

# Advanced Automated Administration with Windows PowerShell

MOC 10962

In diesem 3-tägigen Seminar lernen Sie, wie Sie alltägliche Management und Administrations-Aufgaben und Funktionen in Ihrer Windows Server Infrastruktur automatisieren und rationalisieren. Dieser Kurs baut auf den Kurs 10961: Automating Administration with Windows PowerShell auf. Er basiert auf Windows Server 2012 R2 und Windows 8.1 und behandelt vordergründig Windows PowerShell v4.0, ist allerdings auch relevant für die v2.0 and v3.0 Windows PowerShell Umgebungen.

Der Fokus liegt darauf, noch besser skalierbare und anwendbare Windows PowerShell Skripts für den Gebrauch in Ihrem Unternehmen durch den Aufbau Ihrer eigenen Windows PowerShell Tools zu erstellen. Sie erfahren mehr über das Erstellen erweiterter Funktionen, Skript-Module, erweiterter Parameter Attribute und Controller Skripte. Sie erfahren, wie Sie Ihre Skripte robuster machen, indem Sie das Beheben von Skript-Fehlern und die Analyse und das Debugging von Windows PowerShell Skripten lernen. In diesem Kurs werden ebenfalls Windows PowerShell cmdlets with .NET Framework und die Konfiguration Ihrer Windows Server mithilfe der Desired State Configuration vermittelt, sowie der Aufbau und das Verständnis des Windows PowerShell Workflows behandelt.

In detaillierten Hands-on Labs und durch Inhalte, die in die Tiefe gehen, lernen Sie, manuelle Aufgaben, die Sie als Administrator vermutlich täglich erledigen, durch automatisierte, wiederkehrende, und genau gemanagte und provisionierte Prozesse Ihrer Windows Serverinfrastruktur mithilfe von Windows PowerShell Tools zu ersetzen.

## Zielgruppe

Diese Schulung eignet sich für IT Professionals, die bereits grundlegende Erfahrung mit der Administration von Windows Servern und Windows Clients verfügen oder die bereits Erfahrungen in der Administration und dem Support von Application Servern und Services inklusive Applikationen wie Exchange, SharePoint und SQL verfügen.

Der Kurs ist ebenfalls interessant für System-, Infrastruktur- und Applikations-Administratoren, die in einer Windows- oder einer hybriden Windows Umgebung arbeiten.

Der Kurs eignet sich ebenfalls für IT Professionals, die ihre vorhandenen PowerShell Kenntnisse ausbauen möchten und lernen möchten, wie sie ihre eigenen PowerShell Tools erstellen, um sie in ihrem Unternehmen einzusetzen. Interessant sind Unternehmen, welche jegliche Microsoft-Produkte oder Produkte unabhängiger Softwareanbieter (ISV) einsetzen, die die PowerShell-Verwaltung unterstützen..

### **Kerninhalte**

- Erstellen von Erweiterten Funktionen
- Verwenden von Cmdlets und Microsoft .NET Framework in Windows PowerShell
- Schreiben von Controller Skripten
- Handling von Skript Fehlern
- Verwenden von XML Datendateien
- Verwalten von Server-Konfigurationen unter der Verwendung der Desired State Configuration
- Analysieren und Debugging von Skripten
- Verstehen des Windows PowerShell Workflows

### **Voraussetzungen**

- Es empfehlen sich Kenntnisse und Erfahrungen in der Arbeit mit Windows PowerShell oder vergleichbare Kenntnisse der im Kurs MOC 10961B: Automating Administration with Windows PowerShell vermittelter Inhalte.
- Grundlegende Erfahrungen in der Administration von Windows Servern und Windows Clients oder Erfahrungen in der Administration und dem Support von Application Servern und Services inklusive Applikationen wie Exchange, SharePoint und SQL.

### **Dauer**

3 Tage

### **Hinweis**

Für diesen Kurs können Sie Voucher aus dem Microsoft Software Assurance Programm (SA-Voucher) einlösen. Dafür erhalten Sie pro Voucher einen kostenfreien Trainingstag.

## Kursmodule

### Module 1: Creating Advanced Functions

In this module students will learn how to parameterize a command into an advanced function. It is designed to teach several key principles in a single logical sequence, by using frequent hands-on exercises to reinforce new skills.

