Welcome to Cyber Aces, Module 3! This module provides an introduction to the latest shell for Windows, PowerShell. In this session we'll discuss cmdlets.
This training material was originally developed to help students, teachers, and mentors prepare for the Cyber Aces Online Competition. This module focuses on the basics of what an operating systems is as well as the two predominant OS's, Windows and Linux. In this session we will provide a walkthrough of the installation a Windows VM using VMware Fusion (MacOS) and VMware Player (Windows & Linux). These sessions include hands-on labs, but before we begin those labs we need to install the operating systems used in those labs. We will be using VMware to virtualize these operating systems. You can use other virtualization technologies if you like, but instruction for their setup and use are not included in this training.

The three modules of Cyber Aces Online are Operating Systems, Networking, and System Administration.

For more information about the Cyber Aces program, please visit the Cyber Aces website at https://CyberAces.org/.
Is this section, we'll spend time discussing cmdlets. We'll cover the help system and how to find cmdlets. We'll also discuss aliases for cmdlets and some of the common cmdlets.
The most important commands to know are the ones that get more help and information. The two most important commands in this regard are Get-Help and Get-Command. The Get-Help cmdlet is the PowerShell equivalent of "man" on Linux. It displays information on PowerShell's commands and concepts. When used with the name of a cmdlet, it returns the synopsis and syntax for the command. To get examples of the cmdlet in use, use the "-Examples" switch. For the full output, including synopsis, syntax, parameter descriptions and examples, use the "-Full" switch. The formatting of Get-Help's output is very similar to that of Linux's man.
The Get-Command cmdlet "gets basic information about cmdlets and other elements of Windows PowerShell commands." Its most common use is to find other cmdlets based on a verb or noun by using the "-Verb" or "-Noun" parameters. To see all the commands used to manage services we can use the following command:

```powershell
PS C:\> Get-Command -Noun Services
```

The -Verb parameter is available as well. To list all cmdlets that use the Get verb we can use the following command:

```powershell
PS C:\> Get-Command -Verb Get
```

The -Module parameter can be used to find command specific to a loaded module. Many 3rd party products have a PowerShell interface which loads another module. We can list all loaded modules with “Get-Module”. To see the commands specific to a loaded module we use “Get-Command -Module ModuleName”.

All of these parameters take wildcard characters and they can be combined to provide a more granular search.

```powershell
PS C:\> Get-Command -Module Vm* -Verb Get
```
Aliases are a very handy way to simplify the commands that you use and make typing faster and more efficient. Many commands that are implemented in CMD or Bash are aliased using the respective shell's command name. To copy an item in CMD the command "copy" is used, in Bash the command is "cp". Both of these are aliases for the "Copy-Item" cmdlet.

Many times it is useful to create an alias for a commonly used command. The most commonly used command without an alias is Select-String and the common alias is ss. To create the alias for the command we use this command:

```
PS C:\> Set-Alias -Name ss -Value Select-String
```

The Set-Alias takes positional parameters, so it knows the first input is the alias name and the second is the command we want to alias. We could type this command instead.

```
PS C:\> Set-Alias ss Select-String
```
1) What is the best way to see which cmdlets are available to manipulate or get the list of commands entered during the current session?
   - Get-Command History
   - Get-Command -Noun History
   - Get-YeOldeCommands
   - Get-Command -Noun History -Verb History
   - Get-Command -Verb History

2) Which command would find Aliases for the Get-ChildItem cmdlet?
   - Get-ChildItem -Help
   - Get-Help Get-ChildItem
   - Get-Alias -Definition Get-ChildItem
   - Get-Command Get-ChildItem
   - Get-ChildItem -?
1) What is the best way to see which cmdlets are available to manipulate or get the list of commands entered during the current session?

`Get-Command -Noun History`

This will find commands that have the noun of History and will show the commands to Get, Add, Clear, and Invoke items in the command history.

2) Which command would find Aliases for the Get-ChildItem cmdlet?

`Get-Alias -Definition Get-ChildItem`

The cmdlet we need to use is Get-Alias with the `-Definition` parameter.
Above is a list of the common cmdlets and the equivalent commands in Bash and CMD.

