

Automating Windows Server 2008 Administration with Windows PowerShell

Course GWE64340: Three days; Instructor-Led



On This Page

↓ [Introduction](#)

↓ [Audience](#)

↓ [At Course Completion](#)

↓ [Prerequisites](#)

↓ [Course Outline](#)

Introduction

This three-day instructor-led course provides students with the knowledge and skills to utilize Windows PowerShell for administering and automating administration of Windows Server 2008. The course focuses on cmdlets, script structure and flow control, language syntax, and implementation details of scripting administrative tasks using COM, WMI, and .NET foundations.

↑ [Top of page](#)

Audience

This course is intended for Windows administrators interested in automating Windows Server 2008 administration tasks, as well as those people looking for a full-featured interactive command-line environment for Windows operating systems. Windows end users or developers who need to understand what is involved in Windows administration or command-line environments may also find this course helpful.

↑ [Top of page](#)

At Course Completion

After completing this course, students will be able to:

- Install and launch Windows PowerShell
- Work with basic objects in Windows PowerShell, including using cmdlets, data types, variables, and fundamental object-based information models
- Implement sequences of operations by putting them together into a pipeline
- Control the formatting of the resultant set of objects that are emitted at the end of a pipeline
- Implement sequences of operations by putting them together into a script
- Implement flow control within scripts and define functions and filters to help modularize complex scripts
- Manipulate files and registry values
- Manage disk storage volumes, shadow copies, shared folders, Terminal Services and IIS properties using WMI in Windows PowerShell
- Administer and maintain Active Directory directory services and IIS 7.0 Web sites using Windows

PowerShell

- Maintain Group Policy using Windows PowerShell

[↑Top of page](#)

Prerequisites

Before attending this course, students should have completed:

- Planning and Administering Windows Server 2008 Servers, or have equivalent knowledge of administrative tasks.

In addition, it is recommended, but not required, that students have completed:

- Microsoft Visual Basic Scripting Edition and Microsoft Windows Script Host Essentials, or have equivalent knowledge of scripting and automation in Windows.

-

[↑Top of page](#)

Course Outline

Module 1: Introduction to Microsoft Windows PowerShell

This module explains how to check your system for prerequisites for Windows PowerShell, use Server Manager to install Windows PowerShell architecture, confirm installation, and use Windows PowerShell commands to customize the Windows PowerShell environment.

Lessons

- Introduction to Windows PowerShell
- Installing Windows PowerShell in Windows Server 2008

Lab 1: Implementing Windows PowerShell

- Installing Windows PowerShell
- Customizing Windows PowerShell

After completing this module, students will be able to:

- Describe the architecture, platforms, and prerequisites of the Windows PowerShell environment
- Install Windows PowerShell using the Windows Server 2008 Server Manager

Module 2: Overview of Microsoft Windows PowerShell

This module explains basic concepts in Windows PowerShell, including objects, variables, cmdlets, and pipelines. It describes how to invoke available cmdlets and aliases, assign aliases. The module also includes demonstrations of tab expansion and basic operators.

Lessons

- Overview of Objects
- Working with Cmdlets
- Tab Expansion, Aliases, and History
- Using Variables and Types

Lab 1: Working with Windows PowerShell Cmdlets, Aliases, Objects, and Variables

- Learning Cmdlets and Defining Aliases
- Holding the Output of a Cmdlet

After completing this module, students will be able to:

- Explain the fundamental relationship between information and operations that are bundled together into

various classes of objects

- Use the Windows PowerShell cmdlets Get-Command and Get-Help to obtain information about other cmdlets and their parameters.
- Use tab expansion, aliases, and history in Windows PowerShell to get more done with less typing
- Perform basic numeric and string operations using Windows PowerShell, including holding temporary values in variables

Module 3: Building Pipelines for Assembly-Line Style Processing

This module explains how to use a pipeline to connect the output of one cmdlet to the input of another, reorder objects, and filter objects based on specific properties. Arrays and their uses are also discussed.

Lessons

- Using Pipelines
- Using Arrays
- Filtering and Iterating Through the Pipeline
- Reordering Objects in a Pipeline

Lab 1: Implementing Pipelines in Windows PowerShell

- Evaluating Process Properties Using the Get-Member Cmdlet
- Calculating Process Memory Usage
- Using Associative Array Variables
- Sorting and Selecting Elements from a Resultant Set of Data

After completing this module, students will be able to:

- Connect the output of one cmdlet to the input of another cmdlet as a method of building sequences of processing relationships toward a goal
- Define arrays of data and hold cmdlet and pipeline results in an array
- Filter objects that are flowing through a pipeline by using cmdlets such as Where-Object
- Reorder objects and choose specific properties to filter objects that are coming down a pipeline by using the Sort-Object cmdlet and Select-Object cmdlets

Module 4: Managing Processes and Formatting Cmdlet Output

This module explains how to choose a format in which to present data that is appropriate to the data set, format specific process properties, such as memory usage or CPU time, and use custom formatting. It also describes how you can view, start, and stop processes and services.

Lessons

- Managing Windows Processes with Microsoft Windows PowerShell
- Formatting Cmdlet Output

Lab 1: Output Formatting and Process Control with Windows PowerShell

- Implementing Basic Formatting Control
- Formatting with the -f Operator
- Implementing Advanced Formatting

After completing this module, students will be able to:

- Monitor and control services and processes running on Windows operating systems
- Present information with specific formatting through the use of formatting operators and cmdlets

Module 5: Introduction to Scripting with Microsoft Windows PowerShell

This module explains how to write and modify scripts to perform a sequence of cmdlets. Security and working with credentials are also discussed.

