

## Import-Module ActiveDirectory

```
function GenerateForm {
#####
# Code Generated By: SAPIEN Technologies PrimalForms (Community Edition) v1.0.9.0
# Generated On: 18/10/2018 23:27
# Generated By: Ian A Morris
# Get Group Members v3.0
#####

#region Import the Assemblies
[reflection.assembly]::loadwithpartialname("System.Drawing") | Out-Null
[reflection.assembly]::loadwithpartialname("System.Windows.Forms") | Out-Null
#endregion

#region Generated Form Objects
$form1 = New-Object System.Windows.Forms.Form
$buttDATA = New-Object System.Windows.Forms.Button
$progressBar1 = New-Object System.Windows.Forms.ProgressBar
$buttEXIT = New-Object System.Windows.Forms.Button
$buttEXPORT = New-Object System.Windows.Forms.Button
$listBox1 = New-Object System.Windows.Forms.ListBox
$dataGridView1 = New-Object System.Windows.Forms.DataGridview
$buttGETMEM = New-Object System.Windows.Forms.Button
$comboBox1 = New-Object System.Windows.Forms.ComboBox
$label1 = New-Object System.Windows.Forms.Label
$InitialFormWindowState = New-Object System.Windows.Forms.FormWindowState
#endregion Generated Form Objects

#-----
#Generated Event Script Blocks
#-----

#####
#####
function Load-ComboBox {
    <#
    .SYNOPSIS
        This functions helps you load items into a ComboBox.

    .DESCRIPTION
        Use this function to dynamically load items into the ComboBox control.

    .PARAMETER ComboBox
        The ComboBox control you want to add items to.

    .PARAMETER Items
        The object or objects you wish to load into the ComboBox's Items collection.

    .PARAMETER DisplayMember
```

Indicates the property to display for the items in this control.

.PARAMETER Append

Adds the item(s) to the ComboBox without clearing the Items collection.

#>

```
Param (  
    [Parameter(Mandatory = $true)]  
    [System.Windows.Forms.ComboBox]$ComboBox,  
    [Parameter(Mandatory = $true)]  
    $Items,  
    [Parameter(Mandatory = $false)]  
    [string]$DisplayMember,  
    [switch]$Append  
)  
  
if (-not $Append) {  
    $comboBox.Items.Clear()  
}  
  
if ($Items -is [Array]) {  
    $comboBox.Items.AddRange($Items)  
}  
else {  
    $comboBox.Items.Add($Items)  
}  
  
$comboBox.DisplayMember = $DisplayMember  
}
```

```
#####  
#####
```

```
#####  
#####
```

```
function Load-ListBox {  
    <#  
    .SYNOPSIS  
        This functions helps you load items into a ListBox or CheckedListBox.  
  
    .DESCRIPTION  
        Use this function to dynamically load items into the ListBox control.  
  
    .PARAMETER ListBox  
        The ListBox control you want to add items to.  
  
    .PARAMETER Items  
        The object or objects you wish to load into the ListBox's Items  
        collection.  
  
    .PARAMETER DisplayMember
```

Indicates the property to display for the items in this control.

**.PARAMETER Append**

Adds the item(s) to the ListBox without clearing the Items collection.

**.EXAMPLE**

```
Load-ListBox $ListBox1 "Red", "White", "Blue"
```

**.EXAMPLE**

```
Load-ListBox $listBox1 "Red" -Append  
Load-ListBox $listBox1 "White" -Append  
Load-ListBox $listBox1 "Blue" -Append
```

