font list (figlet) browse http://artii.herokuapp.com/fonts_list or user **-FontName** Parameter. Font name (figlet name) is case sensitive.

(/files/images/Powershell-convertto-textasciiart-convert-text-to-ascii-art-using-online-website-api-powershell-figlet-bit-font.png)

This code is available on GitHub (https://github.com/kunaludapi/Powershell-trick-Convert-text-to-ASCII-Art) or download from here (/files/documents/Convertto-TextASCIIArt.ps1).

```
 1 #
 2 #requires -version 4
 3 <#
 4 .SYNOPSIS
 5     Convertto-TextASCIIArt converts text string to ASCII Art.
 6 .DESCRIPTION
 7     The Convertto-TextASCIIArt show normal string or text as big font nicely on console. I h
 8 .PARAMETER Text
 9     This is common parameter for inbuilt and online and incase not provided default value is
10 .PARAMETER Online
11     To use this parameter make sure you have active internet connection, as it will connect
12 .PARAMETER FontName
13     There are wide variaty of font list available on http://artii.herokuapp.com/fonts_list,
14 .PARAMETER FontColor
15     Below is the list of font color can be used to show ascii art.
16     'Black', 'DarkBlue','DarkGreen','DarkCyan', 'DarkRed','DarkMagenta','DarkYellow','Gray',
17 .PARAMETER FontHight
18     This parameter is optional and default value is 5, this is font hight and required for t
19 .INPUTS
20     [System.String]
21 .OUTPUTS
22     [console]
23 .NOTES
24     Version:        1.0
25     Author:         Kunal Udapi
26     Creation Date:  30 September 2017
27     Purpose/Change: Personal use to show text to ascii art.
28     Useful URLs: http://vcloud-lab.com, http://artii.herokuapp.com/fonts_list
29 .EXAMPLE
30     PS C:\>.\Convertto-TextASCIIArt -Online -Text "http://vcloud-lab.com" -FontColor Gray -F
31     _   _   _       _      ___     _  _         _ _       _       _     _
32    | | | | | |       _   / / /      _| |         | |     _| |     _| |
33    | |_| | |_| |_ _ _(_) / / /_  _ _| |__  _ _ _| |__ _| |__ _| |__ _ _ __ _
34    | '_ \| _| _| '_ \ / / /\ \/ / _| |/ _ \| | | |/ _` |_____| |/ _` | '_ \/ _ \_ \|
35    | | | | |_| |_| |_) | / / /  \ V / (_| | (_) | |_| | (_| |     | | (_| | |_) | (__ (_) |
36    |_| |_|\_|\_| ._(_)_/_/    \_/ \_|_|\_/ \_,_|\_,_|     |_|\_,_|.__(_)__\__/|_
37             | |
38             |_|
39
40     Shows and converts text to cool ascii art from online site http://artii.herokuapp.com us
41 .EXAMPLE
42     PS C:\>.\Convertto-TextASCIIArt -Text '# This !'
43
44
45
46       ▓▓  ▓▓  ▓▓▓▓▓ ▓▓ ▓▓ ▓▓  ▓▓▓  ▓▓
47   ▓▓▓▓▓▓▓▓  ▓▓     ▓▓ ▓▓ ▓▓ ▓▓     ▓▓
48       ▓▓  ▓▓  ▓▓    ▓▓ ▓▓ ▓▓  ▓▓▓  ▓▓
49
50     Shows local font on the script not internet required
51 #>
52
53 [CmdletBinding(SupportsShouldProcess=$True,
54     ConfirmImpact='Medium',
55     HelpURI='http://vcloud-lab.com',
56     DefaultParameterSetName='Inbuilt')]
```

```powershell
Param
(
    [parameter(Position=0, ParameterSetName='Inbuilt', ValueFromPipelineByPropertyName=$true
    [parameter(Position=0, ParameterSetName='Online', ValueFromPipelineByPropertyName=$true,
    [string]$Text = '# This is test !',
    [parameter(Position=2, ParameterSetName='Inbuilt', ValueFromPipelineByPropertyName=$true
    [Alias('Hight')]
    [string]$FontHight = '5',

    [parameter(Position=2, ParameterSetName='Online', ValueFromPipelineByPropertyName=$true,
    [ValidateSet('3-d','3x5','5lineoblique','1943____','4x4_offr','64f1____','a_zooloo','adv
    [Alias('Font')]
    [string]$FontName = 'big',

    [parameter(ParameterSetName = 'Online', Position=0, Mandatory=$false)]
    [Switch]$Online,

    [parameter(Position=1, ParameterSetName='Inbuilt', ValueFromPipelineByPropertyName=$true
    [parameter(ParameterSetName = 'Online', Position=1, Mandatory=$false, HelpMessage='Provi
    [Alias('Color')]
    [ValidateSet('Black', 'DarkBlue','DarkGreen','DarkCyan', 'DarkRed','DarkMagenta','DarkYe
    [string]$FontColor = 'Yellow'
)
Begin {
    #$NonExistFont = $null
    if ($PsCmdlet.ParameterSetName -eq 'Inbuilt') {
        $a = {<#a 07#> Write-Host $(" " * 6) -BackgroundColor $FontColor -NoNewline; Write-H
        <#a#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
        <#a#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
        <#a#> Write-Host $(" " * 6) -BackgroundColor $FontColor -NoNewline; Write-Host " "
        <#a#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
        $b = {<#b 07#> Write-Host $(" " * 4) -BackgroundColor $FontColor -NoNewline; Write-H
        <#b#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
        <#b#> Write-Host $(" " * 6) -BackgroundColor $FontColor -NoNewline; Write-Host " "
        <#b#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
