

KOENIG SOLUTIONS • COURSE OUTLINE

MS-55039: Windows PowerShell Scripting and Toolmaking

Learn to design, build and publish PowerShell tools, from basic functions to error handling, testing and help documentation.

22

MODULES

91

TOPICS

19

HANDS-ON ACTIVITIES

COURSE OVERVIEW

This course covers designing PowerShell tools that follow native usage patterns, then building basic functions and modules and extending them with CmdletBinding, parameters, and object output. It moves on to verbose and error handling, debugging, comment-based and full help, and unit testing with Pester. Later modules address extending output types, script analysis, publishing tools, proxy functions, and working with XML, JSON, and SQL Server data, concluding with a practical final exam.

WHO THIS COURSE IS FOR

PowerShell Scripters

Systems Administrators

Automation Engineers

IT Professionals

DevOps Engineers

LEARNING OBJECTIVES

- Design PowerShell tools that follow native usage patterns
- Build basic functions and script modules
- Apply CmdletBinding and parameterize commands
- Construct and emit custom objects as output
- Implement verbose, warning, and informational output
- Handle errors and exceptions within a tool
- Debug scripts using Write-Debug and Set-PSBreakpoint
- Test code using Pester and analyse it with Script Analyzer

22 Modules, 91 Topics

01

Activity

Tool Design

This module explains how to design tools and units of automation that comply with native PowerShell usage patterns.

TOPICS

Tools do one thing · Tools are flexible · Tools look native

02

Activity

Start with a Command

This module explains how to start the scripting process by beginning in the interactive shell console.

TOPICS

Why start with a command? · Discovery and experimentation

03

Activity

Build a Basic Function and Module

This module explains how to build a basic function and module, using commands already experimented with in the shell.

TOPICS

Start with a basic function · Create a script module · Check prerequisites · Run the new command

04

Activity

Adding CmdletBinding and Parameterizing

This module explains how to extend the functionality of a tool, parameterize input values, and use CmdletBinding.

TOPICS

About CmdletBinding and common parameters · Accepting pipeline input · Mandatory-ness · Parameter validation · Parameter aliases

05

Activity

Emitting Objects as Output

This module explains how to create tools that produce custom objects as output.

TOPICS

Assembling information · Constructing and emitting output · Quick tests

06

An Interlude: Changing Your Approach

This module explains how to re-think tool design, using concrete examples of how it's often done wrong.

TOPICS

Examining a script · Critiquing a script · Revising the script

07

Activity

Using Verbose, Warning, and Informational Output

This module explains how to use additional output pipelines for better script behaviors.

TOPICS

Knowing the six channels · Adding verbose and warning output · Doing more with verbose output · Informational output

08

Activity

Comment-Based Help

This module explains how to add comment-based help to tools.

TOPICS

Where to put your help · Getting started · Going further with comment-based help · Broken help

09

Activity

Handling Errors

This module explains how to create tools that deal with anticipated errors.

TOPICS

- Understanding errors and exceptions
- Bad handling
- Two reasons for exception handling
- Handling exceptions in our tool
- Capturing the actual exception
- Handling exceptions for non-commands
- Going further with exception handling
- Deprecated exception handling

10

Activity

Basic Debugging

This module explains how to use native PowerShell script debugging tools.

TOPICS

Two kinds of bugs · The ultimate goal of debugging · Developing assumptions · Write-Debug · Set-PSBreakpoint · The PowerShell ISE

11

Going Deeper with Parameters

This module explains how to further define parameter attributes in a PowerShell command.

TOPICS

Parameter positions · Validation · Multiple parameter sets · Value from remaining arguments · Help messages · Aliases · More CmdletBinding

12

Activity

Writing Full Help

This module explains how to create external help for a command.

TOPICS

External help · Using PlatyPS · Supporting online help · “About” topics · Making your help updatable

13

Activity

Unit Testing Your Code

This module explains how to use Pester to perform basic unit testing.

TOPICS

Sketching out the test · Making something to test · Expanding the test · Going further with Pester

14

Extending Output Types

This module explains how to extend objects with additional capabilities.

