

Colin Smith X00175174

# Instructions:

Create a MS Word document with the solutions of each exercise. Include your name and x-number.

# Exercise 1.

Create the following PowerShell Module (***Hello.psm1***) and show the execution of it (screenshot). Remember to change the Execution Policy to Unrestricted.

## Code:

Function Get-Hello

{

Write-Host "Hello this is my first PowerShell Module."

}

Function Get-Hello2

{

Param($name)

Write-Host "Hello $name this is my first PowerShell Module."

}

Graphical user interface

Description automatically generated

# Exercise 2.

* Create the following PowerShell Module (***ChristmasDays.psm1***) and show the execution of it (screenshot).
* Remember to change the Execution Policy to Unrestricted.

## Code:

Function Get-DaysTilChristmas

{

<#

.Synopsis

This function calculates the number of days until Christmas

.Description

This function calculates the number of days until Christmas

.Example DaysTilChristmas

#>

$Christmas=Get-Date("25 Dec " + (Get-Date).Year.ToString() + " 7:00 AM")

$Today = (Get-Date)

$TimeTilChristmas= $Christmas -$Today

Write-Host $TimeTilChristmas.Days"Days 'til Christmas"

}

Graphical user interface, text, application

Description automatically generated



# Exercise 3.

* Write a PowerShell module (***BirtdayDays.psm1***) to check the number of days ‘till your next

birthday.

* Include the code and screenshot.

Graphical user interface, text

Description automatically generated

# Exercise 4.

* Write a PowerShell module (***myloops.psm1***) with two functions to count from ***1*** to ***a limit***

passed by the user as a parameter.

* + The first function should use a “***do while”*** loop
  + and the second should use a “***for”*** loop.
* Display a count iteration message to console for each loop iteration.
* Use basic parameter validation to ensure that the user passes an integer value.
* Include the code and screenshot.