```
#Load SharePoint CSOM Assemblies
#Add-Type -Path "C:\Program Files\Common Files\Microsoft Shared\Web Server Extensions\16\ISAPI\Microsoft.SharePoint.Client.dll"
#Add-Type -Path "C:\Program Files\Common Files\Microsoft Shared\Web Server Extensions\16\ISAPI\Microsoft.SharePoint.Client.Runtime.dll"
cls

Set-Location $PSScriptRoot


$fileName = "File_Uploading_Report"
#'yyyyMMddhhmm   yyyyMMdd
$enddate = (Get-Date).tostring("yyyyMMddhhmmss")
#$filename =  $enddate + '_VMReport.doc'
$logFileName = $fileName +"_"+ $enddate+"_Log.txt"
#$invocation = (Get-Variable MyInvocation).Value
#$directoryPath = Split-Path $invocation.MyCommand.Path
$directoryPath = $PSScriptRoot

$directoryPathForLog=$directoryPath+"\"+"LogFiles"
if(!(Test-Path -path $directoryPathForLog))
        {
            New-Item -ItemType directory -Path $directoryPathForLog
            #Write-Host "Please Provide Proper Log Path" -ForegroundColor Red
        }


#$logPath = $directoryPath + "\" + $logFileName

$logPath = $directoryPathForLog + "\" + $logFileName

$isLogFileCreated = $False





#DLL location

$directoryPathForDLL=$directoryPath+"\"+"Dependency Files"
if(!(Test-Path -path $directoryPathForDLL))
        {
            New-Item -ItemType directory -Path $directoryPathForDLL
            #Write-Host "Please Provide Proper Log Path" -ForegroundColor Red
        }

#DLL location

$clientDLL=$directoryPathForDLL+"\"+"Microsoft.SharePoint.Client.dll"
$clientDLLRuntime=$directoryPathForDLL+"\"+"Microsoft.SharePoint.Client.Runtime.dll"

Add-Type -Path $clientDLL
Add-Type -Path $clientDLLRuntime


#Files to upload location

$directoryPathForFileToUploadLocation=$directoryPath+"\"+"Files To Upload"
if(!(Test-Path -path $directoryPathForFileToUploadLocation))
```

```
        {
            New-Item -ItemType directory -Path $directoryPathForFileToUploadLocation
            #Write-Host "Please Provide Proper Log Path" -ForegroundColor Red
        }

#Files to upload location ends here.



function Write-Log([string]$logMsg)
{
    if(!$isLogFileCreated){
        Write-Host "Creating Log File..."
        if(!(Test-Path -path $directoryPath))
        {
            Write-Host "Please Provide Proper Log Path" -ForegroundColor Red
        }
        else
        {
            $script:isLogFileCreated = $True
            Write-Host "Log File ($logFileName) Created..."
            [string]$logMessage = [System.String]::Format("[$(Get-Date)] - {0}", $logMsg)
            Add-Content -Path $logPath -Value $logMessage
        }
    }
    else
    {
        [string]$logMessage = [System.String]::Format("[$(Get-Date)] - {0}", $logMsg)
        Add-Content -Path $logPath -Value $logMessage
    }
}



#The below function will upload the file from local directory to SharePoint Online library.