TOPICS

Understanding types · The Extensible Type System · Extending an object · Using Update-TypeData

15**Activity**

Analyzing Your Script

This module explains how to use Script Analyzer to support best practices and prevent common problems.

TOPICS

Performing a basic analysis · Analyzing the analysis

16**Activity**

Publishing Your Tools

This module explains how to publish tools to public and private repositories.

TOPICS

Begin with a manifest · Publishing to PowerShell Gallery · Publishing to private repositories

17**Activity**

Basic Controllers: Automation Scripts and Menus

This module explains how to create controller scripts that put tools to use.

TOPICS

Building a menu · Using UIChoice · Writing a process controller

18**Activity**

Proxy Functions

This module explains how to create and use proxy functions.

TOPICS

A proxy example · Creating the proxy base · Modifying the proxy · Adding or removing parameters

19**Activity**

Working with XML Data

This module explains how to work with XML data in PowerShell.

TOPICS

Simple: CliXML · Importing native XML · ConvertTo-XML · Creating native XML from scratch

20

Activity

Working with JSON Data

This module explains how to using JSON data in PowerShell.

TOPICS

Converting to JSON · Converting from JSON

21

Working with SQL Server Data

This module explains how to use SQL Server from within a PowerShell script.

TOPICS

SQL Server terminology and facts · Connecting to the server and database · Writing a query · Running a query · Invoke-SqlCmd · Thinking about tool design patterns

22

Activity

Activity

Final Exam

This module provides a chance for students to use everything they have learned in this course within a practical example.

TOPICS

Lab problem · Break down the problem · Do the design · Test the commands · Code the tool

HANDS-ON ACTIVITIES

Practise What You Learn

#	MODULE	ACTIVITY
01	Tool Design	Lab : Designing a Tool: Design a tool
02	Start with a Command	Lab : Designing a Tool: Start with a command
03	Build a Basic Function and Module	Lab : Designing a Tool: Build a basic function and module
04	Adding CmdletBinding and Parameterizing	Lab : Designing a Tool: Adding CmdletBinding and Parameterizing
05	Emitting Objects as Output	Lab : Designing a Tool: Emitting objects as output
07	Using Verbose, Warning, and Informational Output	Lab : Designing a Tool: Using Verbose, Warning, and Informational Output

#	MODULE	ACTIVITY
08	Comment-Based Help	Lab : Designing a Tool: Comment-based help
09	Handling Errors	Lab : Designing a Tool: Handling errors
10	Basic Debugging	Lab : Designing a Tool: Basic debugging
12	Writing Full Help	Lab : Designing a Tool: Writing full help
13	Unit Testing Your Code	Lab : Designing a Tool: Unit testing your code
15	Analyzing Your Script	Lab : Designing a Tool: Analyzing your script
16	Publishing Your Tools	Lab : Designing a Tool: Publishing your tools
17	Basic Controllers: Automation Scripts and Menus	Lab : Designing a Tool: Basic controllers
18	Proxy Functions	Lab : Designing a Tool: Proxy functions
19	Working with XML Data	Lab : Designing a Tool: Working with XML
20	Working with JSON Data	Lab : Designing a Tool: Working with JSON data
22	Final Exam	Lab : Final Exam: Lab one Lab : Final Exam: Lab two

WHAT IS INCLUDED

Everything You Need to Succeed

- Koenig course material for every module
- Delivery by certified instructors and subject matter experts
- Hands-on exercises throughout the course
- Course completion certificate from Koenig

ABOUT KOENIG

Who We Are

Established in 1993, Koenig Solutions is a globally recognised training provider with 35+ technology partnerships, 5,000+ training programmes and a network of certified instructors and subject matter experts, supporting individuals and enterprise teams worldwide.

FLEXIBLE DELIVERY

Learn the Way That Suits You

Classroom Training

Live Online Classes (ILO)

Fly-Me-A-Trainer (FMAT)

Flexi (Self-Paced)

1-on-1 Training

Ready to Get Started?

Talk to a learning advisor about schedules, group training and customised delivery.

Visit koenig-solutions.com

✉ info@koenig-solutions.com