        <#b#> Write-Host $(" " * 4) -BackgroundColor $FontColor -NoNewline; Write-Host $(" "
        $c = {<#c 07#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-H
        <#c#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host $(" " * 4)
        <#c#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host $(" " * 4)
        <#c#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host $(" " * 4)
        <#c#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-Host " "}
        $d = {<#d 07#> Write-Host $(" " * 4) -BackgroundColor $FontColor -NoNewline; Write-H
        <#d#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
        <#d#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
        <#d#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
        <#d#> Write-Host $(" " * 4) -BackgroundColor $FontColor -NoNewline; Write-Host $(" "
        $e = {<#e 06#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-H
        <#e#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host $(" " * 4)
        <#e#> Write-Host $(" " * 4) -BackgroundColor $FontColor -NoNewline; Write-Host "  "
        <#e#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host $(" " * 4)
        <#e#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-Host " "}
        $f = {<#f 07#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-H
        <#f#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host $(" " * 4)
        <#f#> Write-Host $(" " * 4) -BackgroundColor $FontColor -NoNewline; Write-Host "  "
        <#f#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host $(" " * 4)
        <#f#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host $(" " * 4)}
        $g = {<#g 07#> Write-Host $(" " * 6) -BackgroundColor $FontColor -NoNewline; Write-H
        <#g#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host $(" " * 5)
        <#g#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNewl
        <#g#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
```

```
117        <#g#> Write-Host $(" " * 6) -BackgroundColor $FontColor -NoNewline; Write-Host " "}
118    $h = {<#h 07#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "
119        <#h#> Write-Host "   " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
120        <#h#> Write-Host $(" " * 6) -BackgroundColor $FontColor -NoNewline; Write-Host "  "
121        <#h#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
122        <#h#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "   " -NoNew
123    $i = {<#i 03#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "
124        <#i#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  "
125        <#i#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  "
126        <#i#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  "
127        <#i#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  "}
128    $j = {<#j 07#> Write-Host "  " -NoNewline; Write-Host $(" " * 4) -BackgroundColor $F
129        <#j#> Write-Host $(" " * 4) -NoNewline; Write-Host "  " -NoNewline -BackgroundColor
130        <#j#> Write-Host $(" " * 4) -NoNewline; Write-Host "  " -NoNewline -BackgroundColor
131        <#h#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
132        <#j#> Write-Host $(" " * 6) -BackgroundColor $FontColor -NoNewline; Write-Host " "}
133    $k = {<#k 09#> Write-Host "  " -NoNewline -BackgroundColor $FontColor; Write-Host $(
134        <#k#> Write-Host "  " -NoNewline -BackgroundColor $FontColor; Write-Host " " -NoNewl
135        <#k#> Write-Host $(" " * 3) -BackgroundColor $FontColor -NoNewline; Write-Host $(" "
136        <#k#> Write-Host "  " -NoNewline -BackgroundColor $FontColor; Write-Host " " -NoNewl
137        <#k#> Write-Host "  " -NoNewline -BackgroundColor $FontColor; Write-Host $(" " * 3)
138    $l = {<#l 05#> Write-Host "   " -NoNewline -BackgroundColor $FontColor; Write-Host "
139        <#l#> Write-Host "  " -NoNewline -BackgroundColor $FontColor; Write-Host $(" " * 3)
140        <#l#> Write-Host "  " -NoNewline -BackgroundColor $FontColor; Write-Host $(" " * 3)
141        <#l#> Write-Host "  " -NoNewline -BackgroundColor $FontColor; Write-Host $(" " * 3)
142        <#l#> Write-Host "  " -NoNewline -BackgroundColor $FontColor; Write-Host "  " -Backg
143    $m = {<#m 09#> Write-Host "  " -NoNewline -BackgroundColor $FontColor; Write-Host $(
144        <#m#> Write-Host $(" " * 3) -NoNewline -BackgroundColor $FontColor; Write-Host "  "
145        <#m#> Write-Host "  " -NoNewline -BackgroundColor $FontColor; Write-Host " " -NoNewl
146        <#m#> Write-Host "  " -NoNewline -BackgroundColor $FontColor; Write-Host $(" " * 4)
