# Lab-2-1: PowerShell Functions

# Objectives

* Gather a basic understanding of and experience with Microsoft PowerShell



# Preparation

1. Open the PowerShell ISE (To do that type ‘powershell’ in the search field on your workstation
2. Select Windows **PowerShell ISE**). You can also perform this lab in the basic PowerShell environment (Windows **PowerShell**), but you may be missing the code completion facilities of the Interactive Scripting Environment (ISE).
3. You can also pin it to your taskbar for ease of access in future.

# Exploring PowerShell

Solve the following tasks based on the **integrated help** (get-help), **lecture slides** and PowerShell **online reference**.

**Write down the commands you used to achieve the tasks.**

1. Find information about your computer

**Ans:**

PS C:\> Get-Command | where {$\_.Name -like "\*computer\*"}

CommandType Name Version Source

----------- ---- ------- ------

Function Get-MpComputerStatus 1.0 Defender

Cmdlet Add-Computer 3.1.0.0 Microsoft.PowerShell.Management

Cmdlet Checkpoint-Computer 3.1.0.0 Microsoft.PowerShell.Management

Cmdlet Disable-ComputerRestore 3.1.0.0 Microsoft.PowerShell.Management

Cmdlet Enable-ComputerRestore 3.1.0.0 Microsoft.PowerShell.Management

Cmdlet Get-ComputerInfo 3.1.0.0 Microsoft.PowerShell.Management

Cmdlet Get-ComputerRestorePoint 3.1.0.0 Microsoft.PowerShell.Management

Cmdlet Remove-Computer 3.1.0.0 Microsoft.PowerShell.Management

PS C:\> Get-ComputerInfo -Property "Bios\*"

BiosCharacteristics : {7, 9, 11, 12...}

BiosBIOSVersion : {HPQOEM - 20200220}

BiosBuildNumber :

BiosCaption : Default System BIOS

BiosCodeSet :

BiosCurrentLanguage : en|US|iso8859-1

BiosDescription : Default System BIOS

BiosEmbeddedControllerMajorVersion : 255

BiosEmbeddedControllerMinorVersion : 255

BiosFirmwareType : Bios

BiosIdentificationCode :

BiosInstallableLanguages : 12

BiosInstallDate :

BiosLanguageEdition :

BiosListOfLanguages : {en|US|iso8859-1, fr|FR|iso8859-1, es|ES|iso8859-1, de|DE|iso8859-1...}

BiosManufacturer : Hewlett-Packard

BiosName : Default System BIOS

BiosOtherTargetOS :

BiosPrimaryBIOS : True

BiosReleaseDate : 20/02/2020 1:00:00 pm

BiosSeralNumber : AUD507057C

BiosSMBIOSBIOSVersion : L01 v02.78

BiosSMBIOSMajorVersion : 2

BiosSMBIOSMinorVersion : 7

BiosSMBIOSPresent : True

BiosSoftwareElementState : Running

BiosStatus : OK

BiosSystemBiosMajorVersion : 2

BiosSystemBiosMinorVersion : 78

BiosTargetOperatingSystem : 0

BiosVersion : HPQOEM - 20200220

PS C:\> Get-ComputerInfo -Property "Windows\*"

WindowsBuildLabEx : 19041.1.x86fre.vb\_release.191206-1406

WindowsCurrentVersion : 6.3

WindowsEditionId : Enterprise

WindowsInstallationType : Client

WindowsInstallDateFromRegistry : 1/01/1970 12:00:00 am

WindowsProductId :

WindowsProductName : Windows 10 Enterprise

WindowsRegisteredOrganization :

WindowsRegisteredOwner : BUILD

WindowsSystemRoot : C:\WINDOWS

WindowsVersion : 2004

1. Find the name of your computer

**Ans:** PS C:\> Get-ComputerInfo -Property "\*csname\*"

CsName

------

D207-61581

1. Create a file named hello.txt on your desktop and Copy that file from your Desktop to C:\

**Ans:**

PS C:\> Out-File -FilePath C:\Users\leggtc1\Desktop\test.txt -Encoding utf8

PS C:\> Copy-Item -Path C:\Users\leggtc1\Desktop\test.txt -Destination C:\

1. Copy all files from Desktop to C:\myfolder

**Ans:**

PS C:\> Copy-Item -Path "C:\Users\leggtc1\Desktop" -Destination "C:\myfolder" -Recurse

1. Find all pdf files created on April 2019 on your c:\ drive.

**Ans:**

PS C:\> Get-ChildItem -Path C:\\* -File -Include \*.txt | where {$\_.CreationTime -gt "07/01/2020"}

Directory: C:\

Mode LastWriteTime Length Name

---- ------------- ------ ----

-a---- 30/01/2021 2:03 pm 53 ICT.txt

-a---- 28/07/2021 9:10 am 3 test.txt

1. Write a function named HelloWorld to print ‘Hello World’

**Ans:**

function HelloWorld() {

Write-Host "Hello World"

}

HelloWorld;

PS C:\> HelloWorld;

Hello World

1. Write a function named get-myfiles which will take parameters like the following and show the desired output:

Param (

[string[]] $fileTypes,

[int] $month,

[int] $year,

[string[]] $path

)

The function can be run like this:

get-myfiles -filesTypes \*.pdf -month 6 -year 2018 -path c:\user\yourname\

* Hints:
  + use get-childitem
  + get-childitem | get-member
  + user where-object

**Ans:**

function get-myfiles {

param([string]$filetypes, [int]$month, [int]$year, [string]$path)

Get-ChildItem -Path $path -File $filetypes | Where-Object {$\_.CreationTime -gt [DateTime]"$year-$month-01"}

}

get-myfiles -filetypes \*.pdf -month 1 -year 2020 -path C:\Users\leggtc1\Desktop\

PS C:\> C:\Users\leggtc1\OneDrive - Otago Polytechnic\BIT\_Year 2\2.OPs Eng\Week 2\lab2.1.ps1

Directory: C:\Users\leggtc1\Desktop

Mode LastWriteTime Length Name

---- ------------- ------ ----

-a---- 4/07/2020 12:35 am 3617184 git-cheatsheet-AR-grey.pdf

-a---- 4/07/2020 12:38 am 1886623 git-cheatsheet-CN-dark.pdf

-a---- 3/07/2020 9:34 pm 172109 git-cheatsheet-DE-dark.pdf

-a---- 3/07/2020 8:21 pm 150846 git-cheatsheet-DE-grey.pdf

-a---- 3/07/2020 8:33 pm 150572 git-cheatsheet-DE-white.pdf

-a---- 2/12/2020 11:50 pm 170868 git-cheatsheet-EN-dark.pdf

-a---- 3/07/2020 8:27 pm 151838 git-cheatsheet-EN-grey.pdf

-a---- 3/07/2020 8:32 pm 151000 git-cheatsheet-EN-white.pdf

-a---- 3/07/2020 11:58 pm 165619 git-cheatsheet-ESP-dark.pdf

-a---- 3/07/2020 8:23 pm 152034 git-cheatsheet-ESP-grey.pdf

-a---- 3/07/2020 9:39 pm 151881 git-cheatsheet-ESP-white.pdf

-a---- 4/07/2020 12:01 am 161065 git-cheatsheet-PT-dark.pdf

-a---- 3/07/2020 8:48 pm 159593 git-cheatsheet-PT-grey.pdf

-a---- 3/07/2020 11:54 pm 150023 git-cheatsheet-PT-white.pdf