Lessons

- Writing Windows PowerShell Scripts
- Script Parameters
- Security in Windows PowerShell
- Customizing Windows PowerShell with Profiles

Lab 1: Implementing Scripts in Windows PowerShell

- Writing and Running a Script
- Customizing Profiles

After completing this module, students will be able to:

- Design, write, and test sequences of operations and cmdlets using sequences, variables, and pipelines
- Use parameters to pass additional data to a script in a structured way
- Establish security with adequate execution policy and script signing
- Customize profile files and describe the scope of profile files

Module 6: Implementing Flow Control and Functions

This module explains how to move scripts into functions and add functions to profiles. Flow of execution based on a common input, iterating in general and iterating through an array or collection are also discussed.

Lessons

- Controlling the Flow of Execution Within Scripts
- Iteration Flow Control
- Developing and Using Functions

Lab 1: Implementing Functions and Flow Control in Windows PowerShell

- Adding Flow Control in a Script
- Creating Functions

After completing this module, students will be able to:

- Use Windows PowerShell flow control language features to implement choices in scripts
- Use Windows PowerShell flow control language features to implement repetition in scripts
- Define functions to encapsulate a sequence of operations

Module 7: Working with Files, the Registry, and Certificate Stores

This module explains how to write scripts that perform specific tasks, such as searching files for particular text and modifying all matching files, or searching the event logs for events that match specific criteria. It also describes how to access data stores, the file store, the registry, certificate stores, and other stores, use wildcards and regular expressions, and import and export aliases and objects.

Lessons

- Using Data Stores
- Using Providers
- Filtering and Selecting with Regular Expressions
- Implementing Event Log Management
- Persisting Objects in Files

Lab 1: Working with Files, the Registry, and Certificate Stores

- Searching for Certain Files
- Modifying Registry Entries
- Generating Reports
- Generating Reports on the Security Log
- Comparing Files

After completing this module, students will be able to:

- Use providers and cmdlets to access folders and files
- Use providers and cmdlets to access registry keys and values, and public key certificate stores and certificates
- Filter the set of files, values, or certificates with which to work, based on patterns in their attributes or content
- Implement filtering techniques when using the Get-EventLog cmdlet
- Move aliases and objects from Windows PowerShell memory (that is, RAM) in and out of files

Module 8: Managing the Windows Operating System Using Microsoft Windows PowerShell and WMI

This module explains how to use WMI to access system features, enumerate, defragment, and mount disk volumes in Windows PowerShell. Listing and configuring volume shadow copies, listing and creating shared folders with WMI, and configuring Terminal Services and IIS properties are also discussed.

Lessons

- Introduction to WMI and WMI Objects
- Managing Disks and Disk Volumes Using Windows PowerShell with WMI
- Managing Shadow Copies Using Windows PowerShell with WMI
- Managing Shared Folders with Windows PowerShell
- Managing Terminal Services with WMI
- Managing IIS 7.0 with WMI

Lab 1: Managing the Windows Operating System with Windows PowerShell and WMI

- Using WMI Classes in Windows PowerShell
- Using WMI Type Accelerators
- Managing Disk Volumes in Windows PowerShell
- Defragmenting Disk Volumes Using Windows PowerShell
- Managing IIS 7.0 Properties Using WMI

After completing this module, students will be able to:

- Use Get-WMIObject to retrieve WMI data from a local or remote system
- Perform some common administrative tasks using Windows PowerShell and WMI
- Manage volume shadow copies using Windows PowerShell
- Manage shared folders using Windows PowerShell
- Configure Terminal Services via WMI in Windows PowerShell
- Administer IIS 7.0 with Windows PowerShell

Module 9: Administering Active Directory with Microsoft Windows PowerShell

This module explains how to write scripts to perform Active Directory administration tasks such as changing the domain functional level, moving FSMO roles, and creating and modifying objects such as groups and user accounts. Managing relationships between user accounts and groups is also demonstrated.

Lessons

- Administering Domains and Forests Using .NET Objects
- Managing User Accounts and Groups Using ADSI
- Managing Relationships Between Users and Groups
- Web Administration Using IIS 7.0

Lab 1: Administering Active Directory with Windows PowerShell

- Managing Active Directory Domain and Forest Properties
- Maintaining Active Directory with ADSI
- Maintaining Relationships in Active Directory with ADSI
- Managing IIS 7.0 with the .NET Web.Administration.ServerManager Class

After completing this module, students will be able to:

- Administer Active Directory domain and forest roles and functionality using Windows PowerShell with .NET objects
- Manage Active Directory–based user accounts and groups using the ADSI in Windows PowerShell
- Manage relationships between user accounts and groups in Active Directory
- Administer IIS 7.0 with Windows PowerShell

Module 10: Administering Group Policy in Microsoft Windows PowerShell Using COM

This module explains how to write scripts to perform Active Directory administration tasks such as changing the domain functional level, moving FSMO roles, and creating and modifying objects such as groups and user accounts. Managing relationships between user accounts and groups is also demonstrated.

Lessons

- Managing GPOs Using the GPMC COM Interface
- Managing Group Policy Objects
- Reporting Group Policy

Lab 1: Administering Group Policy in Microsoft Windows PowerShell

- Retrieving a GPO by Using a COM Object
- Copying Group Policy Settings
- Backing Up and Restoring a GPO
- Generating Group Policy Reports

After completing this module, students will be able to:

- Manage GPOs in an Active Directory environment using Windows PowerShell
- Search, back up, and restore Group Policy Objects (GPOs) using Windows PowerShell
- Generate reports of Group Policy in Windows PowerShell