**.EXAMPLE**

```
Load-ListBox $listBox1 (Get-Process) "ProcessName"
```

#>

```
Param (  
    [ValidateNotNull()]  
    [Parameter(Mandatory = $true)]  
    [System.Windows.Forms.ListBox]$ListBox,  
    [ValidateNotNull()]  
    [Parameter(Mandatory = $true)]  
    $Items,  
    [Parameter(Mandatory = $false)]  
    [string]$DisplayMember,  
    [switch]$Append  
)  
  
if (-not $Append) {  
    $ListBox.Items.Clear()  
}  
  
if ($Items -is [System.Windows.Forms.ListBox+ObjectCollection]) {  
    $ListBox.Items.AddRange($Items)  
}  
elseif ($Items -is [Array]) {  
    $ListBox.BeginUpdate()  
    foreach ($obj in $Items) {  
        $ListBox.Items.Add($obj)  
    }  
    $ListBox.EndUpdate()  
}  
else {  
    $ListBox.Items.Add($Items)  
}  
  
$ListBox.DisplayMember = $DisplayMember  
}
```

```
#####  
#####
```

```
#region Report Path  
$PathOUT = $env:USERPROFILE  
#$SplitPath = "\Documents\Get Group Members\Reports\  
$ReportPath = "$PathOut\Documents\Reports\Get Group Members\  
}
```

```
If (!(Test-Path $ReportPath)) {  
    New-Item -ItemType Directory -Force -Path $ReportPath | Out-Null  
}  
#endregion Report Path
```

```
#region Housekeeping  
$limit = (Get-Date).AddDays(-7)  
$ClearPath = $ReportPath  
Get-ChildItem -Path $ClearPath -Recurse -Force | Where-Object { !$_.PSIsContainer -and  
$_.CreationTime -lt $limit } | Remove-Item -Force  
#endregion Housekeeping
```

```
$combobox1.Text = "Paste, Type or Press 'Get Group Data' to Import Groups From AD..."
```

```
#####  
#####
```

```
#Provide Custom Code for events specified in PrimalForms.  
$buttEXPORT_OnClick =  
{  
    #TODO: Place custom script here  
    $progressBar1.Visible = $False  
    $progressBar1.BackColor = [System.Drawing.Color]::FromArgb(255, 255, 255, 255)  
    $progressBar1.ForeColor = [System.Drawing.Color]::FromArgb(255, 0, 0, 255)  
    $dataGridView1.BackgroundColor = [System.Drawing.Color]::FromArgb(255, 176, 224, 230)  
    $listBox1.ForeColor = [System.Drawing.Color]::Black  
    $listBox1.BackColor = [System.Drawing.Color]::FromArgb(255, 176, 224, 230)  
    $datagridview1.DataSource = $null  
    $listBox1.Items.Clear()  
}
```

```
Load-ListBox $ListBox1 "Exporting Group Membership Information..."  
Start-Sleep -Seconds 2  
$buttDATA.Enabled = $False  
$buttEXPORT.Enabled = $False  
$buttEXIT.Enabled = $False  
$buttGETMEM.Enabled = $False
```

```
$GroupSec = $comboBox1.Text
```

```
if ($GroupSec -eq "" -or $GroupSec -eq "Paste, Type or Press 'Get Group Data' to Import Groups From AD..." -or $GroupSec -eq "Paste, Type or Select a Group...") {
```

```
    $dataGridView1.BackgroundColor = [System.Drawing.Color]::FromArgb(255, 255, 0, 0)
    $listBox1.ForeColor = [System.Drawing.Color]::White
    $listBox1.BackColor = [System.Drawing.Color]::FromArgb(255, 255, 0, 0)
    Load-ListBox $listBox1 "No Group Name Entered. Enter a Group Name or click the"
    Load-ListBox $listBox1 "Get Group Data button; to import groups from AD" -Append
```

```
    $buttDATA.Enabled = $true
    $buttEXPORT.Enabled = $true
    $buttEXIT.Enabled = $true
    $buttGETMEM.Enabled = $true
```

```
    }
    else {
```

```
        $ExportRep = "$ReportPath $GroupSec - Members Report-{0:yyyy-MM-dd_HH.mm.ss}" -f
        (get-date)
```

```
