



Advanced Automated Administration with Windows PowerShell

Course 55318: 3 days; Instructor-Led

Introduction

55318A is the updated version of the retired legacy MOC course 10962C. This three-day course teaches students how to automate administrative tasks using PowerShell. Students will learn crucial scripting skills such as creating advanced functions, writing controller scripts, and handling script errors. Candidates will learn how to use PowerShell when working with Microsoft Azure, SQL Server, Active Directory, IIS, Windows PowerShell Workflow, .NET resources, the REST API and XML, CSV & JSON formatted data files.

At Course Completion

After completing this course, students will be able to:

- Create advanced functions.
- Use Microsoft .NET Framework and REST API in Windows PowerShell.
- Handle script errors.
- Use XML, JSON, and custom formatted data.
- Manage Microsoft Azure resources
- Analyze and debug scripts
- Understand Windows PowerShell workflow.

Audience

This course is intended for IT professionals who are experienced in general Windows Server and Windows Client administration. Students should have a foundational knowledge of Windows PowerShell, which they can obtain by taking course 10961C: Automating Administration with Windows PowerShell.

In addition, this course provides scripting guidance for Microsoft Azure administrators and developers who support development environments and deployment processes.

Prerequisites

- Experience as a Windows System Administrator
- Prior scripting or programming experience
- All the prerequisites can be met by successfully taking the 10961 – Automating Administration with Windows PowerShell course

Course Outline

Module 1: Creating advanced functions

This module explains how to parameterize a command into an advanced function.

Lessons

- Lesson 1: Converting a command into an advanced function
- Lesson 2: Creating a script module
- Lesson 3: Defining parameter attributes and input validation
- Lesson 4: Writing functions that accept pipeline input
- Lesson 5: Producing complex pipeline output
- Lesson 6: Using comment-based Help
- Lesson 7: Using Whatif and Confirm parameters

Lab 1: Creating advanced functions

- Exercise 1: Converting a command into a function
- Exercise 2: Creating a script module
- Exercise 3: Defining parameter attributes

- Exercise 4: Writing functions to accept pipeline input
- Exercise 5: Producing complex function output
- Exercise 6: Using comment-based Help
- Exercise 7: Using Whatif and Confirm parameters

After completing this module, students will be able to:

- Convert a command into an advanced function.
- Create a script module.
- Define parameter attributes and input validation.
- Write functions that accept pipeline input.
- Produce complex pipeline output.
- Document functions by using comment-based help.
- Support -WhatIf and -Confirm.

Module 2: Using Microsoft .NET Framework and REST API in Windows PowerShell

This module explains how to use Microsoft .NET Framework and REST API to supplement functionality that Windows PowerShell commands provide.

Lessons

- Lesson 1: Using .NET Framework in PowerShell
- Lesson 2: Using REST API in PowerShell

Lab 1: Using Microsoft .NET Framework and REST API in Windows PowerShell

- Exercise 1: Using .NET Framework in PowerShell
- Exercise 2: Using REST API in PowerShell

After completing this module, students will be able to:

- Use Microsoft .NET Framework in Windows PowerShell.
- Use REST API in Windows PowerShell.

Module 3: Writing controller scripts

This module explains how to combine advanced functions that perform a specific task and a controller script that provides a user interface or automates a business process.

Lessons

- Lesson 1: Understanding controller scripts
- Lesson 2: Writing controller scripts with a user interface
- Lesson 3: Writing controller scripts that create reports

Lab 1: Writing controller scripts

- Exercise 1: Understanding controller scripts
- Exercise 2: Writing controller scripts with a user interface
- Exercise 3: Writing controller scripts that create reports

After completing this module, students will be able to:

- Describe controller scripts.
- Write controller scripts that display a user interface.
- Write controller scripts that produce reports.

Module 4: Handling script errors

This module explains how to perform basic error handling in scripts.

Lessons

- Lesson 1: Understanding error handling
- Lesson 2: Handling errors in a script

Lab 1: Handling script errors

- Exercise 1: Understanding error handling
- Exercise 2: Handling errors in a script

After completing this module, students will be able to:

- Describe error handling.
- Handle errors and timeouts in a script.

Module 5: Using XML, JSON, and custom-formatted data

This module explains how to read, manipulate, and write data in XML and JSON format.

Lessons

- Lesson 1: Working with XML formatted data
- Lesson 2: Working with JSON formatted data
- Lesson 3: Working with custom-formatted data

Lab 1: Working with XML and JSON formatted data

- Exercise 1: Working with XML formatted data
- Exercise 2: Working with JSON formatted data
- Exercise 3: Working with custom-formatted data
- Exercise 4: Using PowerShell Core (Optional)

After completing this module, students will be able to:

- Read, manipulate, and write XML-formatted data
- Read, manipulate, and write JSON-formatted data
- Read and manipulate custom-formatted data

Module 6: Enhancing server management with Desired State Configuration and Just Enough Administration

This module explains how to write Desired State Configuration (DSC) configuration files, deploy those files to servers, and monitor server's configurations. This module also explains how to restrict administrative access with Just Enough Administration (JEA).

Lessons

- Lesson 1: Implementing Desired State Configuration
- Lesson 2: Implementing Just Enough Administration

Lab 1: Creating and deploying a DSC configuration

- Exercise 1: Implementing Desired State Configuration
- Exercise 2: Implementing Just Enough Administration

After completing this module, students will be able to:

- Understand DSC.
- Create and deploy a DSC configuration.
- Implement JEA.

Module 7: Analyzing and debugging scripts

This module explains how to use native Windows PowerShell features to analyze and debug existing scripts

Lessons

- Lesson 1: Debugging in Windows PowerShell
- Lesson 2: Analyzing and debugging an existing script

Lab 1: Analyzing and debugging scripts

- Exercise 1: Debugging in Windows PowerShell
- Exercise 2: Analyzing and debugging an existing script
- After completing this module, students will be able to:
- Debug in Windows PowerShell.
- Analyze and debug an existing script.

Module 8: Understanding Windows PowerShell Workflow

- This module explains the features of the Windows PowerShell Workflow technology.

Lessons

- Lesson 1: Understanding Windows PowerShell Workflows
- Lesson 2: Running Windows PowerShell Workflows

Lab 1: Create and run Windows PowerShell Workflows

- Exercise 1: Understanding Windows PowerShell Workflows
- Exercise 2: Running Windows PowerShell Workflows

After completing this module, students will be able to:

- Describe and implement the Workflow feature of Windows PowerShell.