147        <#m#> Write-Host "   " -NoNewline -BackgroundColor $FontColor; Write-Host $(" " * 4)
148    $n = {<#n 09#> Write-Host $(" " * 3) -NoNewline -BackgroundColor $FontColor; Write-H
149        <#n#> Write-Host "  " -NoNewline -BackgroundColor $FontColor; Write-Host " " -NoNewl
150        <#n#> Write-Host "  " -NoNewline -BackgroundColor $FontColor; Write-Host "  " -NoNew
151        <#n#> Write-Host "  " -NoNewline -BackgroundColor $FontColor; Write-Host $(" " * 3)
152        <#n#> Write-Host "  " -NoNewline -BackgroundColor $FontColor; Write-Host $(" " * 4)
153    $o = {<#o 07#> Write-Host $(" " * 6) -BackgroundColor $FontColor -NoNewline; Write-H
154        <#o#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
155        <#o#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
156        <#o#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
157        <#o#> Write-Host $(" " * 6) -BackgroundColor $FontColor -NoNewline; Write-Host " "}
158    $p = {<#p 07#> Write-Host $(" " * 6) -BackgroundColor $FontColor -NoNewline; Write-H
159        <#p#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
160        <#p#> Write-Host $(" " * 6) -BackgroundColor $FontColor -NoNewline; Write-Host "  "
161        <#p#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host $(" " * 5)
162        <#p#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host $(" " * 5)}
163    $q = {<#q 09#> Write-Host "  " -NoNewline; Write-Host $(" " * 4) -BackgroundColor $F
164        <#q#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host $(" " * 4)
165        <#q#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host $(" " * 4)
166        <#q#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
167        <#q#> Write-Host "  " -NoNewline; Write-Host "  " -BackgroundColor $FontColor -NoNew
168    $r = {<#r 07#> Write-Host $(" " * 6) -BackgroundColor $FontColor -NoNewline; Write-H
169        <#r#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
170        <#r#> Write-Host $(" " * 6) -BackgroundColor $FontColor -NoNewline; Write-Host "  "
171        <#r#> Write-Host $(" " * 4) -BackgroundColor $FontColor -NoNewline; Write-Host $(" "
172        <#r#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
173    $s = {<#s 07#> Write-Host "  " -NoNewline; Write-Host $(" " * 4) -BackgroundColor $F
174        <#s#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host $(" " * 5)
175        <#s#> Write-Host "  " -NoNewline; Write-Host "  " -NoNewline -BackgroundColor $FontC
176        <#s#> Write-Host $(" " * 4) -NoNewline; Write-Host "  " -BackgroundColor $FontColor
```

```
177          <#s#> Write-Host $(" " * 4) -BackgroundColor $FontColor -NoNewline; Write-Host "   "
178          $t = {<#t 07#> Write-Host $(" " * 6) -BackgroundColor $FontColor -NoNewline; Write-H
179          <#t#> Write-Host "  " -NoNewline; Write-Host "  " -NoNewline -BackgroundColor $FontC
180          <#t#> Write-Host "  " -NoNewline; Write-Host "  " -NoNewline -BackgroundColor $FontC
181          <#t#> Write-Host "  " -NoNewline; Write-Host "  " -NoNewline -BackgroundColor $FontC
182          <#t#> Write-Host "  " -NoNewline; Write-Host "  " -NoNewline -BackgroundColor $FontC
183          $u = {<#u 07#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "
184          <#u#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
185          <#u#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
186          <#u#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
187          <#u#> Write-Host $(" " * 6) -BackgroundColor $FontColor -NoNewline; Write-Host " "}
188          $v = {<#v 11#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host $(
189          <#v#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host $(" " * 6)
190          <#v#> Write-Host " " -NoNewline; Write-Host "  " -BackgroundColor $FontColor -NoNewl
191          <#v#> Write-Host "  " -NoNewline; Write-Host "  " -NoNewline -BackgroundColor $FontC
192          <#v#> Write-Host $(" " * 4) -NoNewline; Write-Host "  " -BackgroundColor $FontColor
193          $w = {<#W 09#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host  $
194          <#W#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host  $(" " * 4)
195          <#W#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host " " -NoNewl
196          <#W#> Write-Host $(" " * 3) -BackgroundColor $FontColor -NoNewline; Write-Host "  "
197          <#W#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host  $(" " * 4)
198          $x = {<#x 09#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host $(
199          <#x#> Write-Host " " -NoNewline ; Write-Host "  " -NoNewline -BackgroundColor $FontC