        If (Get-ADGroup -F { Name -eq $GroupSec }) {
            $Members = [System.Collections.ArrayList](((Get-ADGroup $GroupSec -Properties
members).members | Get-ADUser -Properties GivenName, Surname, DisplayName,
SamAccountName, EmailAddress, Enabled, ObjectClass | Select GivenName, Surname, DisplayName,
SamAccountName, EmailAddress, Enabled, ObjectClass | Sort-Object GivenName | Export-Csv -Path
"$ExportRep.csv" -Append -NoTypeInfo))
```

```
            $buttDATA.Enabled = $true
            $buttEXPORT.Enabled = $true
            $buttEXIT.Enabled = $true
            $buttGETMEM.Enabled = $true
            $listBox1.BackColor = [System.Drawing.Color]::FromArgb(255, 255, 255, 255)
            $dataGridView1.BackgroundColor = [System.Drawing.Color]::FromArgb(255, 255, 255, 255)
            Load-ListBox $listBox1 "Exporting Group Membership Information Complete"
            $progressBar1.BackColor = [System.Drawing.Color]::FromArgb(255, 255, 255, 255)
            $progressBar1.ForeColor = [System.Drawing.Color]::FromArgb(255, 0, 0, 255)
            $progressBar1.Visible = $true
```

```
<#####
#####
```

```
#####
#####>
```

```
Invoke-Item -Path "$ExportRep.csv"
```

```

        #Start-Process excel.exe -wait -ArgumentList "$ExportRep.csv /x" -WindowState
Maximized

    }
    Else {

        $dataGridView1.BackgroundColor = [System.Drawing.Color]::FromArgb(255, 255, 0, 0)
        $listBox1.ForeColor = [System.Drawing.Color]::White
        $listBox1.BackColor = [System.Drawing.Color]::FromArgb(255, 255, 0, 0)
        Load-ListBox -ListBox $listBox1 " WARNING : The Group -"
        Load-ListBox -ListBox $listBox1 " WARNING : >$GroupSec<" -Append
        Load-ListBox -ListBox $listBox1 " WARNING : Cannot be found." -Append
        Load-ListBox -ListBox $listBox1 " WARNING : Check details and retry." -Append
        $buttDATA.Enabled = $true
        $buttEXPORT.Enabled = $true
        $buttEXIT.Enabled = $true
        $buttGETMEM.Enabled = $true
        # $listBox1.BackColor = [System.Drawing.Color]::FromArgb(255,255,255,255)
        $progressBar1.BackColor = [System.Drawing.Color]::FromArgb(255, 255, 0, 0)
        $progressBar1.ForeColor = [System.Drawing.Color]::FromArgb(255, 255, 0, 0)
    }
}
}

$buttEXIT_OnClick =
{
    #TODO: Place custom script here
    $form1.Close()
}

$buttDATA_OnClick =
{
    #TODO: Place custom script here
    $progressBar1.BackColor = [System.Drawing.Color]::FromArgb(255, 255, 255, 255)
    $progressBar1.ForeColor = [System.Drawing.Color]::FromArgb(255, 0, 0, 255)
    $dataGridView1.BackgroundColor = [System.Drawing.Color]::FromArgb(255, 176, 224, 230)
    $listBox1.ForeColor = [System.Drawing.Color]::Black
    $listBox1.BackColor = [System.Drawing.Color]::FromArgb(255, 176, 224, 230)
    $datagridview1.DataSource = $null
    $combobox1.Items.Clear()
    $listBox1.Items.Clear()
    $progressBar1.Visible = $False
    $progressbar1.Value = 100
    $dataGridView1.BackgroundColor = [System.Drawing.Color]::FromArgb(255, 176, 224, 230)
    $listBox1.BackColor = [System.Drawing.Color]::FromArgb(255, 176, 224, 230)
    $Listbox1.Items.Clear()
    $combobox1.Text = "Gathering Group Data, Please Wait..."
    $buttDATA.Enabled = $False
    $buttEXPORT.Enabled = $False
    $buttEXIT.Enabled = $False
    $buttGETMEM.Enabled = $False
}

