

運用管理において、よく使いそうなPowerShellを纏めています。

管理者権限で使えるSharePoint Management Shell のコマンドレット  
以下のコマンドは最初に実行が必要です。

```
# SPO 管理センターに接続 nnnnn.mmm@xxxx.yyy  
# 構文 Connect-SPOService -Url [Admin Portal Url]  
Connect-SPOService -Url https://xxxxx-admin.sharepoint.com
```

### 01.取得するプロパティを絞り込む (Url/現在の使用量/使用量の上限/警告レベル)

```
# Get-SPOSite -Limit All | select Url,StorageUsageCurrent, StorageQuota, StorageQuotaWarningLevel  
Get-SPOSite -Limit All | Where-Object {$_.Url -CLike "*/sites/000*"} | select Url,StorageUsageCurrent, StorageQuota, StorageQuotaWarningLevel  
# 取得するプロパティを絞り込む (最終更新日時/Url/現在の使用量/使用量の上限)  
Get-SPOSite -Limit All | Where-Object {$_.Url -CLike "*/sites/000*"} | select LastContentModifiedDate, Url, StorageUsageCurrent, StorageQuota
```

### 02.指定したサイトの Url/現在の使用量/使用量の上限/警告レベルを取得

```
# Get-SPOSite -Identity [URL] | select Url,StorageUsageCurrent, StorageQuota, StorageQuotaWarningLevel  
Get-SPOSite -Identity https://xxxxx.sharepoint.com/sites/Study2 | select Url,StorageUsageCurrent, StorageQuota, StorageQuotaWarningLevel  
# 取得するプロパティを絞り込む (Url/全プロパティ情報)  
Get-SPOSite -Identity https://xxxxx.sharepoint.com/sites/Study2 | select *
```

### 03.サイトURLが"000"で始まるサイト一覧を取得

```
Get-SPOSite -Limit All | Where-Object {$_.Url -CLike "*/sites/000*"} | select Url, Title, LastContentModifiedDate  
# サイトURLが"000"で始まるサイトで最終更新日時「LastContentModifiedDate」が3月のサイト一覧を取得  
Get-SPOSite -Limit All | Where-Object {($_.Url -CLike "*/sites/000*") -and ($_.LastContentModifiedDate -CLike "*03/*")} | select Url, Title, LastContentModifiedDate
```

### 04.サイト種別ごとの一覧取得 URL/管理者/容量

```
# クラシックのチームサイトを列挙 Get-SPOSite -Limit All -Template STS#0  
# モダン チームサイトを列挙 Get-SPOSite -Limit All -Template STS#3  
# Microsoft 365 グループが紐づくチームサイトを列挙 Get-SPOSite -Limit All -Template GROUP#0  
# コミュニケーション サイトを列挙 Get-SPOSite -Limit All -Template SITEPAGEPUBLISHING#0
```

### 05.カスタム スクリプトの設定値を有効化

```
# 構文 : Set-SPOSite -Identity <対象サイトの URL> -DenyAddAndCustomizePages 0  
Set-SPOSite -Identity https://xxxxx.sharepoint.com/sites/study2 -DenyAddAndCustomizePages 0
```

**06.カスタムスクリプトの設定を取得**

```
# 構文 Get-SPOSite -Limit All | select Url,Title,DenyAddAndCustomizePages
Get-SPOSite -Identity https://xxxxx.sharepoint.com/sites/Study2 | select Url,Title,DenyAddAndCustomizePages
```

**07.サイト容量の上限が 指定した容量 (MB) より大きいサイトのみ列挙**

```
# 構文 Get-SPOSite -Limit All | where{$_StorageQuota -gt [MB]}
Get-SPOSite -Identity https://xxxxx.sharepoint.com/sites/Study2 | where{$_StorageQuota -gt 204800}
```

