

// Step by step

Windows PowerShell

Third Edition



Ed Wilson

Windows PowerShell Step by Step, Third Edition

Ed Wilson

Windows PowerShell Step by Step

Table of Contents

Cover

Title Page

Copyright Page

Contents at a glance

Contents

Introduction

Chapter 1 Overview of Windows PowerShell 5.0

- Understanding Windows PowerShell

 - Using cmdlets

 - Installing Windows PowerShell

 - Deploying Windows PowerShell to down-level operating systems

- Using command-line utilities

- Security issues with Windows PowerShell

 - Controlling execution of Windows PowerShell cmdlets

 - Confirming actions

 - Suspending confirmation of cmdlets

- Working with Windows PowerShell

 - Accessing Windows PowerShell

 - Configuring the Windows PowerShell console

- Supplying options for cmdlets

- Working with the help options

- Exploring commands: Step-by-step exercises

Table of Contents

Chapter 1 quick reference

Chapter 2 Using Windows PowerShell cmdlets

Understanding the basics of cmdlets

Using the Get-ChildItem cmdlet

Obtaining a directory listing

Formatting a directory listing by using the Format-List cmdlet

Using the Format-Wide cmdlet

Formatting a directory listing by using Format-Table

Formatting output with Out-GridView

Taking advantage of the power of Get-Command

Searching for cmdlets by using wildcard characters

Using the Get-Member cmdlet

Using the Get-Member cmdlet to examine properties and methods

Using the New-Object cmdlet

Creating and using the wshShell object

Using the Show-Command cmdlet

Windows PowerShell cmdlet naming helps you learn

Windows PowerShell verb grouping

Windows PowerShell verb distribution

Creating a Windows PowerShell profile

Working with cmdlets: Step-by-step exercises

Chapter 2 quick reference

Chapter 3 Understanding and using Windows PowerShell providers

Understanding Windows PowerShell providers

Understanding the alias provider

Understanding the certificate provider

Table of Contents

Understanding the environment provider

Understanding the filesystem provider

Understanding the function provider

Using the registry provider to manage the Windows registry

The two registry drives

The short way to create a new registry key

Dealing with a missing registry property

Understanding the variable provider

Exploring Windows PowerShell providers: Step-by-step exercises

Chapter 3 quick reference

Chapter 4 Using Windows PowerShell remoting and jobs

Understanding Windows PowerShell remoting

Classic remoting

WinRM

Using Windows PowerShell jobs

Using Windows PowerShell remoting and jobs: Step-by-step exercises

Chapter 4 quick reference

Chapter 5 Using Windows PowerShell scripts

Why write Windows PowerShell scripts?

The fundamentals of scripting

Running Windows PowerShell scripts

Turning on Windows PowerShell scripting support

Transitioning from command line to script

Manually running Windows PowerShell scripts

Understanding variables and constants

Using the While statement

Constructing the While statement in Windows PowerShell

Table of Contents

A practical example of using the While statement

Using special features of Windows PowerShell

Using the Do...While statement

Using the range operator

Operating over an array

Casting to ASCII values

Using the Do Until statement

Comparing the Windows PowerShell Do...Until statement with VBScript

Using the Windows PowerShell Do statement

The For statement

Using the For statement

Using the Foreach statement

Exiting the Foreach statement early

Using the If statement

Using assignment and comparison operators

Evaluating multiple conditions

The Switch statement

Using the Switch statement

Controlling matching behavior

Creating multiple folders: Step-by-step exercises

Chapter 5 quick reference

Chapter 6 Working with functions

Understanding functions

Using functions to provide ease of code reuse

Including functions in the Windows PowerShell environment

Using dot-sourcing

Using dot-sourced functions

Adding help for functions

Table of Contents

Using a here-string object for help

Using two input parameters

Using a type constraint in a function

Using more than two input parameters

Using functions to encapsulate business logic

Using functions to provide ease of modification

Understanding filters

Creating a function: Step-by-step exercises

Chapter 6 quick reference

Chapter 7 Creating advanced functions and modules

The [cmdletbinding] attribute

Easy verbose messages

Automatic parameter checks

Adding support for the -WhatIf switch parameter

Adding support for the -Confirm switch parameter

Specifying the default parameter set

The Parameter attribute

The Mandatory parameter property

The Position parameter property

The ParameterSetName parameter property

The ValueFromPipeline property

The HelpMessage property

Understanding modules

Locating and loading modules

Installing modules

Creating a module

Creating an advanced function and installing a module: Step-by-step

Table of Contents

exercises

Chapter 7 quick reference

Chapter 8 Using the Windows PowerShell ISE

Running the Windows PowerShell ISE

 Navigating the Windows PowerShell ISE

 Working with the script pane

 Using tab expansion and IntelliSense

Working with Windows PowerShell ISE snippets

 Using Windows PowerShell ISE snippets to create code

 Creating new Windows PowerShell ISE snippets

 Removing user-defined Windows PowerShell ISE snippets

Using the Commands add-on and snippets: Step-by-step exercises

Chapter 8 quick reference

Chapter 9 Working with Windows PowerShell profiles

Six different Windows PowerShell profiles

 Understanding the six Windows PowerShell profiles

 Examining the \$profile variable

 Determining whether a specific profile exists

 Creating a new profile

Design considerations for profiles

 Using one or more profiles

 Using the All Users, All Hosts profile

 Using your own file

Grouping similar functionality into a module

 Where to store the profile module

Creating and adding functionality to a profile: Step-by-step exercises

Chapter 9 quick reference

Table of Contents

Chapter 10 Using WMI

- Understanding the WMI model
- Working with objects and namespaces
- Listing WMI providers
- Working with WMI classes
- Querying WMI
- Obtaining service information: Step-by-step exercises
- Chapter 10 quick reference