```

```
$datagridview1.DataSource = $null
```

```
    $DataGroups = Start-Job -ScriptBlock { Get-ADGroup -Filter * | Where-Object { $_.Name -ne "-*" } | Select-Object -ExpandProperty Name | Sort Name -Descending }  
    #Wait for the background jobs  
    $DataGroups | Wait-Job  
    #Get the data from them  
    $Groups = $DataGroups | Receive-Job  
    #And delete the jobs - just cleaning up after ourselves  
    $DataGroups | Remove-Job  
    ForEach ($Group in $Groups) { Load-ComboBox -ComboBox $combobox1 -Items $Group -  
Append }
```

```
# Load-ListBox $ListBox1 "Importing All Active Directory Groups..." -Append |
```

```
$progressBar1.Visible = $true  
$buttDATA.Enabled = $true  
$buttEXPORT.Enabled = $true  
$buttEXIT.Enabled = $true  
$buttGETMEM.Enabled = $true  
$combobox1.Text = ""  
$dataGridView1.BackgroundColor = [System.Drawing.Color]::FromArgb(255, 255, 255, 255)  
$ListBox1.BackColor = [System.Drawing.Color]::FromArgb(255, 255, 255, 255)  
Load-ListBox $ListBox1 "Completed Gathering Group Data Information From Active Directory"  
#$combobox1.Text = ""  
$combobox1.Text = "Paste, Type or Select a Group..."
```

```
}
```

```
$buttGETMEM_OnClick =
```

```
{
```

```
    #TODO: Place custom script here  
    $progressBar1.Visible = $False  
    $progressBar1.BackColor = [System.Drawing.Color]::FromArgb(255, 255, 255, 255)  
    $progressBar1.ForeColor = [System.Drawing.Color]::FromArgb(255, 0, 0, 255)  
    $dataGridView1.BackgroundColor = [System.Drawing.Color]::FromArgb(255, 176, 224, 230)  
    $ListBox1.ForeColor = [System.Drawing.Color]::Black  
    $ListBox1.BackColor = [System.Drawing.Color]::FromArgb(255, 176, 224, 230)  
    $datagridview1.DataSource = $null  
    $ListBox1.Items.Clear()  
    Load-ListBox $ListBox1 "Gathering Group Membership Information..."  
    Start-Sleep -Seconds 2  
    $buttDATA.Enabled = $False  
    $buttEXPORT.Enabled = $False  
    $buttEXIT.Enabled = $False  
    $buttGETMEM.Enabled = $False
```

```

$GroupSec = $comboBox1.Text

if ($GroupSec -eq "" -or $GroupSec -eq "Paste, Type or Press 'Get Group Data' to Import Groups
From AD..." -or $GroupSec -eq "Paste, Type or Select a Group..." ) {

    $dataGridView1.BackgroundColor = [System.Drawing.Color]::FromArgb(255, 255, 0, 0)
    $listBox1.ForeColor = [System.Drawing.Color]::White
    $listBox1.BackColor = [System.Drawing.Color]::FromArgb(255, 255, 0, 0)
    Load-ListBox $listBox1 "No Group Name Entered. Enter a Group Name or click the"
    Load-ListBox $listBox1 "Get Group Data button; to import groups from AD" -Append

    $buttDATA.Enabled = $true
    $buttEXPORT.Enabled = $true
    $buttEXIT.Enabled = $true
    $buttGETMEM.Enabled = $true

}
else {

    If (Get-ADGroup -F { Name -eq $GroupSec }) {
        $Members = [System.Collections.ArrayList]((Get-ADGroup $GroupSec -Properties
members).members | Get-ADUser -Properties GivenName, Surname, DisplayName,
SamAccountName, EmailAddress, Enabled, ObjectClass | Select GivenName, Surname, DisplayName,
SamAccountName, EmailAddress, Enabled, ObjectClass | Sort GivenName)
        $datagridview1.DataSource = $Members

        $progressBar1.Visible = $true
        $buttDATA.Enabled = $true
        $buttEXPORT.Enabled = $true
        $buttEXIT.Enabled = $true
        $buttGETMEM.Enabled = $true
        $listBox1.BackColor = [System.Drawing.Color]::FromArgb(255, 255, 255, 255)
        Load-ListBox $listBox1 "Gathering Group Membership Information Complete"

    }
    Else {

        $dataGridView1.BackgroundColor = [System.Drawing.Color]::FromArgb(255, 255, 0, 0)
        $listBox1.ForeColor = [System.Drawing.Color]::White
        $listBox1.BackColor = [System.Drawing.Color]::FromArgb(255, 255, 0, 0)
        Load-ListBox -ListBox $listBox1 " WARNING : The Group -"
        Load-ListBox -ListBox $listBox1 " WARNING : >$GroupSec<" -Append
        Load-ListBox -ListBox $listBox1 " WARNING : Cannot be found." -Append
        Load-ListBox -ListBox $listBox1 " WARNING : Check details and retry." -Append
        $buttDATA.Enabled = $true
        $buttEXPORT.Enabled = $true
        $buttEXIT.Enabled = $true
    }
}

