Course 10325A: Automating Administration with Windows PowerShell 2.0

Duration: 5 days

Course OutlineModule 1: Fundamentals for Using Windows PowerShell v2 This module provides background on Windows PowerShell v2 and where it fits into the Windows technology family. It also covers installation and configuration of Windows PowerShell. It familiarizes students with the interactive shell console, and shows how to operate and interpret the built-in help system. This module focuses on shell's discoverability features, including the online help system and cmdlet inventory. Finally, this module describes how the Windows PowerShell pipeline works at a basic level.

Lessons

- Windows PowerShell Technology Background and Overview
- Windows PowerShell as an Interactive Command-Line Shell
- Using the Windows PowerShell Pipeline

Lab : Using Windows PowerShell as an Interactive Command-Line Shell

- Searching for text files.
- Browsing the registry.
- Discovering additional commands and viewing help.
- Adding additional commands to your session.
- Formatting output.

Lab : Using the Windows PowerShell Pipeline

- Stopping and restarting a Windows service.
- Exploring objects returned by PowerShell commands.
- Processing PowerShell output.

After completing this module, students will be able to:

- Provide an overview of Windows PowerShell Technology.

- Use Windows PowerShell as an Interactive Command-Line Shell.

- Use Windows PowerShell Pipeline.

Module 2: Understanding and Using the Formatting System This module explains how the PowerShell formatting subsystem works, and shows how to customize the output of cmdlets. It covers the rules that the shell follows for formatting objects by default, and explains how to use the four formatting cmdlets – and their parameters – to customize and control the output displayed on-screen or written to a file, printer, or other output destination.

Lessons

- Understanding the Formatting System
- Using the Formatting System

Lab : Using the Formatting Subsystem
- Displaying calculated properties
- Displaying a limited number of columns
- Displaying all properties and values of objects
- Viewing objects via HTML
- Displaying a limited number of properties
- Displaying objects using different formatting
- Displaying a sorted list of objects

After completing this module, students will be able to:

- Explain the Formatting System.

- Use the Formatting System.

Module 3: Core Windows PowerShell Cmdlets This module describes several core cmdlets that are used in many different administrative tasks. This module also covers the basics of filtering objects that are in the PowerShell pipeline. It explains advanced pipeline techniques including pipeline parameter binding and in-pipeline object manipulation.

Lessons
- Core Cmdlets for Everyday Use
- Comparison Operators, Pipeline Filtering, and Object Enumeration
- Advanced Pipeline Techniques

Lab : Using the Core Cmdlets
- Sorting and selecting objects
- Retrieving a number of objects and saving to a file
- Comparing objects using XML
- Saving objects to a CSV file
- Measuring a collection of objects

Lab : Filtering and Enumerating Objects in the Pipeline
- Comparing numbers (integer objects)
- Comparing string objects
- Retrieving processes from a computer
- Retrieving services from a computer
- Iterating through a list of objects

Lab : Using Pipeline Parameter Binding
- Using advanced pipeline features
- Working with multiple computers
- Stopping a list of processes
- Binding properties to parameters

After completing this module, students will be able to:

- Identify Core Cmdlets for Everyday Use.

- Explain Comparison Operators, Pipeline Filtering, and Object Enumeration

- Apply Advanced Pipeline Techniques.

Module 4: Windows Management Instrumentation
This module explains what Windows Management Instrumentation (WMI) is, how it works, and how it can be used from within Windows PowerShell. It describes the structure and security of WMI, and how to query WMI information both from local and remote computers. It also explains how to invoke WMI methods to accomplish configuration changes and other tasks, and how to write commands that respond to WMI events triggered by the operating system.**Lessons**
- Windows Management Instrumentation Overview
- Using Windows Management Instrumentation
Lab : Using WMI in Windows PowerShell
- Building computer inventory
- Discovering the WMI classes and namespaces
- Generating a logical disk report for all computers
- Listing local users and groups
After completing this module, students will be able to:

- Give an overview of Windows Management Instrumentation.