Function FileUploadToSPOnlineLibrary()
{
    param
    (
        [Parameter(Mandatory=$true)] [string] $SPOSiteURL,
        [Parameter(Mandatory=$true)] [string] $SourceFilePath,
        [Parameter(Mandatory=$true)] [string] $File,
        [Parameter(Mandatory=$true)] [string] $TargetLibrary,
        [Parameter(Mandatory=$true)] [string] $UserName,
        [Parameter(Mandatory=$true)] [string] $Password
    )

    Try
    {

        $securePassword= $Password | ConvertTo-SecureString -AsPlainText -Force
        #Setup the Context
        $ctx = New-Object Microsoft.SharePoint.Client.ClientContext($SPOSiteURL)
        $ctx.Credentials = New-Object Microsoft.SharePoint.Client.SharePointOnlineCredentials($UserName, (Get-Content
$passwordfile | ConvertTo-SecureString))

        #$list = $ctx.Web.Lists.GetByTitle($TargetLibrary)
        #$ctx.Load($list)
        #$ctx.ExecuteQuery()

        #Get the Target Folder to upload

        $Web = $Ctx.Web
        $Ctx.Load($Web)
        $TargetFolder = $Web.GetFolderByServerRelativeUrl($TargetLibrary)
        $Ctx.Load($TargetFolder)
        $Ctx.ExecuteQuery()

        #Get Source File From Disk
        #$fileOpenStream = New-Object IO.FileStream($SourceFilePath, [System.IO.FileMode]::Open)
        $fileOpenStream = ([System.IO.FileInfo] (Get-Item $SourceFilePath)).OpenRead()
       # $tarGetFilePath=$siteURL+"/"+"$TargetLibrary"+"/"+$File
        $tarGetFilePath=$TargetLibrary+"/"+$File


        $fileCreationInfo = New-Object Microsoft.SharePoint.Client.FileCreationInformation
        $fileCreationInfo.Overwrite = $true
        $fileCreationInfo.ContentStream = $fileOpenStream
        #$fileCreationInfo.URL = "$File"
        $fileCreationInfo.URL = $tarGetFilePath
        #$uploadFileInfo = $list.RootFolder.Files.Add($FileCreationInfo)
        $uploadFileInfo = $TargetFolder.Files.Add($FileCreationInfo)
        $ctx.Load($uploadFileInfo)
        $ctx.ExecuteQuery()


        Write-host -f Green "File '$SourceFilePath' has been uploaded to '$tarGetFilePath' successfully!"
    }
    Catch
    {

            $ErrorMessage = $_.Exception.Message +"in uploading File!: " +$tarGetFilePath
            Write-Host $ErrorMessage -BackgroundColor Red
            Write-Log $ErrorMessage


    }
}

#Variables
$siteURL="https://wvuhsc.sharepoint.com/cscforms"
$PHIFolder="Shared Documents/PHI Info Reports"
$CCFolder="Shared Documents/Credit Card Info Reports"
$TaxesFolder="Shared Documents/Taxes Info Reports"
$SSNFolder="Shared Documents/SSN Info Reports"
$defaultFolder="Shared Documents"
$fromDate="2019-10-28"
$toDate="2019-11-09"
```

```powershell
$filesFolderLocation=$(Get-ChildItem "\\hs\public\ITS\security services\ID Finder Scan Files\*\Processed" | sort CreationTime | select -last 1).FullName;
$userName = "mattadmin@hsc.wvu.edu"
#Hashed Password File
$passwordfile = "$PSScriptRoot\Dependency Files\SPPassword.txt"
if (Test-Path ($passwordfile)) {
    Write-Verbose "Password File found"
}
else {
    Write-Host "Password File Not Created Cannot Connect to SP"
}
if (!(Test-Path ($passwordfile))) {

    $password = (Get-Credential).Password | ConvertFrom-SecureString | Out-File $passwordfile
}
#$securePassword= $password | ConvertTo-SecureString  -Force

#Variables ends here.


$filesCollectionInSourceDirectory=Get-ChildItem $filesFolderLocation -File

$uploadItemCount=1;

    #Extract the each file item from the folder.
    ForEach($oneFile in $filesCollectionInSourceDirectory)
    {
        $docLibFolder=""
            try
            {
                #Add Additional Folder Names here
                if ($oneFile -like "*ePHI*") {
                    $docLibFolder = $PHIFolder

                }
                elseif ($oneFile -like "*CC*") {
                    $docLibFolder = $CCFolder

                }
                elseif ($oneFile -like "*TAXES*") {
                    $docLibFolder = $TaxesFolder

                }
                elseif ($oneFile -like "*SSN*") {
                    $docLibFolder = $SSNFolder

                }

                else {
                    $docLibFolder = $defaultFolder
                }

                FileUploadToSPOnlineLibrary -SPOSiteURL $siteURL -SourceFilePath $oneFile.FullName -File $oneFile -TargetLibrary $docLibFolder -UserName $UserName -Password $(Get-Content $passwordfile | ConvertTo-SecureString)

                $fileUploadingMessage=$uploadItemCount.ToString()+": "+$oneFile.Name;
                Write-Host $fileUploadingMessage -BackgroundColor DarkGreen
                Write-Log $fileUploadingMessage

        $uploadItemCount++

        }
        catch
        {
            $ErrorMessage = $_.Exception.Message +"in: " +$oneFile.Name
            Write-Host $ErrorMessage -BackgroundColor Red
            Write-Log $ErrorMessage

        }

    }
    Write-Host "========================================================================"
    Write-Host "Total number of files uploaded: " $filesCollectionInSourceDirectory.Count
    Write-Host "========================================================================"

```