ASSIGNMENT 7

OSYS1200 – Introduction to Windows Administration

NSCC

Ricardo Oliveira

W0428722

Introduction

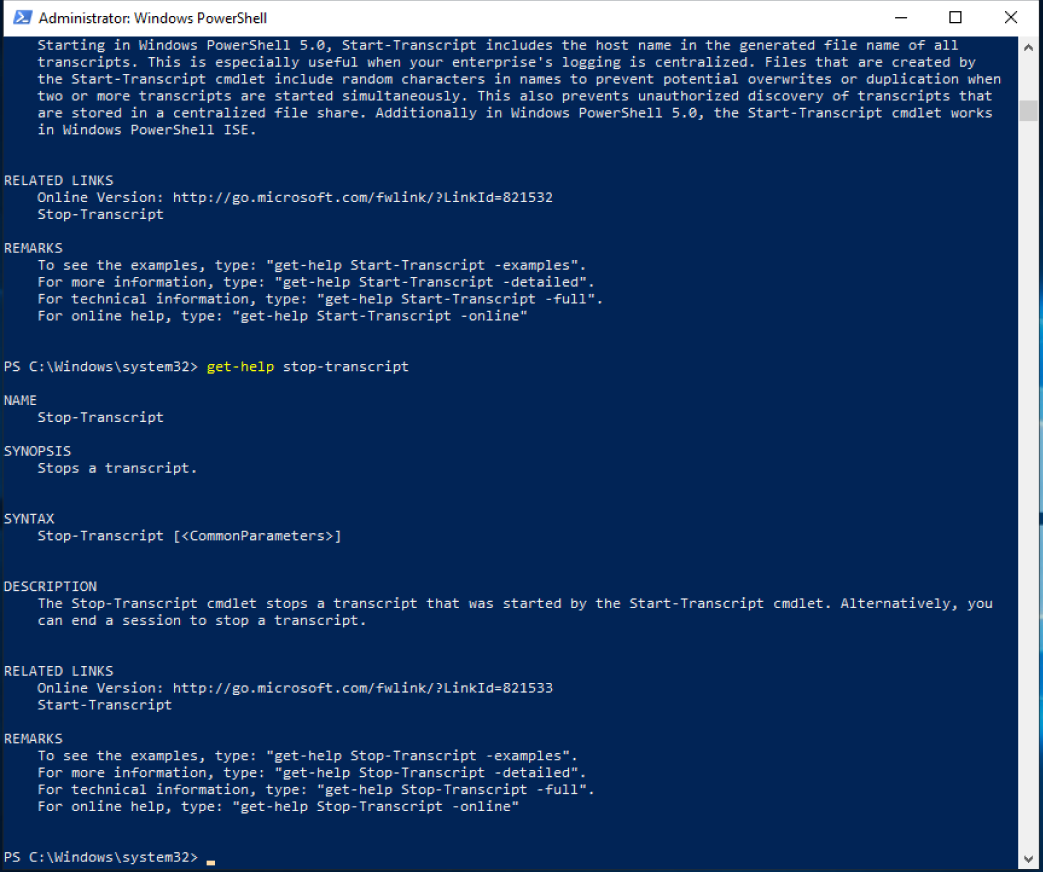
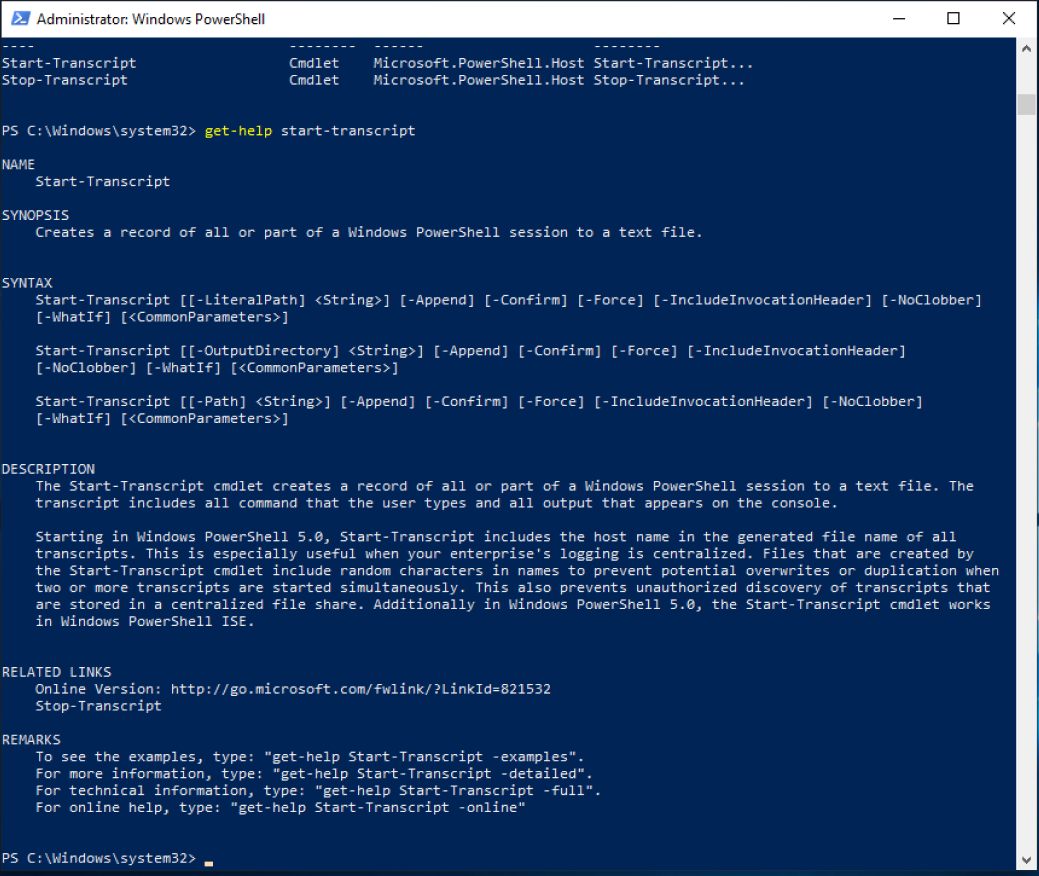
The purpose of this document is to answer any questions asked in the assignment as well as a means to display your change management log and any attachments or screenshots as required.

