```powershell
#Update-ComputerLocationfromSOLE
#Written by: Matt Logue
#Last Updated: September 29, 2020

#The purpose of this file is to updated the location attribute in AD from
#a form submitted to SOLE when a computer changes departments
#Form URL: https://sole.hsc.wvu.edu/Form/5492/Public/Start/QuMQADQRC4bR

#Dependancies:
#1) SqlServer Module is imported
#2) HSC-CommonCodeModule
#3) HSC-SQLModule
#4) SCCM Powershell Module is imported (or SCCM Admin Console is installed)

#Configure Environment
Set-HSCEnvironment

#Get SOLE SQL instance connection string
$SQLPassword = Get-HSCSQLPassword -SOLEDB -Verbose
$ConnectionString = Get-HSCConnectionString -Password $SQLPassword


$Query = 'Select * From HSCITS_ComputerMoveFormData'
Write-Output "Querying SQL..."
$SQLResults = Invoke-SqlCmd -Query $Query -ConnectionString $ConnectionString
$UniqueResults = $SQLResults | Where-Object {$_.FormDate -gt (Get-Date).AddDays(-10)} | Sort-Object -property SerialNumber -unique | Sort-Object -Descending FormDate
Write-output "`nResults from SQL:"
$UniqueResults
Write-output "`n"

###Builds SCCM Device list###

#Connect to SCCM with ConfigMan Module
try {
    $SiteCode = "HS1" # Site code
    $ProviderMachineName = "hssccm.hs.wvu-ad.wvu.edu" # SMS Provider machine name
    $initParams = @{}
    if($null -eq (Get-Module ConfigurationManager)) {
        Import-Module "$($ENV:SMS_ADMIN_UI_PATH)\..\ConfigurationManager.psd1" @initParams
    }
    if($null -eq (Get-PSDrive -Name $SiteCode -PSProvider CMSite -ErrorAction SilentlyContinue)) {
        New-PSDrive -Name $SiteCode -PSProvider CMSite -Root $ProviderMachineName @initParams
    }
    Set-Location "$($SiteCode):\" @initParams
    Write-Output "Connected to SCCM"
}
catch {
    Write-Output "Error in connecting to SCCM.  Check to make sure SCCM powershell module exists on server."

}
##End SCCM Connection


Write-Output "Getting Device List"
```

```powershell
$SCCMDeviceList = Get-CMResource -Fast -ResourceType System | Select-Object Name,MacAddresses
###Device list complete ######

Write-Output "Processing data from SQL and SCCM"
foreach ($entry in $UniqueResults)
{
    $DeviceName = $null
    $inputMac = $entry.SerialNumber
    $inputMac = $inputMac.toUpper()
    $inputMac = $inputMac.Replace(" ","")
    $inputMac = $inputMac.replace("-",":")
    if ($inputMac -notlike "*:*")
    {
        $count = 0
        while ($count -lt ($inputMac.Length - 2))
        {
            $inputMac = $inputMac.Insert(($count)+2,':')
            $count += 3
        }
    }
    #Searching SCCM device information to find Mac Address from SQL query
    $DeviceName = $SCCMDeviceList | Where-Object -property MacAddresses -like "*$inputMac*" | Select-Object -ExpandProperty Name

    #Check if device is not found
    if ($null -eq $DeviceName)
    {
        Write-Output "Device with Mac Address = $inputMac not found in SCCM"
    }

    #If device is found from SCCM list
    else
    {
        try
        {
            #Finds computer name and sets location from SQL query
            Set-ADComputer -Identity $DeviceName -Location $entry.NewLocation
            Write-Output "$DeviceName updated to $($entry.NewLocation)"
        }
        catch
        {
            #Should never execute, but just in case device is in SCCM and not AD
            Write-Output "$DeviceName not found in AD"
        }


    }

}


Invoke-HSCExitCommand -ErrorCount $Error.Count
```