

TechMentor @ Microsoft HQ 2025 - PowerShell Hands-On Labs

Lab 1: Introduction to PowerShell

Install PowerShell

1. Install PowerShell 7 (if you can)
2. Get help for a command before help is installed:

```
Get-Help Get-Command -Full
```

Update help

1. Run `Update-Help` to install the latest help on your computer.

```
Update-Help -Force
```

This requires that the shell is running with elevated privileges (use the Run as Administrator option).

NOTE: You will get error messages about failing to download help for specific modules. This is a known issue.

2. Get help after help is installed, run:

```
Get-Help Get-Command -Full
```

Notice the difference in the amount of help available.

Lab 2 - Command discovery and problem solving

Command discovery

1. Can you find any cmdlets capable of converting other cmdlets' output into HTML?
2. How many cmdlets are available for working with processes? (Hint: remember that cmdlets all use a singular noun.)
3. Is there a way to shutdown a remote computer?

Problem solving

1. How would you find the Background Intelligent Transfer Service?

2. What is the current status of the Background Intelligent Transfer Service?
 3. Is the Background Intelligent Transfer Service set to start automatically?
 4. How would you stop the Background Intelligent Transfer Service?
-

Lab 3 - Modules & Pipeline

Modules

1. What modules are available on your system?
2. What is the latest version of the Pester module available in the PowerShell Gallery?

Using the pipeline

1. What's difference between the following commands?

```
Get-ChildItem -File | Format-Table Name, Length | Sort-Object Length
Get-ChildItem -File | Format-Table Name, Length | Get-Member
Get-ChildItem -File | Sort-Object Length | Format-Table Name, Length
```

2. What properties are available on the objects returned by `Get-Process`? How can you display all of them?
3. Run the following command and compare the output:

```
Get-Process pwsh
Get-Process pwsh | Format-Table
Get-Process pwsh | Format-List
Get-Process pwsh | Select-Object -Property *
```

Lab 4 - Functions & Scripts

Create a function

Create a function called `Get-AssetInfo` that retrieves information about the computer system and BIOS. The function should accept a parameter for the computer name, defaulting to the local computer if none is provided. It should return a custom object with properties such as **ComputerName**, **Manufacturer**, **Model**, **BIOSVersion**, and **SerialNumber**.

Lab 5 - Create a profile

1. Create a profile script
2. Add the function you created in Lab 4

3. Configure PSReadLine key bindings for the following keys:

Key chord	Function
Enter	ValidateAndAcceptLine
Alt+a	SelectCommandArgument
F1	ShowCommandHelp
Alt+h	ShowParameterHelp

Lab 6 - PowerShell Remoting

There is no Lab 6 in this workshop.