- Use Windows Management Instrumentation.

- Apply advanced Windows Management Instrumentation techniques.

Module 5: Automating Active Directory Administration

This module introduces the concept of Active Directory administrative automation. It explains how to retrieve, create, modify, move, and remove objects in the Active Directory. It focuses on PowerShell-centric techniques rather than scripting, and heavily leverages on basic and advanced pipeline techniques covered in previous modules.

Lessons
- Active Directory Automation Overview
- Managing Users and Groups
- Managing Computers and Other Directory Objects
Lab : Managing Users and Groups
- Retrieving a filtered list of users from Active Directory
- Resetting user passwords and address information
- Disabling users that belong to a specific group
Lab : Managing Computers and Other Directory Objects
- Listing all computers that appear to be running a specific operating system according to Active Directory information
- Creating a report showing all Windows Server 2008 R2 servers
- Managing fine-grained password policies in Active Directory
- Discovering Organizational Units that are not protected against accidental deletion

After completing this module, students will be able to:

- Give an overview of Active Directory automation.

- Manage Users and Groups using Windows PowerShell.

- Manage Computers and Other Active Directory Objects using Windows PowerShell.

Module 6: Windows PowerShell Scripts

This module introduces basic Windows PowerShell scripts that execute a batch of shell commands in a single operation. It points out the security concerns associated with scripting, and how to configure and control the shell's security settings that relate to scripting. It also explains how to write basic scripts that execute batches of commands, and how to parameterize scripts in order to make them more flexible in a variety of situations.

Lessons
- Script Security
- Basic Scripts
- Parameterized Scripts

Lab : Writing Windows PowerShell Scripts
- Executing scripts
- Using positional script parameters
- Using named script parameters

After completing this module, students will be able to:

- Identify security concerns associated with scripting.

- Write basic scripts.

- Write parameterized scripts.

Module 7: Background Jobs and Remote Administration
This module explains how to work with Windows PowerShell's background jobs and remote administration functionality. It shows how to create, monitor, and manage local background jobs, and receive results from completed jobs. It also covers how to configure Windows PowerShell remoting both locally and in a domain environment. This module describes how to create and manage session connections to remote computers, and explain how to use those session connections in one-to-one remote shell instances as well as one-to-many remote command invocation. Finally, it shows how to invoke remote commands as background jobs, and how to manage those jobs and receive results from them.

Lessons
- Working with Background Jobs
- Using Windows PowerShell Remoting

Lab : Working with Background Jobs
- Using background jobs with WMI
- Using background jobs for local computers
- Receiving the results from a completed job
- Removing a completed job
- Waiting for a background job to complete
- Stopping a background job before it completes
- Working with the properties of a job

Lab : Using Windows PowerShell Remoting
- Interactive remoting
- Fan-out remoting
- Fan-out remoting using background jobs
- Saving information from background jobs

After completing this module, students will be able to:

- Work with Background Jobs.

- Use Windows PowerShell Remoting.

Module 8: Advanced Windows PowerShell Tips and Tricks

This module introduces several advanced Windows PowerShell techniques. While these techniques do not contribute directly to any particular business goal, they do enable more efficient use of the shell itself, which leads to more efficient administration and automation. It shows how to use profiles to consistently configure the shell environment, and how to use several techniques for effectively re-using and sharing existing modularized scripts. It also points out best practices and techniques related to script documentation.

Lessons
- Using Profiles
- Re-Using Scripts and Functions
- Writing Comment-Based Help

Lab : Advanced PowerShell Tips and Tricks
- Writing a profile script
- Creating a script module
- Adding help information to a function

After completing this module, students will be able to:

- Use Profiles.

- Re-Use Scripts and Functions.

- Write Comment-Based Help.

