# destin\_decompress\_CTR-X.ps1

**#region script global variables**

**#get the root directory to extract ".in" files**

$rootFolder = "**<pathToRootFolder\>**"

**#get the path of logs**

$logFolder = "$rootFolder\logs\"

**#build date with two different formats**

$date = Get-Date

$date = $date.ToString("yyyy-MM-dd-HH-mm-ss")

$dateHeader = Get-Date

$dateHeader = $dateHeader.ToString("yyyy/MM/dd-HH:mm")

**#build customized date fo log files**

$dateLog = Get-Date

$dateLog = $dateLog.ToString("yyyy-MM-dd")

**#endregion**

**#region select zip packages and create log folder**

$destinZipFiles = Get-ChildItem "$rootFolder\\*.zip"

foreach ($zipped in $destinZipFiles)

{

if (!(Test-Path -path $logFolder))

{

New-Item $logFolder -Type Directory

}

$zipped = Split-Path -Path $zipped -leaf

Write-Host $zipped

Start-Sleep -Seconds 1

}

**#endregion**

**#region search for ".in" files in zip packages**

$search = ".in"

$zips = "$rootFolder"

$Manifest = "$logFolder\$dateLog.log"

Write-Output "########################################################################################################################`r`n Log file created on $dateHeader `r`n########################################################################################################################`r`n" >> $Manifest

Function GetZipFileItems

{

Param([string]$zip)

$split = $split.Split(".")

$shell = New-Object -Com Shell.Application

$zipItem = $shell.NameSpace($zip)

$items = $zipItem.Items()

$split = $zipFile.Split("\")[-1]

Write-output "Contents of ${split}:"

GetZipFileItemsRecursive $items

}

Function GetZipFileItemsRecursive

{

Param([object]$items)

ForEach($item In $items)

{

$strItem = [string]$item.Name

If ($strItem -Like "\*$search\*")

{

If ((Test-Path ($zips + $strItem)) -eq $False)

{

$zipFile = Split-Path -Path $zipFile -leaf

Write-Host "Extracted file: $strItem"

$shell.NameSpace($zips).CopyHere($item)

}

Write-output "$strItem"

}

}

}

Function GetZipFiles

{

$zipFiles = Get-ChildItem -Path $zips -Filter "\*.zip" | % { $\_.DirectoryName + "\$\_" }

ForEach ($zipFile in $zipFiles)

{

$split = $zipFile.Split("\")[-1]

GetZipFileItems $zipFile

$count = GetZipFileItems $zipFile

$ZipFilesItemsCount = ($count.count)-1

Write-Output "Total: $ZipFilesItemsCount files found on $Split`r`n------------------------------------------------------------------------------------------------------------------------"

}

}

**#endregion**

**#region counting of ".in" files and outputs to the log file**

GetZipFiles >> $Manifest

$total = 0

$zipFiles = Get-ChildItem -Path $zips -Filter "\*.zip"

$Shell = New-Object -ComObject Shell.Application

$result = foreach( $ZipFile in $ZipFiles ){

$total += $Shell.NameSpace($ZipFile.FullName).Items() |

Where-Object { $\_.Name -match '\.IN$' } |

Measure-Object |

Select-Object -ExpandProperty Count

}

write-output "Files found in zip files: $total`r`n------------------------------------------------------------------------------------------------------------------------" >> $Manifest

write-host "`r`nFiles found in zip files: $total`r`n-------------------------------------------------------------------------"

$zippedFile = Split-Path $zipped -leaf

Write-Output "All .in files were extracted!`r`n------------------------------------------------------------------------------------------------------------------------" >> $Manifest

Write-host "`r`nAll .in files were extracted!`r`n-------------------------------------------------------------------------"

**#endregion**

**#region remove .zip packages after extraction and outputs to log file**

foreach ($zipped in $destinZipFiles)

{

#delete zipfile

Remove-Item "$zipped"

$zippedFile = Split-Path $zipped -leaf

Write-Output "Zip file $zippedFile was deleted!" >> $Manifest

Write-host "Zip file $zippedFile was deleted!"

}

**#endregion**