200          <#x#> Write-Host $(" " * 3) -NoNewline; Write-Host "  " -NoNewline -BackgroundColor
201          <#x#> Write-Host " " -NoNewline ; Write-Host "  " -NoNewline -BackgroundColor $FontC
202          <#x#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host $(" " * 4)
203          $y = {<#y 11#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host $(
204          <#y#> Write-Host " " -NoNewline;  Write-Host "  " -NoNewline -BackgroundColor $FontC
205          <#y#> Write-Host $(" " * 3) -NoNewline; Write-Host "  " -NoNewline -BackgroundColor
206          <#y#> Write-Host $(" " * 3) -NoNewline; Write-Host "  " -NoNewline -BackgroundColor
207          <#y#> Write-Host $(" " * 3) -NoNewline; Write-Host "  " -NoNewline -BackgroundColor
208          $z = {<#z 07#> Write-Host $(" " * 6) -BackgroundColor $FontColor -NoNewline; Write-H
209          <#z#> Write-Host $(" " * 4) -NoNewline; Write-Host "  " -NoNewline -BackgroundColor
210          <#z#> Write-Host "  " -NoNewline; Write-Host "  " -NoNewline -BackgroundColor $FontC
211          <#z#> Write-Host "  " -NoNewline -BackgroundColor $FontColor; Write-Host $(" " * 5)
212          <#z#> Write-Host $(" " * 6) -BackgroundColor $FontColor -NoNewline; Write-Host " "}
213          $hyphen = {<#- 05#> Write-Host $(" " * 5)
214          <#-#> Write-Host $(" " * 5)
215          <#-#> Write-Host $(" " * 4) -BackgroundColor $FontColor -NoNewline; Write-Host " "
216          <#-#> Write-Host $(" " * 5)
217          <#-#> Write-Host $(" " * 5)}
218          $Hash = {<## 11#> Write-Host "  " -NoNewline; Write-Host "  " -BackgroundColor $Font
219          <###> Write-Host $(" " * 10) -BackgroundColor $FontColor -NoNewline; Write-Host " "
220          <###> Write-Host "  " -NoNewline; Write-Host "  " -BackgroundColor $FontColor -NoNew
221          <###> Write-Host $(" " * 10) -BackgroundColor $FontColor -NoNewline; Write-Host " "
222          <###> Write-Host "  " -NoNewline; Write-Host "  " -BackgroundColor $FontColor -NoNew
223          $AtRate = {<#@ 09#> Write-Host " " -NosNewline; Write-Host $(" " * 6) -BackgroundCol
224          <#@#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
225          <#@#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host " " -NoNewl
226          <#@#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "  " -NoNew
227          <#@#> Write-Host " " -NoNewline; Write-Host $(" " * 4) -BackgroundColor $FontColor -
228          $Exlaim = {<#! 03#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Ho
229          <#!#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host " "
230          <#!#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host " "
231          <#!#> Write-Host $(" " * 3)
232          <#!#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host " "}
233          $Dot = {<#. 03#> Write-Host $(" " * 3)
234          <#.#> Write-Host $(" " * 3)
235          <#.#> Write-Host $(" " * 3)
236          <#.#> Write-Host $(" " * 3)
```

```
237    <#.#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host " "}
238    $Forward = {<#. 07#> Write-Host $(" " * 4) -NoNewline; Write-Host "  " -BackgroundCo
239    <#/#> Write-Host $(" " * 3) -NoNewline; Write-Host " " -BackgroundColor $FontColor
240    <#/#> Write-Host "  " -NoNewline; Write-Host " " -BackgroundColor $FontColor -NoNew
241    <#/#> Write-Host $(" " * 1) -NoNewline ; Write-Host " " -BackgroundColor $FontColor
242    <#/#> Write-Host " " -BackgroundColor $FontColor -NoNewline; Write-Host $(" " * 5)}
243    $Colun = {<#: 03#> Write-Host $(" " * 3)
244    <#:#> Write-Host " " -BackgroundColor $FontColor -NoNewline; Write-Host " "
245    <#:#> Write-Host $(" " * 3)
246    <#:#> Write-Host " " -BackgroundColor $FontColor -NoNewline; Write-Host " "
247    <#:#> Write-Host $(" " * 3)}
248    $Space = {<#  02#> Write-Host $(" " * 2)
249    <# #> Write-Host $(" " * 2)
250    <# #> Write-Host $(" " * 2)
251    <# #> Write-Host $(" " * 2)
252    <# #> Write-Host $(" " * 2)}
253    $1 = {<#1 04#> Write-Host $(" " * 3) -BackgroundColor $FontColor -NoNewline; Write-H
254    <#1 #> Write-Host " " -NoNewline; Write-Host "  " -BackgroundColor $FontColor -NoNew
255    <#1 #> Write-Host " " -NoNewline; Write-Host "  " -BackgroundColor $FontColor -NoNew
256    <#1 #> Write-Host " " -NoNewline; Write-Host "  " -BackgroundColor $FontColor -NoNew
257    <#1 #> Write-Host " " -NoNewline; Write-Host "  " -BackgroundColor $FontColor -NoNew
258    $2 = {<#2 06#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-H
259    <#2#> Write-Host $(" " * 3) -NoNewline; Write-Host " " -NoNewline -BackgroundColor
260    <#z#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-Host " "