# Task 1 Questions

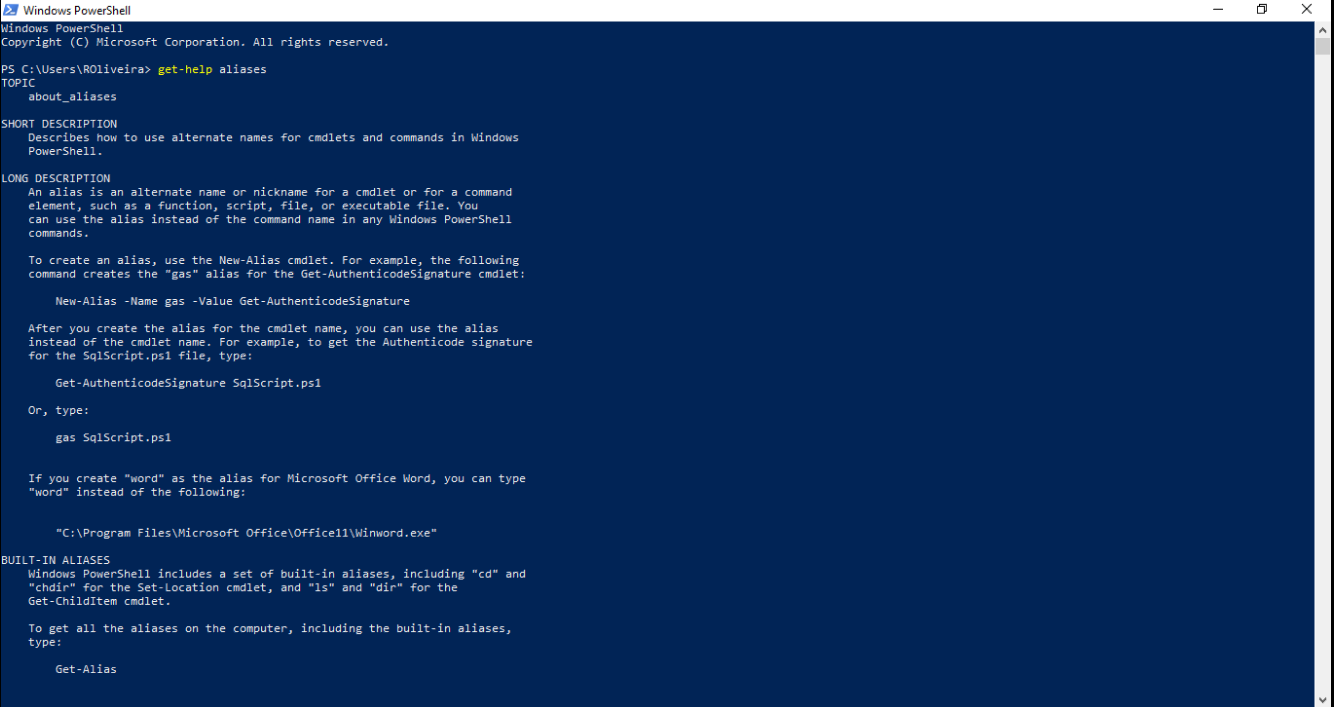
**Part 1**

Using the help system built into PowerShell, find answers to the following questions (each should be answered with a screenshot in your documentation):