Module 9: Automating Windows Server 2008 R2 Administration
This module gives an opportunity to complete several real-world administration tasks related to Windows Server 2008 R2. It provides minimal instruction in how to use the cmdlets and techniques required to accomplish the lab portion of this module; instead, it lets students rely on the skills they have learned in the preceding modules of this course.

Lessons
- Windows Server 2008 R2 Modules Overview
- Server Manager Cmdlets Overview
- Group Policy Cmdlets Overview
- Troubleshooting Pack Overview
- Best Practices Analyzer Cmdlets Overview
- IIS Cmdlets Overview

Lab : Using the Server Manager Cmdlets
- Listing all currently installed features
- Comparing objects
- Installing a new server feature
- Exporting current configuration to XML

Lab : Using the Group Policy Cmdlets
- Listing all the Group Policy Objects in the domain
- Creating a text-based report
- Creating an HTML report
- Backing up all Group Policy Objects

Lab : Using the Troubleshooting Pack Cmdlets
- Importing the Troubleshooting Pack module
- Solving an end-user problem interactively
- Solving a problem using answer files

Lab : Using the Best Practice Analyzer Cmdlets
- Importing the Best Practice module
- Viewing existing models
- Running a Best Practices scan

Lab : Using the IIS Cmdlets
- Importing the IIS module
- Creating a new web site
- Backing up IIS
- Modifying web site bindings
- Using the IIS PSDrive
- Restoring an IIS Configuration

After completing this module, students will be able to:

- Give an overview of Windows Server 2008 Modules.

- Give an overview of Server Manager Cmdlets.

- Give an overview of Group Policy Cmdlets.

- Give an overview of Troubleshooting Pack.

- Give an overview of Best Practices Analyzer Cmdlets.

- Give an overview of IIS Cmdlets.

Module 10: Reviewing and Reusing Windows PowerShell Scripts

One of the core skills administrators need is the ability to take a script that someone else has written, review that script to understand what it does, and identify areas of that script that may need to be modified to run in their environment. Those skills are exactly what this module tries to build.

Lessons
- Example Script Overview
- Understanding Scripts

After completing this module, students will be able to:

- Review a complete, real-world script.

- Develop an expectation of what a script will do by reviewing a script.

Module 11: Writing Your Own Windows PowerShell Scripts This module explains the concepts and techniques related to structured scripting and programming within Windows PowerShell. It describes how to create, manage, and use variables. It introduces the complete "scripting language" of Windows PowerShell, which consists of several programming constructs. This module also covers advanced topics and techniques related to structured programming within Windows PowerShell. It explains how to trap and handle errors that occur during script execution, and also describes the proper techniques and practices for debugging a script that is not executing as expected. It shows how to modularize scripts into a variety of reusable functions, with the ultimate goal of producing a function that mimics the structure of a shell cmdlet.

Lessons
- Variables, Arrays, Escaping, and More Operators
- What is Scope?
- Scripting Constructs
- Error Trapping and Handling
- Debugging Techniques
- Modularization

Lab : Using Variables and Arrays
- Creating variables and interact with them
- Understanding arrays and hashtables

- Using Single- and double-quoted strings and the backtick
- Using Arrays and array lists
- Using 'Contains', 'like', and 'equals' operators

Lab : Using Scripting Constructs

- Processing and validating input
- Working with For, While, ForEach, and Switch
- Exploiting the power of the one-liner

Lab : Error Trapping and Handling

- Retrieving error information
- Handling errors
- Integrating error handling

Lab : Debugging a Script

- Debugging from the Windows PowerShell console
- Debugging using the Windows PowerShell ISE

Lab : Modularization

- Generating an inventory audit report

After completing this module, students will be able to:

- Use Variables, Arrays, Escaping, and More Operators.

- Explain Scope.

- Use Scripting Constructs.

- Trap and handle errors.

- Apply Debugging Techniques.

- Use Modularization.

Microsoft® Partner
Silver Learning