#### Lessons

- Converting a Command into an Advanced Function
- Creating a Script Module
- Defining Parameter Attributes and Input Validation
- Writing Functions that use Multiple Objects
- Writing Functions that Accept Pipeline Input
- Producing Complex Function Output
- Documenting Functions by using Content-Based Help
- Supporting -Whatif and -Confirm

#### Lab: Converting a Command into an Advanced Function

- Converting a Command into an Advanced Function

#### Lab: Creating a Script Module

- Creating a Script Module

#### Lab: Defining Parameter Attributes and Input Validation

- Defining Parameter Attributes and Input Validation

#### Lab: Writing Functions that use Multiple Objects

- Writing Functions that use Multiple Objects

#### Lab: Writing Functions that Accept Pipeline Input

- Writing Functions that Accept Pipeline Input

#### Lab: Producing Complex Function Output

- Producing Complex Function Output

#### Lab: Documenting Functions by using Content-Based Help

- Documenting Functions by using Content-Based Help

#### Lab: Supporting -Whatif and -Confirm

- Supporting -Whatif and -Confirm

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 use Multiple Objects
- Writing Functions that Accept Pipeline Input
- Produce Complex Function Output
- Document Functions by using Content-Based Help
- Support -Whatif and -Confirm

## **Module 2: Using Cmdlets and Microsoft .NET Framework in Windows PowerShell**

Windows PowerShell provides commands that accomplish many of the tasks that you will need in a production environment. Sometimes, a command is not available but the .NET Framework provides an alternate means of accomplishing a task. Because Windows PowerShell is built on the .NET Framework, it is able to access those alternate means. In this module, you will learn how to discover and run Windows PowerShell commands, and how to use .NET Framework components from inside Windows PowerShell. These two techniques will provide you with the most flexibility and capability for accomplishing tasks in a production environment.

### **Lessons**

- Running Windows PowerShell Commands
- Using Microsoft .NET Framework in Windows PowerShell

### **Lab: Using .NET Framework in Windows PowerShell**

- Using Static and Instance Members

After completing this module, students will be able to:

- Run Windows PowerShell commands
- Use Microsoft .NET Framework in Windows PowerShell

## **Module 3: Writing Controller Scripts**

In this module, students will learn how to combine tools – advanced functions that perform a specific task – and a controller script that provides a user interface or automates a business process.

### **Lessons**

- Understanding Controller Scripts
- Writing Controller Scripts that Show a User Interface

**Lab: Writing Controller Scripts that Display a User Interface**

- Write Functions to be Used in the Controller Script
- Write a Controller Script that Implements a Text-Based Menu

After completing this module, students will be able to:

- Understand Controller Scripts
- Write Controller Scripts that show a User Interface

**Module 4: Handling Script Errors**

In this module, students will learn how to perform basic error handling in scripts. The focus will be about how to add error handling to existing tools, primarily as a time-saving mechanism (instead of having students write new tools). A side benefit of this approach is that it will help build the skills that you must have to analyze and reuse existing code written by someone else.

**Lessons**

- Understanding Error Handling
- Handling Errors in a Script

**Lab: Handling Errors in a Script**

- Handling Errors in a Script

After completing this module, students will be able to:

- Understand Error Handling
- Handle Errors in a Script

**Module 5: Using XML Data Files**

In this module, students will learn how to read, manipulate, and write data in XML files. XML files provide a robust, yet straightforward way to store both flat and hierarchical data. XML files are more flexible than CSV, more accessible for small amounts of data than SQL Server, and easier to code against than Excel automation.

**Lessons**

- Reading, Manipulating and Writing Data in XML

**Lab: Reading, Manipulating and Writing Data in XML**

- Test the Provided Tools
- Update an XML Inventory Document

After completing this module, students will be able to:

- Reading, Manipulating and Writing Data in XML

## **Module 6: Managing Server Configurations by Using Desired State Configuration**

In this module, students will learn how to write Desired State Configuration (DSC) configuration files, deploy those files to servers, and monitor servers' configurations.

### **Lessons**

- Understanding Desired State Configuration
- Creating and Deploying a DSC Configuration

### **Lab: Creating and Deploying a DSC Configuration**

- Write, Run and Push a DSC Configuration

After completing this module, students will be able to:

- Understand Desired State Configuration
- Create and Deploy a DSC Configuration

## **Module 7: Analyzing and Debugging Scripts**

In this module, students will learn how to use native Windows PowerShell features to analyze and debug existing scripts. These skills are also useful when students have to debug their own scripts.

### **Lessons**

- Debugging in Windows PowerShell
- Analyzing and Debugging an Existing Script

### **Lab: Analyzing and Debugging an Existing Script**

- Analyzing and Debugging an Existing Script

After completing this module, students will be able to:

- Analyze and Debug a script

## **Module 8: Understanding Windows PowerShell Workflow**

In this module, students will learn about the features of the Windows PowerShell Workflow technology.

### **Lessons**

- Understanding Windows PowerShell Workflow

After completing this module, students will be able to:

- Understanding Windows PowerShell Workflow