```



```

    $buttGETMEM.Enabled = $true
    #$listBox1.BackColor = [System.Drawing.Color]::FromArgb(255,255,255,255)
    $progressBar1.BackColor = [System.Drawing.Color]::FromArgb(255, 255, 0, 0)
    $progressBar1.ForeColor = [System.Drawing.Color]::FromArgb(255, 255, 0, 0)
  }
}
}

$OnLoadForm_StateCorrection =
{ #Correct the initial state of the form to prevent the .Net maximized form issue
  $form1.WindowState = $InitialFormWindowState
}

#-----
#region Generated Form Code
$form1.BackColor = [System.Drawing.Color]::FromArgb(255, 192, 192, 192)
$System_Drawing_Size = New-Object System.Drawing.Size
$System_Drawing_Size.Height = 511
$System_Drawing_Size.Width = 438
$form1.ClientSize = $System_Drawing_Size
$form1.FormBorderStyle = 2
#$form1.Icon =
[System.Drawing.Icon]::ExtractAssociatedIcon('C:\Support\PS\Scripts\Get_Groups_Members\bin\10
5930_5Aa_icon.ico')
$errorActionPreference = "SilentlyContinue"
$form1.Icon = [System.Drawing.Icon]::ExtractAssociatedIcon("C:\Program Files
(x86)\MoZZaToolkit\Get Group Members v3.0\Media\logo_bcl_ooC_icon.ico")
$errorActionPreference = "Continue"
$form1.DataBindings.DefaultDataSourceUpdateMode = 0
$System_Drawing_Point = New-Object System.Drawing.Point
$System_Drawing_Point.X = 141
$System_Drawing_Point.Y = 0
$form1.StartPosition = 1
$form1.Location = $System_Drawing_Point
$form1.MaximizeBox = $False
$form1.Name = "form1"
$form1.Text = "Get Group Members v3.0"
#$form1.TransparencyKey = [System.Drawing.Color]::FromArgb(255,255,255,255)

$buttDATA.BackColor = [System.Drawing.Color]::FromArgb(255, 128, 128, 128)

$buttDATA.DataBindings.DefaultDataSourceUpdateMode = 0
$buttDATA.ForeColor = [System.Drawing.Color]::FromArgb(255, 255, 255, 0)

$System_Drawing_Point = New-Object System.Drawing.Point
$System_Drawing_Point.X = 305
$System_Drawing_Point.Y = 9
$buttDATA.Location = $System_Drawing_Point
$buttDATA.Name = "buttDATA"
$System_Drawing_Size = New-Object System.Drawing.Size
$System_Drawing_Size.Height = 24

