Questions	Notes
What is Powershell?	-> Powershell is an object based command line interface, similar as command prompt with larger scale of capabilities, because command prompt is text based and powershell is object based. It is build on .NET Framework.
Why Powershell?	-> In text-based command line interface, you need to handle output somehow to use it further, for example using of pipe(|) operator. Powershell on the other hand can gather variables, objects etc that powershell command returns. -> PowerShell scripts are saved with .ps1 extension.
Basic Commands	• $psversiontable.psversion - Check powershell version. • get-command - List of all commandlets available. • cd, ls, mkdir, clear
Command Line Interpreter	-> Interprets what your provide to it. -> Get-ExecutionPolicy : If returns restricted, remote access does not have admin access. -> You can use Get cmdlet to retrieve values, and Set for setting the values.
Powershell Cmdlets	-> Purpose of a cmdlet is to carry out a particular function. -> write-host "statement" : used to print message. -> We can add certain parameters to customize, for e.g. write-host "Hackedddd" -ForegroundColor yellow -> get-help "cmdlet" : Information about the cmdlet. -> get-process, get-service : To get all processes and services running. -> We can start or stop and process or service using - start-process, start-service or stop-process, stop-service.

Objects	-> PowerShell is based on .NET framework, which means that all outputs and responses generated are .NET objects, and these objects can be use by powershell. Objects are represented by properties - attributes of objects and methods - actions that objects can do. -> By piping out cmdlets with get-member cmdlet, we can see such information. e.g. - Get-Process | Get-Member -> We can also filter out the output using options such as : Get-Process | Get-Member -MemberType Property
Functions	-> Blocks of code that performs a specific task is called a function. -> PowerShell cmdlets are not written in Powershell, instead they are written in some other language and compiled to be used in PowerShell. On the other hand, functions are written in PowerShell itself. ->Syntax : function Function_Name {//Function_Body} -> The function, then gets saved in the PowerShell, and be called by typing in the function name.
Modules	-> .psm1 extension -> Used to import functions from different previously written code.