261    <#z#> Write-Host " " -NoNewline -BackgroundColor $FontColor; Write-Host $(" " * 4)
262    <#z#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-Host " "}
263    $3 = {<#3 06#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-H
264    <#3#> Write-Host $(" " * 3) -NoNewline;  Write-Host " " -BackgroundColor $FontColor
265    <#3#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-Host " "
266    <#3#> Write-Host $(" " * 3) -NoNewline;  Write-Host " " -BackgroundColor $FontColor
267    <#3#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-Host " "}
268    $4 = {<#4 06#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host "
269    <#4#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host " " -NoNewl
270    <#4#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-Host " "
271    <#4#> Write-Host $(" " * 3) -NoNewline;  Write-Host " " -BackgroundColor $FontColor
272    <#4#> Write-Host $(" " * 3) -NoNewline;  Write-Host " " -BackgroundColor $FontColor
273    $5 = {<#5 06#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-H
274    <#s#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host $(" " * 4)
275    <#s#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-Host " "
276    <#s#> Write-Host $(" " * 3) -NoNewline; Write-Host "  " -BackgroundColor $FontColor
277    <#s#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-Host " "}
278    $6 = {<#6 06#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-H
279    <#6#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host $(" " * 4)
280    <#6#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-Host " "
281    <#6#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host " " -NoNewl
282    <#6#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-Host " "}
283    $7 = {<#7 06#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-H
284    <#7#> Write-Host $(" " * 3) -NoNewline;  Write-Host " " -BackgroundColor $FontColor
285    <#7#> Write-Host $(" " * 3) -NoNewline;  Write-Host "  " -BackgroundColor $FontColor
286    <#7#> Write-Host $(" " * 3) -NoNewline;  Write-Host "  " -BackgroundColor $FontColor
287    <#7#> Write-Host $(" " * 3) -NoNewline;  Write-Host "  " -BackgroundColor $FontColor
288    $8 = {<#8 06#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-H
289    <#8#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host " " -NoNewl
290    <#8#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-Host " "
291    <#8#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host " " -NoNewl
292    <#8#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-Host " "}
293    $9 = {<#9 06#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-H
294    <#9#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host " " -NoNewl
295    <#9#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-Host " "
296    <#9#> Write-Host $(" " * 3) -NoNewline; Write-Host "  " -BackgroundColor $FontColor
```

```powershell
        <#9#> Write-Host $(" " * 5) -BackgroundColor $FontColor -NoNewline; Write-Host " "}
        $0 = {<#0 06#> Write-Host " " -NoNewline ;Write-Host $(" " * 3) -BackgroundColor $Fo
        <#0#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host " " -NoNewl
        <#0#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host " " -NoNewl
        <#0#> Write-Host "  " -BackgroundColor $FontColor -NoNewline; Write-Host " " -NoNewl
        <#0#> Write-Host " " -NoNewline; Write-Host $(" " * 3) -BackgroundColor $FontColor -
    }#if
}
Process {
    switch ($PsCmdlet.ParameterSetName) {
        'Inbuilt' {
            [char[]]$AllCharacters = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890'
            foreach ($singleChar in $AllCharacters) {
                $CharScript = Get-Variable -Name $singleChar | Select-Object -ExpandProperty
                $CharScript = ($CharScript -split "`r`n")
                New-Variable -Name $singleChar -Value $CharScript -Force
            }
            $hyphen = $hyphen -split "`r`n"
            $Hash = $Hash -split "`r`n"
            $AtRate = $AtRate -split "`r`n"
            $Exlaim = $Exlaim -split "`r`n"
            $Dot = $Dot -split "`r`n"
            $Forward = $Forward -split "`r`n"
            $Colun = $Colun -split "`r`n"
            $Space = $Space -split "`r`n"

            $textlength = $text.Length
            [char[]]$TextBreakDown = $text

            $wordart = @()
            $FindWidth  = @()
            $NonExistFont = @()

            for ($ind= 0; $ind -lt $FontHight; $ind++) {
                $Conf = 1
                foreach ($character in $TextBreakDown) {
                    $NoFont = $True