```

```
$System_Drawing_Size.Width = 121
$buttDATA.Size = $System_Drawing_Size
$buttDATA.TabIndex = 8
$buttDATA.Text = "Get Group Data"
$buttDATA.UseVisualStyleBackColor = $False
$buttDATA.add_Click($buttDATA_OnClick)
```

```
$form1.Controls.Add($buttDATA)
```

```
$progressBar1.BackColor = [System.Drawing.Color]::FromArgb(255, 255, 255, 255)
$progressBar1.DataBindings.DefaultDataSourceUpdateMode = 0
$System_Drawing_Point = New-Object System.Drawing.Point
$System_Drawing_Point.X = 13
$System_Drawing_Point.Y = 477
$progressBar1.Location = $System_Drawing_Point
$progressBar1.Name = "progressBar1"
$System_Drawing_Size = New-Object System.Drawing.Size
$System_Drawing_Size.Height = 23
$System_Drawing_Size.Width = 413
$progressBar1.Size = $System_Drawing_Size
$progressBar1.TabIndex = 7
$progressBar1.Visible = $False
```

```
$form1.Controls.Add($progressBar1)
```

```
$buttEXIT.BackColor = [System.Drawing.Color]::FromArgb(255, 128, 128, 128)
```

```
$buttEXIT.DataBindings.DefaultDataSourceUpdateMode = 0
$buttEXIT.ForeColor = [System.Drawing.Color]::FromArgb(255, 0, 0)
```

```
$System_Drawing_Point = New-Object System.Drawing.Point
$System_Drawing_Point.X = 305
$System_Drawing_Point.Y = 75
$buttEXIT.Location = $System_Drawing_Point
$buttEXIT.Name = "buttEXIT"
$System_Drawing_Size = New-Object System.Drawing.Size
$System_Drawing_Size.Height = 39
$System_Drawing_Size.Width = 121
$buttEXIT.Size = $System_Drawing_Size
$buttEXIT.TabIndex = 6
$buttEXIT.Text = "Exit"
$buttEXIT.UseVisualStyleBackColor = $False
$buttEXIT.add_Click($buttEXIT_OnClick)
```

```
$form1.Controls.Add($buttEXIT)
```

```
$buttEXPORT.BackColor = [System.Drawing.Color]::FromArgb(255, 128, 128, 128)
```

```
$buttEXPORT.DataBindings.DefaultDataSourceUpdateMode = 0
$buttEXPORT.ForeColor = [System.Drawing.Color]::FromArgb(255, 0, 255, 0)
```

```
$System_Drawing_Point = New-Object System.Drawing.Point
$System_Drawing_Point.X = 159
$System_Drawing_Point.Y = 75
$buttEXPORT.Location = $System_Drawing_Point
$buttEXPORT.Name = "buttEXPORT"
$System_Drawing_Size = New-Object System.Drawing.Size
$System_Drawing_Size.Height = 39
$System_Drawing_Size.Width = 121
$buttEXPORT.Size = $System_Drawing_Size
$buttEXPORT.TabIndex = 5
$buttEXPORT.Text = "Export Results"
$buttEXPORT.UseVisualStyleBackColor = $False
$buttEXPORT.add_Click($buttEXPORT_OnClick)
```

```
$form1.Controls.Add($buttEXPORT)
```

```
$listBox1.DataBindings.DefaultDataSourceUpdateMode = 0
$listBox1.FormattingEnabled = $True
$System_Drawing_Point = New-Object System.Drawing.Point
$System_Drawing_Point.X = 12
$System_Drawing_Point.Y = 401
$listBox1.Location = $System_Drawing_Point
$listBox1.Name = "listBox1"
$System_Drawing_Size = New-Object System.Drawing.Size
$System_Drawing_Size.Height = 69
$System_Drawing_Size.Width = 414
$listBox1.Size = $System_Drawing_Size
$listBox1.TabIndex = 4
```

```
$form1.Controls.Add($listBox1)
```

```
$System_Windows_Forms_DataGridViewCellStyle_1 = New-Object
System.Windows.Forms.DataGridViewCellStyle
$dataGridView1.AllowUserToOrderColumns = $False
$dataGridView1.AutoSizeColumnsMode = 6
$dataGridView1.BackgroundColor = [System.Drawing.Color]::FromArgb(255, 255, 255, 255)
$dataGridView1.ClipboardCopyMode = 2
$dataGridView1.AlternatingRowsDefaultCellStyle =
$System_Windows_Forms_DataGridViewCellStyle_5
