```powershell
<#
  .SYNOPSIS
    Remove-AzureADUserFromLocalAdminGroup.ps1

    The purpose of this file is to rmove an AzureAD user from the local
    admin group on a computer.

    This file was written for Shane
    and is related to work he is doing with Intune.

  .NOTES
    Written by: Jeff Brusoe
    Last Updated: June 1, 2020

    In order to actually remove a user from the local admin group,
    PowerShell must be run as an administrator.
#>

[CmdletBinding()]
[Diagnostics.CodeAnalysis.SuppressMessageAttribute("PSAvoidTrailingWhiteSpace","",Just
fication = "Not relevant")]
[Diagnostics.CodeAnalysis.SuppressMessageAttribute("PSAvoidUsingCmdletAliases","",Just
fication = "Only MS Default Aliases are Used")]
param (
  [switch]$Testing, #This switch will search users but not delete any.
  [string[]]$ExcludedUsers = $null
)

#Initialize Environment
Clear-Host
$Error.Clear()
Set-StrictMode -Version Latest
Set-Location $PSScriptRoot

[string]$LogFileDirectory = "$PSScriptRoot\Logs\"
if (!(Test-Path $LogFileDirectory))
{
  $LogFileDirectory = "$PSScriptRoot\"
}

$TranscriptLogFile = $LogFileDirectory + (Get-Date -Format yyyy-MM-dd-HH-mm) +
"-RemoveAzureADLocalAdmins.txt"
Start-Transcript $TranscriptLogFile

$ExcludedUsers += "helpdesk"
Write-Output "`nExcluded Users:"
Write-Output "$ExcludedUsers`n"

#End of environment configuration block

#Test if PS is running as administrator (required to delete users from admin group)
if (([Security.Principal.WindowsPrincipal]
[Security.Principal.WindowsIdentity]::GetCurrent()).IsInRole([Security.Principal.Windc
sBuiltInRole]::Administrator))
```

```powershell
{
  Write-Output "PowerShell is running as administrator"
}
else
{
  Write-Output "PowerShell is not running as administrator. Program can't delete users from admin group."
  $Testing = $true #Will only log results
}

#Begin main part of program
try
{
  #Generate list of users in local admin account
  #$LocalAdmins = Get-LocalGroupMember -Group Administrators -ErrorAction Stop
  $commandline = 'net localgroup administrators'
    $LocalAdmins = & cmd.exe /c "$commandline"

  Write-Verbose "****************************"
  Write-Verbose "Local Admins:"
  Write-Verbose $LocalAdmins.toString()
  Write-Verbose "****************************"
}
catch
{
  Write-Warning "Error getting members of local admin group. Program is exiting."
  Stop-Transcript
  return
}

$AADAccounts = $LocalAdmins | Select-String -Pattern "hs\\"

if (($AADAccounts | Measure).count -gt 0)
{
  Write-Output "Removing users from Administrators group..."

  foreach ($account in $AADAccounts)
  {
    Write-Output "Removing Account: $account"

    #First, verify account is not in exclusion list
    $ProcessUser = $true
    foreach ($ExcludedUser in $ExcludedUsers)
    {
      if ($account -like "*$ExcludedUser*")
      {
        Write-Output "Account is in exclusion list. Will be skipped."
        $ProcessUser = $false
      }
    }

    if ($ProcessUser)
    {
      try
      {
```

```
104         if ($Testing)
105         {
106           Write-Output "Logging only is being done currently."
107         }
108         else
109         {
110           #Code to actually remove account goes here
111           Write-Output "Beginning to remove user"
112
113           $Error.Clear()
114
115           $commandline = 'net localgroup administrators {0} /DELETE' -f $account
116           & cmd.exe /c "$commandline"
117
118           if ($Error.Count -gt 0)
119           {
120             Write-Warning "Error removing user from local admin group"
121             $Error | fl -Force
122
123             Start-Sleep -s 5
124           }
125         }
126       }
127       catch
128       {
129         Write-Warning "Could not remove $account from built-in Administrators group."
130       }
131     }
132
133     Write-Output "*****************************"
134   }
135 }
136 else
137 {
138    Write-Output "Found no AzureAD accounts, in the built-in Administrators group."
139 }
140
```