Windows PowerShell Scripting and Toolmaking (55039)

Description

This course is intended for IT professionals who are interested in furthering their skills in Windows PowerShell and administrative automation. The course assumes a basic working knowledge of PowerShell as an interactive command-line shell, and teaches students the correct patterns and practices for building reusable, tightly scoped units of automation.

Niveau
Intermédiaire
Course Content
Module 1: Tool Design

- Lesson 1: Tools do one thing
- Lesson 2: Tools are flexible
- Lesson 3: Tools look native

Module 2: Start with a Command

- Lesson 1: Why start with a command?
- Lesson 2: Discovery and experimentation

Module 3: Build a Basic Function and Module

- Lesson 1: Start with a basic function
- Lesson 2: Create a script module
- Lesson 3: Check prerequisites
- Lesson 4: Run the new command

Module 4: Adding CmdletBinding and Parameterizing

- Lesson 1: About CmdletBinding and common parameters
- Lesson 2: Accepting pipeline input
- Lesson 3: Mandatory-ness
- Lesson 4: Parameter validation
- Lesson 5: Parmeter aliases

Module 5: Emitting Objects as Output

- Lesson 1: Assembling information
- Lesson 2: Constructing and emitting output
- Lesson 3: Quick tests

Module 6: Interlude - Changing Your Approach

- Lesson 1: Examining a script
- Lesson 2: Critiquing a script
- Lesson 3: Revising the script

Module 7: Using Verbose, Warning, and Informational Output

- Lesson 1: Knowing the six channels
- Lesson 2: Adding verbose and warning output
- Lesson 3: Doing more with verbose output
- Lesson 4: Informational output

Module 8: Comment-Based Help

- Lesson 1: Where to put your help
- Lesson 2: Getting started
- Lesson 3: Going further with comment-based help
- Lesson 4: Broken help

Module 9: Handling Errors

- Lesson 1: Understanding errors and exceptions
- Lesson 2: Bad handling
- Lesson 3: Two reasons for exception handling
- Lesson 4: Handling exceptions in our tool
- Lesson 5: Capturing the actual exception
- Lesson 6: Handling exceptions for non-commands
- Lesson 7: Going further with exception handling
- Lesson 8: Deprecated exception handling

Module 10: Basic Debugging

- Lesson 1: Two kinds of bugs
- Lesson 2: The ultimate goal of debugging
- Lesson 3: Developing assumptions
- Lesson 4: Write-Debug
- Lesson 5: Set-PSBreakpoint
- Lesson 6: The PowerShell ISE

Module 11: Going Deeper with Parameters

- Lesson 1: Parameter positions
- Lesson 2: Validation
- Lesson 3: Multiple parameter sets
- Lesson 4: Value from remaining arguments
- Lesson 5: Help messages
- Lesson 6: Aliases
- Lesson 7: More CmdletBinding

Module 12: Writing Full Help

- Lesson 1: External help
- Lesson 2: Using PlatyPs
- Lesson 3: Supporting online help
- Lesson 4: “About” topics
- Lesson 5: Making your help updatable

Module 13: Unit Testing Your Code

- Lesson 1: Sketching out the test
- Lesson 2: Making something to test
- Lesson 3: Expanding the test
- Lesson 4: Going further with Pester

Module 14: Extending Output Types

- Lesson 1: Understanding types
- Lesson 2: The Extensible Type System
- Lesson 3: Extending an object
- Lesson 4: Using Update-TypeData

Module 15: Analyzing Your Script

- Lesson 1: Performing a basic analysis
- Lesson 2: Analyzing the analysis

Module 16: Publishing Your Tools

- Lesson 1: Begin with a manifest
- Lesson 2: Publishing to PowerShell Gallery
- Lesson 3: Publishing to private repositories

Module 17: Basic Controllers: Automation Scripts and Menus

- Lesson 1: Building a menu
- Lesson 2: Using UIChoice
- Lesson 3: Writing a process controller

Module 18: Proxy Functions

- Lesson 1: A proxy example
- Lesson 2: Creating the proxy base
- Lesson 3: Modifying the proxy
- Lesson 4: Adding or removing parameters

Module 19: Working with XML Data

- Lesson 1: Simple: CliXML
- Lesson 2: Importing native XML
- Lesson 3: ConvertTo-XML
- Lesson 4: Creating native XML from scratch

Module 20: Working with JSON Data

- Lesson 1: Converting to JSON
- Lesson 2: Converting from JSON

Module 21: Working with SQL Server Data

- Lesson 1: SQL Server terminology and facts
- Lesson 2: Connecting to the server and database

- Lesson 3: Writing a query
- Lesson 4: Running a query
- Lesson 5: Invoke-SqlCmd
- Lesson 6: Thinking about tool design patterns

Module 22: Final Exam

- Lesson 1: Lab problem
- Lesson 2: Break down the problem
- Lesson 3: Do the design
- Lesson 4: Test the commands
- Lesson 5: Code the tool

Lab / Exercises

- Microsoft Labs Online

Documentation

- Digital course material included

Participant profiles

- Administrators in a Microsoft-centric environment who want to build reusable units of automation, automate business processes, and enable less-technical colleagues to accomplish administrative tasks

Prerequisites

- Experience at basic Windows administration
- Experience using Windows PowerShell to query and modify system information
- Experience using Windows PowerShell to discover commands and their usage
- Experience using WMI and/or CIM to query system information

Objectives

- Describe the correct patterns for building modularized tools in Windows PowerShell
- Build highly modularized functions that comply with native PowerShell patterns
- Build controller scripts that expose user interfaces and automate business processes
- Manage data in a variety of formats
- Write automated tests for tools
- Debug toolsager

Classroom Registration Price (CHF)
3900
Virtual Classroom Registration Price (CHF)
3650
Duration (in Days)
5
Reference
55039BC