$dataGridView1.DataBindings.DefaultDataSourceUpdateMode = 0
$System_Drawing_Point = New-Object System.Drawing.Point
$System_Drawing_Point.X = 13
$System_Drawing_Point.Y = 133
$dataGridView1.Location = $System_Drawing_Point
$System_Windows_Forms_Padding = New-Object System.Windows.Forms.Padding
$System_Windows_Forms_Padding.Bottom = 3
$System_Windows_Forms_Padding.Left = 3
$System_Windows_Forms_Padding.Right = 3
$System_Windows_Forms_Padding.Top = 3
$dataGridView1.Margin = $System_Windows_Forms_Padding
```

```
$dataGridView1.Name = "dataGridView1"
$dataGridView1.RowHeadersWidthSizeMode = 3
$dataGridView1.SelectionMode = 0
[System_Drawing_Size] = New-Object System.Drawing.Size
[System_Drawing_Size].Height = 261
[System_Drawing_Size].Width = 413
$dataGridView1.Size = [System_Drawing_Size]
$dataGridView1.TabIndex = 3

$form1.Controls.Add($dataGridView1)

$buttGETMEM.BackColor = [System.Drawing.Color]::FromArgb(255, 128, 128, 128)

$buttGETMEM.DataBindings.DefaultDataSourceUpdateMode = 0
$buttGETMEM.ForeColor = [System.Drawing.Color]::FromArgb(255, 0, 255, 0)

[System_Drawing_Point] = New-Object System.Drawing.Point
[System_Drawing_Point].X = 13
[System_Drawing_Point].Y = 75
$buttGETMEM.Location = [System_Drawing_Point]
$buttGETMEM.Name = "buttGETMEM"
[System_Drawing_Size] = New-Object System.Drawing.Size
[System_Drawing_Size].Height = 39
[System_Drawing_Size].Width = 121
$buttGETMEM.Size = [System_Drawing_Size]
$buttGETMEM.TabIndex = 2
$buttGETMEM.Text = "Get Members"
$buttGETMEM.UseVisualStyleBackColor = $False
$buttGETMEM.add_Click($buttGETMEM_OnClick)

$form1.Controls.Add($buttGETMEM)

$comboBox1.AllowDrop = $True
$comboBox1.AutoCompleteMode = 3
$comboBox1.DataBindings.DefaultDataSourceUpdateMode = 0
$comboBox1.FormattingEnabled = $True
[System_Drawing_Point] = New-Object System.Drawing.Point
[System_Drawing_Point].X = 12
[System_Drawing_Point].Y = 39
$comboBox1.Location = [System_Drawing_Point]
$comboBox1.Name = "comboBox1"
[System_Drawing_Size] = New-Object System.Drawing.Size
[System_Drawing_Size].Height = 21
[System_Drawing_Size].Width = 414
$comboBox1.Size = [System_Drawing_Size]
$comboBox1.Sorted = $True
$comboBox1.TabIndex = 1
#$comboBox1.Text = "Paste, Type or Select a Group..."

$form1.Controls.Add($comboBox1)
```

```
$label1.DataBindings.DefaultDataSourceUpdateMode = 0
$label1.Font = New-Object System.Drawing.Font("Microsoft Sans Serif", 8.25, 1, 3, 0)
$label1.ForeColor = [System.Drawing.Color]::FromArgb(255, 0, 0, 255)
```

```
$System_Drawing_Point = New-Object System.Drawing.Point
$System_Drawing_Point.X = 13
$System_Drawing_Point.Y = 10
$label1.Location = $System_Drawing_Point
$label1.Name = "label1"
$System_Drawing_Size = New-Object System.Drawing.Size
$System_Drawing_Size.Height = 23
$System_Drawing_Size.Width = 157
$label1.Size = $System_Drawing_Size
$label1.TabIndex = 0
$label1.Text = "Enter / Select Group Name"
```

```
$form1.Controls.Add($label1)
```

```
#endregion Generated Form Code
```

```
#Save the initial state of the form
```

```
$InitialFormWindowState = $form1.WindowState
```

```
#Init the OnLoad event to correct the initial state of the form
```

```
$form1.add_Load($OnLoadForm_StateCorrection)
```

```
#Show the Form
```

```
$form1.ShowDialog() | Out-Null
```

```
} #End Function
```

```
#Call the Function
```

```
GenerateForm
```