

PowerShell et la photo

Organiser ces photos avec la lecture de l'exif

L'EXIF, c'est un format de fichier utilisé pour les images prises par un appareil photographique numérique, la première version de l'Exif date de 1995. C'est une mine d'or pour organiser ces photos.

==== Effacer les dossiers vides ==== <sxh powershell> # Set to true to test the script \$whatIf = \$false # Remove hidden files, like thumbs.db \$removeHiddenFiles = \$false # Get hidden files or not. Depending on removeHiddenFiles setting \$getHiddelFiles = !\$removeHiddenFiles # Remove empty directories locally Function Delete-EmptyFolder(\$path) { # Go through each subfolder, Foreach (\$subFolder in Get-ChildItem -Force -Literal \$path -Directory) { # Call the function recursively Delete-EmptyFolder -path \$subFolder.FullName } # Get all child items \$subItems = Get-ChildItem -Force:\$getHiddelFiles -LiteralPath \$path # If there are no items, then we can delete the folder # Exluc folder: If ¹⁾)) If (\$subItems -eq \$null) { Write-Host "Removing empty folder '\$(\$path)'" Remove-Item -Force -Recurse:\$removeHiddenFiles -LiteralPath \$Path -WhatIf:\$whatIf } } # Run the script Delete-EmptyFolder -path "E:_Backup" </sxh> Lire des informations dans un fichier image <sxh powershell> Function Get-Image { begin{ [void][System.Reflection.Assembly]::LoadWithPartialName("System.Drawing") | Out-Null } process{ \$fi=[System.IO.FileInfo]\$_ if(\$fi.Exists){ \$img = [System.Drawing.Image]::FromFile(\$_) \$img.Clone() \$img.Dispose() }else{ Write-Host "File not found: \$_" -fore yellow } } end{} } Get-ChildItem -Path "F:\Deploy\Scripts\Misc\SG - SetWallpaper\Source\Wallpaper" -Filter *.jpg | ForEach-Object { \$image = \$_ | Get-Image New-Object PSObject -Property @{ File = \$_.name Fullname = \$_.FullName Height = \$image.Height Width = \$image.Width }; } </sxh> ==== Déplacer les fichiers par date vers un autre dossier ==== Je n'utilise plus cette version de script, je préfère utiliser le script avec l'utilisation des informations EXIF <sxh powershell> \$Destination = 'F:\Photos\' \$Source = 'F:\Photos\2019' Get-ChildItem -Path \$Source -File -Filter *.jpg -Recurse| ForEach-Object { \$Year = \$_.lastwritetime.year.ToString("0000") \$Month = \$_.lastwritetime.month.ToString("00") \$Day = \$_.lastwritetime.day.ToString("00") \$D = Join-Path \$Destination -ChildPath "\$Year\\$Month-\$Year\\$Day-\$Month-\$Year" if ((Test-Path \$D) -eq \$false){New-Item -Path \$D -ItemType Directory -Force | Out-Null} Move-Item -Path \$_.FullName -Destination (Join-Path -Path \$D -ChildPath \$_.name) -Force } </sxh>

¹⁾

\$subItems -eq \$null) -and (-Not(\$path.contains("DfsrPrivate"))

From:
<http://poste2travail.free.fr/dokuwiki/> - Poste2Travail

Permanent link:
<http://poste2travail.free.fr/dokuwiki/doku.php?id=script:powershell:multimedia:photo&rev=1666509626>

Last update: **2022/10/23 09:20**