**08.指定したサイトの容量を 100GB に、警告レベルを70GB に設定する**

```
# 構文 GSet-SPOSite -Identity [URL] -StorageQuota 102400 -StorageQuotaWarningLevel 71680
Set-SPOSite -Identity https://xxxxx.sharepoint.com/PnPsites/Study2 -StorageQuota 102400 -StorageQuotaWarningLevel 71680
```

**09.グループの一覧とグループに設定されたユーザー一覧を取得**

```
$SiteUrl = "https://xxxxx.sharepoint.com/sites/Study2"
$x = Get-SPOSiteGroup -Site $SiteUrl
foreach ($y in $x) {
    Write-Host $y.Title -ForegroundColor "Yellow"
    Get-SPOUser -Site $SiteUrl -Group $y.Title |
    select displayName,LoginName,IsGroup,UserType | Format-table -Wrap -AutoSize
    Write-Host
}
}
```

**10. サイトURLが"000"で始まるサイトの SharePointグループ 所有者 に設定されたユーザー一覧を取得**

```
$xx = Get-SPOSite -Limit All | Where-Object {$_.Url -CLike "*/sites/000*"} | select Url, Title
foreach ($ss in $xx) {
  $SiteUrl = $ss.Url
  $SiteTitle = $ss.Title
  Write-Host $SiteUrl , $SiteTitle -ForegroundColor "Green"
  try{
    $x = Get-SPOSiteGroup -Site $SiteUrl | Where-Object {$_.Title -CLike "*所有者"}
    $Color = "Yellow"
    $AdminGrpName = $x.Title
  }catch{
    $Color = "Blue"
    $AdminGrpName = $SiteTitle+" 所有者"
  }
  Write-Host $AdminGrpName -ForegroundColor $Color
  Get-SPOUser -Site $SiteUrl -Group $AdminGrpName | select displayName,LoginName,IsGroup
  Write-Host
}
```

**11. グループの一覧とグループに設定されたセキュリティグループの一覧を取得**

```
$SiteUrl = "https://xxxxx.sharepoint.com/sites/Study2"
$x = Get-SPOSiteGroup -Site $SiteUrl
foreach ($y in $x) {
  Write-Host $y.Title -ForegroundColor "Yellow"
  Get-SPOUser -Site $SiteUrl -Group $y.Title | select displayName,LoginName,IsGroup |
  Where-Object { $_.IsGroup -eq "True" } | Format-table -Wrap -AutoSize
  Write-Host
}
```

## 12.モダンチームサイト内のグループの一覧とグループに設定されたセキュリティグループ名の一覧を取得

```
Write-Host セキュリティグループ 確認開始 -ForegroundColor "Yellow"
```

```
$x = Get-SPOSite -Limit All -Template STS#3 | Where-Object {$_.Url -CLike "*/sites/000*"}
```

```
$Cnt = 0
```

```
$mCnt = 0
```

```
$Null
```

```
foreach ($u in $x) {
```

```
    $Cnt = $Cnt + 1
```

```
    Write-Host "_____"
```

```
    Write-Host $Cnt $u.Url $u.Title -ForegroundColor "Yellow"
```

```
    try{
```

```
        $y = Get-SPOSiteGroup -Site $u.Url
```

```
        foreach ($z in $y) {
```

```
            try{
```

```
                Get-SPOUser -Limit All -Site $u.Url -Group $z.Title | select displayName,LoginName,IsGroup | Where-Object { $_.IsGroup -eq "True" } | Format-table -Wrap -AutoSize
```

```
$Rec = Get-SPOUser -Limit All -Site $u.Url -Group $z.Title | select displayName,LoginName,IsGroup | Where-Object { $_.IsGroup -eq "True" }
```

```
if(($Rec.displayName -ne $Null) -and ($Rec.displayName -notlike "Limit*") -and ($Rec.displayName -notlike "Sharing*")) {
```

```
    Write-Host $Cnt $z.Title -ForegroundColor "Green"
```

```
    if(($Rec.displayName -eq "SG_品質保証部GM以上") -or ($Rec.displayName -eq "SG_常勤役員") -or ($Rec.displayName -eq "SG_常務執行役員")) {
```

```
        $mCnt = $mCnt + 1
```

```
        Write-Host $Cnt [$mCnt] $Rec.displayName "登録有り" -ForegroundColor "Red"
```

```
    } else {
```

```
        if($Rec.displayName -ne $Null) {Write-Host $Cnt [$mCnt] $Rec.displayName }
```

```
    }
```

```
    }
```

```
    }catch{
```

```
        Write-Host " セキュリティG登録無し"
```

```
    }
```

```
}catch{Write-Host $Cnt "SPグループ参照エラー" -ForegroundColor "Red" }
```

```
}
```

```
Write-Host 一致したセキュリティグループ件数[$mCnt] セキュリティグループ 確認終了 -ForegroundColor "Yellow"
```