<table>
<thead>
<tr>
<th>Cmdlet</th>
<th>Alias(es)</th>
<th>Equivalent Command Bash</th>
<th>Equivalent Command CMD</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Copy-Item</td>
<td>copy</td>
<td>cp</td>
<td>copy</td>
<td>Copies an item from one location to another</td>
</tr>
<tr>
<td>ForEach-Object</td>
<td>foreach</td>
<td>for</td>
<td></td>
<td>Performs an operation against each of a set of input objects</td>
</tr>
<tr>
<td>Format-List</td>
<td>fi, lists</td>
<td>for</td>
<td></td>
<td>Performs the output as a list of properties</td>
</tr>
<tr>
<td>Format-Table</td>
<td>ft</td>
<td></td>
<td></td>
<td>Formats the output as a table</td>
</tr>
<tr>
<td>Get-Command</td>
<td>cmd</td>
<td></td>
<td></td>
<td>Gets basic information about cmdlets and other elements</td>
</tr>
<tr>
<td>Get-Content</td>
<td>get</td>
<td>cat, type</td>
<td></td>
<td>Gets the content of the item at the specified location (i.e. reads a file)</td>
</tr>
<tr>
<td>Get-Process</td>
<td>gpp</td>
<td>ps, tasklist</td>
<td></td>
<td>Gets the processes that are running</td>
</tr>
<tr>
<td>Group-Object</td>
<td>group</td>
<td></td>
<td></td>
<td>Groups objects that contain the same value for specified properties</td>
</tr>
<tr>
<td>Get-Help</td>
<td>man</td>
<td>man, help</td>
<td></td>
<td>Displays information about Windows PowerShell commands and concepts</td>
</tr>
<tr>
<td>Select-String</td>
<td>grep, find</td>
<td></td>
<td></td>
<td>Finds text in strings and files</td>
</tr>
<tr>
<td>Select-Object</td>
<td>select</td>
<td></td>
<td></td>
<td>Selects specified properties of an object</td>
</tr>
<tr>
<td>Sort-Object</td>
<td>sort</td>
<td>sort</td>
<td></td>
<td>Sorts objects by property values</td>
</tr>
<tr>
<td>Stop-Process</td>
<td>kill</td>
<td>kill, taskkill</td>
<td></td>
<td>Terminates a running process</td>
</tr>
<tr>
<td>Where-Object</td>
<td>where</td>
<td></td>
<td></td>
<td>Filter that controls which objects will be passed along a command pipeline</td>
</tr>
<tr>
<td>Write-Output</td>
<td>echo</td>
<td>echo</td>
<td>echo</td>
<td>Sends the specified objects to the next command in the pipeline. If it is the last command, it displays the object</td>
</tr>
</tbody>
</table>
At first glance, you might wonder why cmd.exe's "dir" command has been replaced by something as weird sounding as "Get-ChildItem". Well, "Get-ChildItem" does more than just list files and directories, and that is why the name is more generic. This cmdlet returns objects from any container, and the filesystem is just one of many containers. For example, it can also be used to list the system certificates ("Get-ChildItem cert:")) and the registry ("Get-ChildItem HKLM:").
1) Which command would display a directory listing where the output is sorted alphabetically?

- `Get-ChildItem a b c d e f g h i j k l m n o p q r s t u v w x y z`
- `Get-ChildItem | Sort-Object -Property Name -Descending`
- `Get-ChildItem | Sort-Object -Property Name`
- `Sort-Object -Property Name | Get-ChildItem`
- `Get-ChildItem | Sort Alphabetically`

2) Which of these commands would NOT display the contents of a text file?

- `type file.txt`
- `cat file.txt`
- `Get-Content file.txt`
- `view file.txt`
- `gc file.txt`
1) Which command would display a directory listing where the output is sorted alphabetically?

- `Get-ChildItem | Sort-Object -Property Name`

The output of our directory listing, from Get-ChildItem, is piped into the Sort-Object cmdlet where sorting is done on the Name property. By default, the sorting is done in Ascending order so no other parameters or switches are necessary.

2) Which of these commands would NOT display the contents of a text file?

- `view file.txt`

There is no view command or default alias in PowerShell
Congratulations! You have completed the session on cmdlets in PowerShell.
Module 3 - System Administration
PowerShell

- Introduction
- cmdlets
- Scripting, Variables & Syntax
- Flow Control & Output
- Practical Uses

This portion intentionally left blank.