

**The following are the end-of-chapter Labs from Chapter 3 through Chapter 6 of our textbook. Answer each question or describe the PowerShell command(s) necessary to accomplish the task described.**

**Chapter 3: *Using the help system***

1. First, run Update-Help and ensure it completes without errors. That will get a copy of the help on your local computer. You'll need an internet connection, and the shell needs to run under elevated privileges (which means it must say "Administrator" in the shell window's title bar).
2. Can you find any cmdlets capable of converting other cmdlets' output into HTML?
3. Are there any cmdlets that can redirect output into a file, or to a printer?
4. How many cmdlets are available for working with processes? (Hint: remember that cmdlets all use a singular noun.)
5. What cmdlet might you use to write to an event log?
6. You've learned that aliases are nicknames for cmdlets; what cmdlets are available to create, modify, export, or import aliases?
7. Is there a way to keep a transcript of everything you type in the shell, and save that transcript to a text file?

8. It can take a long time to retrieve all of the entries from the Security event log. How can you get only the 100 most recent entries?
  
9. Is there a way to retrieve a list of the services that are installed on a remote computer?
  
10. Is there a way to see what processes are running on a remote computer?
  
11. Examine the help file for the Out-File cmdlet. The files created by this cmdlet default to a width of how many characters? Is there a parameter that would enable you to change that width?
  
12. By default, Out-File will overwrite any existing file that has the same filename as what you specify. Is there a parameter that would prevent the cmdlet from overwriting an existing file?
  
13. How could you see a list of all aliases defined in PowerShell?
  
14. Using both an alias and abbreviated parameter names, what is the shortest command line you could type to retrieve a list of running processes from a computer named Server1?
  
15. How many cmdlets are available that can deal with generic objects? (Hint: remember to use a singular noun like “object” rather than a plural one like “objects”.)
  
16. This chapter briefly mentioned arrays. What help topic could tell you more about them?

## Chapter 4: *Running commands*

Note: For this lab, you'll need a Windows 8 or Windows Server 2012 computer running PowerShell v3.

1. Display a list of running processes.
2. Display the 100 most recent entries from the Application event log (don't use Get-WinEvent for this. We've shown you another command that will do this task).
3. Display a list of all commands that are of the "cmdlet" type (this is tricky—we've shown you Get-Command, but you're going to have to read the help to find out how to narrow down the list, as we've asked).
4. Display a list of all aliases.
5. Make a new alias, so you can run `d` to get a directory listing.
6. Display a list of services that begin with the letter M. Again, read the help for the necessary command—and don't forget that the asterisk (\*) is a near-universal wildcard in PowerShell.
7. Display a list of all Windows Firewall rules. You'll need to use Help or Get-Command to discover the necessary cmdlet.
8. Display a list only of inbound Windows Firewall rules. You can use the same cmdlet as in the previous task, but you'll need to read its help to discover the necessary parameter and its allowable values.

## Chapter 5. *Working with Providers*

1. In the registry, go to  
HKEY\_CURRENT\_USER\software\microsoft\Windows\currentversion\explorer. Locate the Advanced key, and set its DontPrettyPath property to 1.
2. Create a zero-length file named C:\Test.txt (use New-Item).
3. Is it possible to use Set-Item to change the contents of C:\Test.txt to TESTING? Or do you get an error? If you get an error, why?
4. What are the differences between the -Filter, -Include, and -Exclude parameters of Get-ChildItem?

## Chapter 6. *The pipeline: connecting commands*

1. What happens if you run `Get-Service | Export-CSV services.csv | Out-File` from the console? Why does that happen?
2. Apart from getting one or more services and piping them to `Stop-Service`, what other means does `Stop-Service` provide for you to specify the service or services you want to stop? Is it possible to stop a service without using `Get-Service` at all?
3. What if you want to create a pipe-delimited file instead of a comma-separated (CSV) file? You would still use the `Export-CSV` command, but what parameters would you specify?
4. Is there a way to eliminate the `#` comment line from the top of an exported CSV file? That line normally contains type information, but what if you want to omit that from a particular file?
5. `Export-CliXML` and `Export-CSV` both modify the system because they can create and overwrite files. What parameter would prevent them from overwriting an existing file? What parameter would ask you if you were sure before proceeding to write the output file?
6. Windows maintains several regional settings, which include a default list separator. On U.S. systems, that separator is a comma. How can you tell `Export-CSV` to use the system's default separator, rather than a comma?