**13. 指定サイトのSharePointグループ一覧を取得**

```
$SiteUrl = "https://xxxxx.sharepoint.com/sites/Study2"  
Get-SPOSiteGroup -Site $SiteUrl | select "Title"
```

**14. 指定サイトのSharePointグループに設定されたユーザー一覧を取得**

```
Get-SPOUser -Site $SiteUrl -Group "Study2 所有者" | select * | Format-table -Wrap -AutoSize  
Get-SPOUser -Site $SiteUrl -Group "Study2 メンバー" | select * | Format-table -Wrap -AutoSize  
Get-SPOUser -Site $SiteUrl -Group "Study2 閲覧者" | select * | Format-table -Wrap -AutoSize
```

PnP 管理者権限で使えそうなSharePoint Shell ISE のコマンドレット 以下のコマンドは最初に実行が必要です。

```
### サイトに接続 yoshikazu.makino@jccu.coop
```

```
# 構文 connect = Connect-PnPOnline -Url [URL] -UseWebLogin ← 推奨 : 多要素認証用 -Interactive
```

```
Connect-PnPOnline -Url https://xxxxx.sharepoint.com/sites/Study2 -Interactive
```

### 1.任意バージョン履歴の削除

```
# Remove-PnPFileVersion -Url [URL] -Identity [VersionNo] -Force
```

```
Remove-PnPFileVersion -Url https://xxxxx.sharepoint.com/sites/Study2/S02/DocLib23 -Identity 2 -Force
```

### 2.指定バージョン履歴[\$RemainingCount]だけ残して削除 : 古い順から「バージョン数」-「残すバージョン数」を削除

```
$FileVersions = Get-PnPFileVersion -Url https://xxxxx.sharepoint.com/sites/Study2/S02/DocLib23
```

```
$FileVersions = $FileVersions | Sort @{Expression = "Created"; Descending = $false}
```

```
$RemainingCount = 2
```

```
foreach ($i in (0..($FileVersions.count - $RemainingCount - 1))) {
```

```
    Remove-PnPFileVersion -Url https://xxxxx.sharepoint.com/sites/Study2/S02/DocLib23 -Identity $FileVersions[$i].Id -Force
```

```
}
```

### 3.全バージョン履歴を削除

```
# 構文 Remove-PnPFileVersion -Url [URL] -All -Force
```

```
Remove-PnPFileVersion -Url https://xxxxx.sharepoint.com/sites/Study2/S02/DocLib23 -All -Force
```

### 4.ライブラリ・リストの情報取得

```
# Get-PnPList -Identity [Name] | Select Title, EnableVersioning, MajorVersionLimit
```

```
Get-PnPList -Identity 23.ライブラリ履歴削除検証② | Select Title, EnableVersioning, MajorVersionLimit
```

### 5.リスト・ライブラリの保存バージョン数設定

```
# Set-PnPList -Identity [Name] -MajorVersions [SetVersion]
```

```
Set-PnPList -Identity 23.ライブラリ履歴削除検証② -MajorVersions 3
```