

Automation Workflow for:  
One Drive Prep Script.ps1

.NOTES  
Script Name : Get-DeletedTeamsSites.ps1  
Author : chris@brennantechnologies.com  
Date Created : 2025-12-12  
Last Modified : 2025-12-12  
This Version : 1.1

## Run the Script:

Run the script with appropriate parameters & permissions to modify users OneDrive folders.

Start

```
### Run the Script with Parameters:  
### -----  
### Example:  
.\\OneDrivePrepScript.ps1 -TenantUrl "https://tenantID.sharepoint.us"  
-LogPath "c:\\tmp\\logs"  
-DataFile "c:\\tmp\\UserData.csv"
```

## Script Parameters:

You could manually edit the script parameters to match your environment.

Set Script Parameters

```
### Script Parameters:  
### -----  
param (  
    ### Project Name/Client:  
    $Project = "Test Dev Script"  
,  
    ### Tenant URL:  
    $TenantUrl = "https://$tenantID-my.sharepoint.us"  
,  
    ### Path to the .csv file containing error log file:  
    $LogPath = ".\\Projects\\OneDrive Prep Script"  
,  
    ### Path to the .csv file containing user data:  
    $DataFile = "UserData.csv")
```

## Import User Data:

Import User .csv file containing user UPN.



Import User Date (Upn)

```
### Import User Data:  
### -----  
try {  
    $users = Import-Csv -Path $DataFile  
}  
catch {  
    Write-RtxLog -Message $_.Exception.Message -Level Error  
    throw }
```

UserData.csv

## Script Logic:

```
Foreach ($user in $users){  
    1. Connect User Site  
    2. Create Folders  
    3. Get Items  
    4. Move Files  
    5. Log & Report  
    6. Disconnect User  
}
```

## Out of scope options:

- Email User Report
- Code Sign Script

Loop through All User

Foreach (\$user in \$users)

Connect to User Site

```
### Connect to User's site:  
### -----  
### Replace '@' and '.' with '-'  
$userUpn = $($user.Urn).Replace("@", "-").Replace(".", "-")
```

```
### User's OneDrive Relative URL  
$userRelativeUrl = "$tenantUrl/personal/$userUpn"
```

```
$pnpSiteConnection = Connect-PnPOnline -Url $userRelativeUrl  
-ReturnConnection -UseWebLogin
```

Create Folders  
"DO NOT MIGRATE",  
"APPROVED TO MIGRATE"

```
### Create Folders:  
### -----  
#|("DO NOT MIGRATE", "APPROVED TO MIGRATE", "TestFolder") | ForEach-Object {  
    Resolve-PnPFolder -siteRelativePath "Documents/$_" | Out-Null }
```

Get User Items

```
### Get OneDrive Items:  
### -----  
$allItems = Get-PnPListItem -List 'documents' | Where-Object {  
    $($_.FieldValues.Title) -ne "DO NOT MIGRATE"  
    -and $($_.FieldValues.Title) -ne "APPROVED TO MIGRATE"  
    -and $($_.FieldValues.Title) -ne "TestFolder"}
```

## Move Items:

Move Item:

Move User Items

```
### Move Items:  
### -----  
$movePnPFileParams = @{  
    ServerRelativeUrl = $item.ServerRelativeUrl  
    TargetSiteRelativeLibrary = $false  
    AllowItemNameMismatch = $false  
    IgnoreVersionHistory = $false  
    Connection = $pnpSiteConnection  
    Force = $true  
    ErrorAction = 'Stop'
```

```
}  
Move-PnPFile @movePnPFileParams ## Move the item
```

## Log &amp; Report:

Log File

User Report



Loop through All User Items

Summary Log/Report/Errors

Email User Report

Out of Scope

Disconnect User

Export Script Reports

Summary Report

Transcript

Out of Scope

Email Summary Reports

End

```
.NOTES  
Script Name : Get-DeletedTeamsSites.ps1  
Author : chris@brennantechnologies.com  
Date Created : 2025-12-12  
Last Modified : 2025-12-12  
This Version : 1.1
```