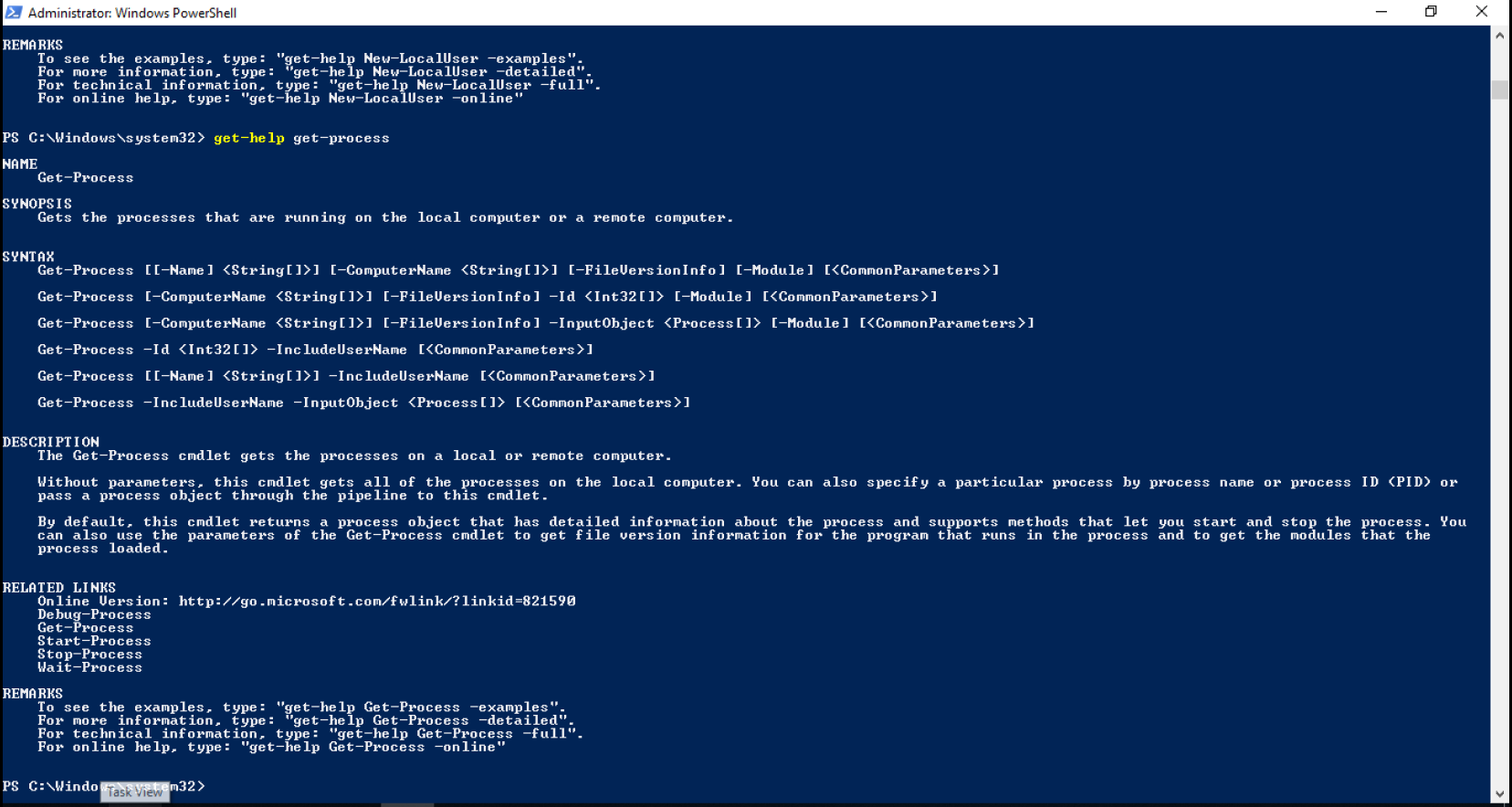
* Find a command that allows you to save a *transcript* of every command ran in the console, and all output to the console, to a text file. Take a screenshot of the commands help page and include in your documentation.



* Find a help page that explains aliases in PowerShell. Take a screenshot of the help page and include in your documentation.



* Find information on the accepted inputs accepted by Get-Process, and the created outputs. Take a screenshot of the information and include in your documentation.