                    Switch -regex ($character) {
                        '-' {
                            $charname =  Get-Variable -Name hyphen
                            break
                        } #-
                        '#' {
                            $charname =  Get-Variable -Name Hash
                            break
                        } ##
                        '@' {
                            $charname =  Get-Variable -Name AtRate
                            break
                        } #-
                        '!' {
                            $charname =  Get-Variable -Name Exlaim
                            break
                        } #!
                        '\.' {
                            $charname =  Get-Variable -Name Dot
                            break
                        } #.
                        ':' {
                            $charname =  Get-Variable -Name Colun
```

```powershell
                                break
                            } #.
                            '/' {
                                $charname =  Get-Variable -Name Forward
                                break
                            } #.
                            '\s' {
                                $charname =  Get-Variable -Name space
                                break
                            } #.
                            "[A-Za-z_0-9]" {
                                $charname = Get-Variable -Name $character
                                break
                            }
                            default {
                                $NoFont = $false
                                break
                            } #default
                        } #switch

                        if ($NoFont -eq $True) {
                            if ($Conf -eq $textlength) {
                                $info = $charname.value[$ind]
                                $wordart += $info
                            } #if conf
                            else {
                                $info = $charname.value[$ind]
                                $wordart += "{0} {1}" -f $info, '-NoNewLine'
                            } #else conf
                            $wordart += "`r`n"

                            #Get First Line to calculate width
                            if ($ind -eq 0) {
                                $FindWidth += $charname.value[$ind]
                            } #if ind

                            #Calculate font width
                            if ($ind -eq 0) {
                                $AllFirstLines = @()
                                $FindWidth = $FindWidth.trim() | Where-Object {$_ -ne ""}
                                $CharWidth = $FindWidth | foreach {$_.Substring(4,2)}
                                $BigFontWidth = $CharWidth | Measure-Object -Sum | Select-Object
                            } #if ind

                        } #if NoFont
                        else {
                            $NonExistFont += $character
                        } #else NoFont
                        $Conf++
                    } #foreach character
                } #for
                $TempFilePath = [System.IO.Path]::GetTempPath()
                $TempFileName = "{0}{1}" -f $TempFilePath, 'Show-BigFontOnConsole.ps1'
                $wordart | foreach {$_.trim()} | Out-File $TempFileName
                & $TempFileName
                if ($NonExistFont -ne $null) {
                    $NonExistFont = $NonExistFont | Sort-Object | Get-Unique
                    $NonResult = $NonExistFont -join " "
                    Write-Host "`n`nSkipping as, No ascii fonts found for $NonResult" -Backgroun
                } # if NonExistFont
```

```
417           } #Inbuilt
418           'Online' {
419               if ($text -eq '# This is test !') {
420                   $text = 'http://vcloud-lab.com'
421               }
422               $testEncode = [uri]::EscapeDataString($Text)
423               $url = "http://artii.herokuapp.com/make?text=$testEncode&font=$FontName"
424               Try {
425                   $WebsiteApi = Invoke-WebRequest -Uri $url -ErrorAction Stop
426                   Write-Host $WebsiteApi.Content -ForegroundColor $FontColor
427               }
428               catch {
429                   $errMessage = "Check your internet connection, Verify below url in browser`n
430                   $errMessage += $url
431                   Write-Host $errMessage -BackgroundColor DarkRed
432               }
433           } #Online
434       } #switch pscmdlet
435 }
436 end {
437 }
```