Chapter 11 Querying WMI

- Alternate ways to connect to WMI
 - Returning selective data from all instances
 - Selecting multiple properties
 - Choosing specific instances
 - Using an operator
 - Shortening the syntax
- Working with software: Step-by-step exercises
- Chapter 11 quick reference

Chapter 12 Remoting WMI

- Using WMI against remote systems
 - Supplying alternate credentials for the remote connection
 - Using Windows PowerShell remoting to run WMI
 - Using CIM classes to query WMI classes
- Working with remote results
 - Reducing data via Windows PowerShell parameters
 - Reducing data via WQL query
- Running WMI jobs
- Using Windows PowerShell remoting and WMI: Step-by-step exercises

Table of Contents

Chapter 12 quick reference

Chapter 13 Calling WMI methods on WMI classes

Using WMI cmdlets to execute instance methods

- Using the Terminate method directly

- Using the Invoke-WmiMethod cmdlet

- Using the [wmi] type accelerator

Using WMI cmdlets to work with static methods

Executing instance methods: Step-by-step exercises

Chapter 13 quick reference

Chapter 14 Using the CIM cmdlets

Using the CIM cmdlets to explore WMI classes

- Using the Get-CimClass cmdlet and the -ClassName parameter

- Finding WMI class methods

- Filtering classes by qualifier

Retrieving WMI instances

- Reducing returned properties and instances

- Cleaning up output from the command

Working with associations

Retrieving WMI instances: Step-by-step exercises

Chapter 14 quick reference

Chapter 15 Working with Active Directory

Creating objects in Active Directory

- Creating an OU

- ADSI providers

- LDAP names

Creating users

- What is user account control?

Table of Contents

Working with users

Creating multiple OUs: Step-by-step exercises

Chapter 15 quick reference

Chapter 16 Working with the AD DS module

Understanding the Active Directory module

Installing the Active Directory module

Getting started with the Active Directory module

Using the Active Directory module

Finding the FSMO role holders

Discovering Active Directory

Renaming Active Directory sites

Managing users

Creating a user

Finding and unlocking Active Directory user accounts

Finding disabled users

Finding unused user accounts

Updating Active Directory objects: Step-by-step exercises

Chapter 16 quick reference

Chapter 17 Deploying Active Directory by using Windows PowerShell

Using the Active Directory module to deploy a new forest

Adding a new domain controller to an existing domain

Adding a read-only domain controller

Installing domain controller prerequisites and adding to a forest:
Step-by-step exercises

Chapter 17 quick reference

Chapter 18 Debugging scripts

Table of Contents

Understanding debugging in Windows PowerShell

- Understanding the three different types of errors

Using the Set-PSDebug cmdlet

- Tracing the script

- Stepping through the script

- Enabling strict mode

Debugging the script

- Setting breakpoints

- Setting a breakpoint on a line number

- Setting a breakpoint on a variable

- Setting a breakpoint on a command

- Responding to breakpoints

- Listing breakpoints

- Enabling and disabling breakpoints

- Deleting breakpoints

Debugging a function: Step-by-step exercises

Chapter 18 quick reference

Chapter 19 Handling errors

Handling missing parameters

- Creating a default value for a parameter

- Making the parameter mandatory

Limiting choices

- Using PromptForChoice to limit selections

- Using Test-Connection to identify computer connectivity

- Using the -contains operator to examine the contents of an array

- Using the -contains operator to test for properties

Handling missing rights

- Using an attempt-and-fail approach

Table of Contents

- Checking for rights and exiting gracefully

- Handling missing WMI providers

- Handling incorrect data types

- Handling out-of-bounds errors

 - Using a boundary-checking function

 - Placing limits on the parameter

- Using Try...Catch...Finally

 - Catching multiple errors

- Using PromptForChoice to limit selections and using Try...Catch...Finally:

 - Step-by-step exercises

- Chapter 19 quick reference

Chapter 20 Using the Windows PowerShell workflow

- Why use workflows?

 - Workflow requirements

 - A simple workflow

- Parallel PowerShell

- Workflow activities

 - Windows PowerShell cmdlets as activities

 - Disallowed core cmdlets

 - Non-automatic cmdlet activities

 - Parallel activities

- Checkpointing Windows PowerShell workflow

 - Understanding checkpoints

 - Placing checkpoints

 - Adding checkpoints

- Adding a sequence activity to a workflow

- Creating a workflow and adding checkpoints: Step-by-step exercises

Table of Contents

Chapter 20 quick reference

Chapter 21 Managing Windows PowerShell DSC

Understanding Desired State Configuration

The DSC process

Configuration parameters

Setting dependencies

Controlling configuration drift

Modifying environment variables

Creating a DSC configuration and adding a dependency: Step-by-step exercises

Chapter 21 quick reference

Chapter 22 Using the PowerShell Gallery

Exploring the PowerShell Gallery

Configuring and using PowerShell Get

Installing a module from the PowerShell Gallery

Configuring trusted installation locations

Uninstalling a module

Searching for and installing modules from the PowerShell Gallery: Step-by-step exercises

Chapter 22 quick reference

Appendix A: Windows PowerShell scripting best practices

Appendix B: Regular expressions quick reference

Index

About the author

Survey