**Part 2** Include a copy of your NewObject.ps1 script converted to text in your Document.

#Set the execution policy as bypass (so you don't get annoying messages from windows, asking for permission)

Set-ExecutionPolicy -ExecutionPolicy Unrestricted -Scope CurrentUser

#Set and store password as a secure string

$Secure\_String\_Pwd = ConvertTo-SecureString "student" -asPlainText -Force

#Set user name as a variable

$UsrName = "Max"

#Set group name as a variable

$GrpName = "ITAdmins"

#Create the new user, set it's password

New-LocalUser $UsrName -Password $Secure\_String\_Pwd -FullName "Maxwell Silverhammer" -Description "Clang Clang" -AccountNeverExpires

#Enable the local user (Don't think that's needed, but got it done when trying out the Logon Screen)

Enable-LocalUser $UsrName

#Create Local Group

New-LocalGroup $GrpName

#Add the user to the Local Group "Users"

Add-LocalGroupMember -Group "Users" -Member $UsrName

#Add the user to the Local Group "ITAdmins"

Add-LocalGroupMember -Group $GrpName -Member $UsrName

#Add the local group "ITAdmins" to the "Administrators" group

Add-LocalGroupMember -Group "Administrators" -Member $GrpName

#Create the new directory "Backup"

New-Item -ItemType "Directory" -Path "C:\Backup"

#Copy the folder and content to the "Backup" folder

Copy-Item -Path "E:\CompanyInc" -Destination "C:\Backup" -Recurse -Container

**Part 3** Include a copy of your Backup.ps1 script converted to text in your Document.

# In the ISE

# open Powershell,

# type set-executionPolicy unrestricted

# then hit Y to accept it.

#Save current suername on a variable

$cname = Get-Content env:username

#Save current date on a variable

$date = Get-Date -Format d.MMM.yyyy

#Save save a directory into a variable

$drive = "C:\Backup\$date"

#Save variable concatenation

$dest = $drive + $cname

#Create a new directory named as the concatenated variable (date/username)

New-Item -Path $dest -ItemType directory

#Save variable with the specific directory

$path = "E:\CompanyInc"

#Save variable with "everything" wildcard

$include = @("\*.\*")

#Reach for the files/folders inside the directory and copy them to the $dest

Get-ChildItem -Path $path -Include $include -Recurse -Force -ErrorAction SilentlyContinue | foreach { copy-item -Path $\_ -Destination $dest -Recurse -Container } -ErrorAction SilentlyContinue

**Part 1**

There are four (4) PowerShell Profiles, create a document in a Summary Table format that:

* + Lists each PowerShell Profile
  + List the Path of each Profile
  + Gives a brief explanation of the description and difference between each profile.

|  |  |  |
| --- | --- | --- |
| **PowerShell Profile** | **Path** | **Description** |
| Current User, Current Host | $Home\[My]Documents\PowerShell\Microsoft.PowerShell\_profile.ps1 | PowerShell standard Profile –Recognizes only the current user and local host |
| Current User, All Hosts | $Home\[My]Documents\PowerShell\Profile.ps1 | Assume the global variables to the current user |
| All Users, Current Host | $PsHome\Microsoft.PowerShell\_profile.ps1 | The standard settings to all the users on the local machine |
| All Hosts, All Users | $PsHome\Profile.ps1 | Settings to all users on global network scale |

**Part 2** There are six (6) Execution Policies for PowerShell (not including Default), create a document in a Summary Table format that:

* + Lists each Execution Policy
  + Gives a brief explanation of each policy.

|  |  |
| --- | --- |
| **PowerShell Execution Policy** | **Description** |
| AllSigned | Requires all the scripts and configuration files to be signed by a trusted publisher, including local ones. Prompts you before running scripts from publishers not classified as trusted or untrusted |
| Bypass | Nothing is blocked and there are no warnings or prompts.  Designed to be used on a larger application that has its own security protocols |
| RemoteSigned | Default execution policy for windows server computers; Scripts can run; Requires trusted publishers sign on downloaded files; |
| Restricted | Default execution policy for windows client computers; Permits individual commands, but does not allow scripts; |
| Undefined | No execution policy set in the current scope; The effective policy is Restricted |
| Unrestricted | The default execution policy for non-Windows computers and cannot be changed; Unsigned scripts can run; Warns before running outside scripts |

**Citation:**

SA, Microsoft, 11/29/2017 – About Profiles - <https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_profiles?view=powershell-6>

SA, Microsoft, 03/31/2019 – About Execution Policies - <https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.core/about/about_execution_policies?view=powershell-6>

# Change management