Other useful blogs

POWERSHELL FUN SEND KEYS ON THE SCREEN (http://www.vcloud-lab.com/entries/powershell/POWERSHELL-FUN-SEND-KEYS-ON-THE-SCREEN)

USE POWERSHELL ON MOBILE - SETUP AND CONFIGURE POWERSHELL WEB ACCESS (PSWA) (http://vcloud-lab.com/entries/active-directory/use-powershell-on-mobile-setup-and-powershell-web-access-pswa-)

Go Back

3 comments (/entries/powershell/powershell-trick-convert-text-to-ascii-art#comments)
Posted in Powershell (/entries/powershell)
Tags powershell (/tags/entries/powershell) powershell tricks (/tags/entries/powershell-tricks) powershell fun (/tags/entries/powershell-fun) automation (/tags/entries/automation) powershell automation (/tags/entries/powershell-automation) powershell scripts (/tags/entries/powershell-scripts)

Share ⑊ Subscribe (http://vcloud-lab.com/entries/powershell/powershell-trick-convert-text-to-ascii-art.atom)

digg Digg (http://digg.com/submit?url=http://vcloud-lab.com/entries/powershell/powershell-trick-convert-text-to-ascii-art&title=Powershell Trick: Convert text to ASCII art)

Reddit (http://reddit.com/submit?url=http://vcloud-lab.com/entries/powershell/powershell-trick-convert-text-to-ascii-art&title=Powershell Trick: Convert text to ASCII art)

Delicious (https://secure.del.icio.us/login?url=http://vcloud-lab.com/entries/powershell/powershell-trick-convert-text-to-ascii-art&title=Powershell Trick: Convert text to ASCII art)

Stumbleupon (http://www.stumbleupon.com/submit?url=http://vcloud-lab.com/entries/powershell/powershell-trick-convert-text-to-ascii-art&title=Powershell Trick: Convert text to ASCII art)

G Google bookmark (http://www.google.com/bookmarks/mark?op=edit&bkmk=http://vcloud-lab.com/entries/powershell/powershell-trick-convert-text-to-ascii-art&title=Powershell Trick: Convert text to ASCII art)

Kunal udapi (http://vcloud-lab.com)
October 5, 2017 12:27PM

1

@John, Thanks for the notifying the same. I have updated the link

@Gyz, -FontHeight default value is 5 and it represents the current font i created (There are 5 horizontal lines), I have used it for future use. This can be modified if you create your own font.

Reply

Gyz
October 5, 2017 12:40AM

2

Hi, Thanks for great script.

One issue I have is when using parameter -FontHeight with higher value than the default value of 5, it produces some exception:
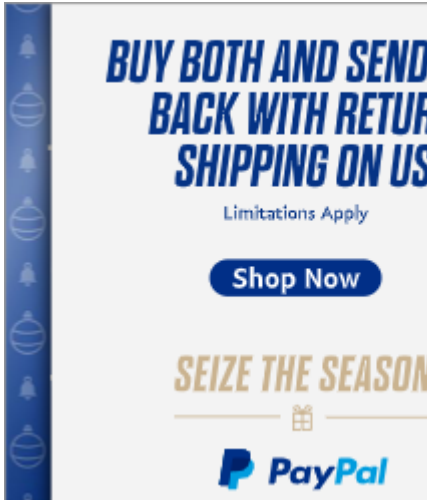
-NoNewLine : The term '-NoNewLine' is not recognized as the name of a cmdlet, function, script file, or operable program. Check the spelling of the name, or if a path was included, verify that the path is correct and try again.
At C:\Temp\Show-BigFontOnConsole.ps1:45 char:1
+ -NoNewLine
+ ~~~~~~~~~~
+ CategoryInfo : ObjectNotFound: (-NoNewLine:String) [], CommandNotFoundException
+ FullyQualifiedErrorId : CommandNotFoundException

Reply

John
October 4, 2017 06:45PM

3

Youre github link is missing a colon

:

https// is what it is currently.

Reply

Comment

Name:

BLOG SEARCH

Search

PAGE VIEWS

1919208

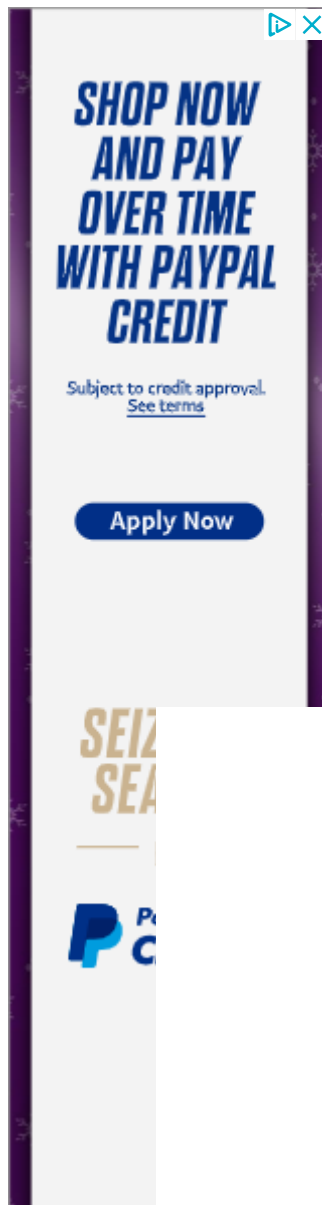SUBSCRIBE TO OUR EMAIL NEWSLETTER & RECEIVE UPDATES RIGHT IN YOUR INBOX (300+ USERS).

ARCHIVE

Show all...

COMMENTS

Syed

COOL POWERSHELL FREE ONLINE GUI GENERATOR TOOL, POSHGUI (/entries/powershell/cool-powershell-free-online-gui-generator-tool-poshgui) November 17, 2018 08:54PM

Kunal (http://vcloud-lab.com)

Powershell Active Directory: ADGroup Managedby - Checkbox Manager can update membership list (/entries/active-directory/powershell-active-directory-adgroup-managedby-checkbox-manager-can-update-membership-list) November 16, 2018 06:11PM

Nick

Reset ESXi root password using Host Profiles on vCenter server: VMWare vSphere Web client (/entries/vcenter-server/reset-esxi-root-password-using-vcenter-server-vmware-vsphere-web-client) November 10, 2018 05:01AM

Tellier

Powershell Active Directory: ADGroup Managedby - Checkbox Manager can update membership list (/entries/active-directory/powershell-active-directory-adgroup-managedby-checkbox-manager-can-update-membership-list)
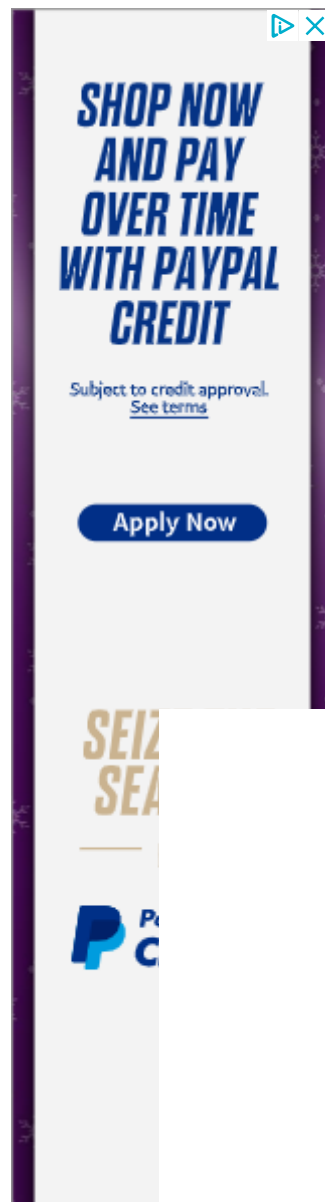November 5, 2018 09:38PM

cividan
Migrating move back from Distributed virtual switch to Standard virtual switch - VMs (Virtual Machine) - Part 3 (/entries/vcenter-server/migrating-move-back-from-distributed-virtual-switch-to-standard-virtual-switch-vms--virtual-machine-----part-3)
October 30, 2018 01:41AM

Disclaimer: All the steps and scripts shown in my posts are tested on non-production servers first. All the scripts provided on my blogs are comes without any warranty, The entire risk and impacts arising out of the use or performance of the sample scripts and documentation remains with you. Author is not liable for any damages

whatsoever arising out of the use of or inability to use the sample scripts or documentation.

Warning: Everything I say and do in these blogs or videos are subject to mistake and criticism. please do everything in your power to correct me if I saying or doing something wrong, or inform me of what I could be doing better. I am a man made out of my environment, and you are the ones creating who I am. Please don't let me fall to stupidity or ignorance, I expect the absolute best in each and every one of you and I hope you expect the same of me. Thank you.

Usage of cookies: In order to optimize the website and for continuous improvement vcloud-lab.com uses cookies. You agree to the usage of cookies when you continue using this site.