

Trainer Preparation Guide for Course 10962B: Intermediate Windows PowerShell® Scripting

Design of the Course

This course provides students with the intermediate knowledge and skills needed to create production-quality scripts by using Windows PowerShell. The course begins with a tutorial in creating Windows PowerShell advanced functions, and then has the student use those skills in a variety of production-oriented tasks. Each successive task includes repetition of previously learned skills, while introducing new skills and challenges. The course targets Windows PowerShell 4.0, but much of the material is applicable to older versions of Windows PowerShell.

This course consists of eight modules, delivered over three days.

Notice the Module 1 occupies the entire first day of class. It is important to keep students on-track, by monitoring their progress through the increasingly complex Module 1 labs. Instructor notes in the labs provide per-task timing estimates, and the Lab Answer Key (LAK) provides sample solutions that can help students catch up if they fall behind.

Module 1, *Creating Advanced Functions.* In this module students will learn how to parameterize a command and create an advanced function, convert a script and function into a script module, define parameter attributes and input validation for a function, enumerate objects by using scripting constructs, modify a function to accept pipeline input, produce complex pipeline output in a function, document a function by using comment-based help, and create functions that support `-WhatIf` and `-Confirm` to parameterize a command into an advanced function.

Module 2, *Using Cmdlets and Microsoft .NET Framework in Windows PowerShell.* In this module students will learn how to use Microsoft .NET Framework classes and instances to supplement functionality provided by Windows PowerShell commands.

Module 3, *Writing Controller Scripts.* In this module, students will learn how to describe the difference between tools and controller scripts, write controller scripts that present a user interface, and write controller scripts that automate a business process.

Module 4, *Handling Script Errors.* In this module, students will learn how to perform basic error handling in scripts. The focus will be on adding error handling to existing tools, primarily as a time-saving mechanism (as opposed to having students write new tools). A side benefit of this approach is that it will help build the skills needed to analyze and re-use existing code written by someone else. Topics include: describe the shell's default error response mechanisms, and add error handling code to existing tools.

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 of storing 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.

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.

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. Additionally, students will analyze and debug an existing script.

Module 8, *Understanding Windows PowerShell Workflow*. In this module, students will learn about the features of the Windows PowerShell Workflow technology. Because this is a niche technology, the focus is on helping students understand what it does and when it is appropriate. This module does not include a lab.

Required Materials to Teach This Course

To teach this course, you will need these materials from the [MCT Download Center](#).

- Trainer Handbook
- Instructor Slide Deck

Important The use of PowerPoint 2013, PowerPoint 2010, or PowerPoint 2007 is recommended to display the slides for this course. If you use PowerPoint Viewer or a version of PowerPoint older than PowerPoint 2007, some of the features of the slides might not display correctly.

- OneNote Trainer Pack
- Microsoft Hyper-V Classroom Setup Guide
- Course virtual machines

Prerequisite Knowledge to Teach This Course

To successfully present this course, you must have the following knowledge and skills:

- Core Windows PowerShell skills, including discovering commands, running commands, and manipulating command output. You can gain these skills by taking Microsoft Official Course 10961, *Automating Administration with Windows PowerShell*.
- Good foundation in Microsoft Windows administration skills, including a basic familiarity with server management, client management, Windows Management Instrumentation (WMI), and Active Directory.
- A background in programming, especially C-based languages like C# or C++, will be helpful but is not required.

Preparation Tasks

It is recommended that you complete the following instructional preparation activities:

- Read the About This Course at the beginning of the Course handbook for the learning product.
- Walk through the Introduction slide deck for the learning product.
- Walk through each module presentation slide deck and read the corresponding Instructor Notes (located in the notes view of the presentation slide deck) for the module. Note that each slide deck has additional hidden slides to accommodate the amount of Instructor Notes information for a given topic.
- Familiarize yourself with the Course Handbook and the Course Companion Content on the <http://www.microsoft.com/learning/companionmoc> site. Make note of when to direct students' attention to the Course Companion Content for further learning support. More information pertaining to the course components is present in the Introduction slide deck.
- Practice presenting each module:

- Identify the key points and must-know information for each topic.
- Perform each demonstration and hands-on lab.
- Anticipate the questions that students might have.
- Identify examples, analogies, impromptu demonstrations, and additional delivery tips that will help to clarify module content and provide a more meaningful learning experience for your specific audience.
- Note any problems that you might encounter during a demonstration or a lab exercise, and determine a course of action for how you will resolve the problems in the classroom. To access the lab answer keys, refer to the appendix in the Course Handbook.
- Work through the Module Review and Takeaways section at the end of each module and determine how you will use this section to reinforce student learning and promote knowledge transfer to on-the-job performance.
- Customize and enhance your instructor notes.

Courseware

The student courseware can be ordered in a digital format.

- Digital courseware is provided through the Arvato Skillpipe reader. The online reader can be accessed at <http://skillpipe.courseware-marketplace.com/reader>. To use the courseware your students will need Internet access.
- You should access the digital courseware and be prepared to help students register and login. The reader includes the ability to add notes, comments, and highlight content just add you would in printed materials.
- The course introduction “Module 0” PowerPoint slide deck includes a slide and an optional demonstration on the digital reader. You can use this information to help students get the most from their online materials.

Labs

The classroom lab virtual machines can be setup on-premise in a traditional classroom setting.

- Practice setting up the classroom by following the instructions in the Microsoft Hyper-V Classroom Setup Guide. This document provides hardware requirements for the instructor computer in addition to detailed setup instructions.
- Become familiar with the course virtual machines and how they are setup and configured—including the base images, mid-tier files, and activation states.
- The course introduction “Module 0” PowerPoint slide deck includes a slides discussing the lab environment. There is also an optional Hyper-V demonstration. Be prepared to answer student questions about how to use the labs in the classroom.

The classroom lab virtual machines can be hosted on Microsoft Labs Online (MLO).

- Become familiar with how the labs are accessed and configured. Ensure that you are ready to demonstrate their use to students and that everything is in place for a smooth experience.
- The course introduction “Module 0” PowerPoint slide deck includes a slide discussing Microsoft Labs online. There is also an optional demonstration. Be prepared to answer student questions about how to use the hosted lab environment.

Course Timing

The following schedule is an estimate of the course timing. Your timing might vary. Every student might not finish every lab. Use your judgment to set a reasonable time to move on to the next module.

Notice that the course timing does not allow for an extensive introduction if class starts at 9:00am. Instructors who prefer a more thorough introduction and familiarization period should start class earlier.

Be careful to observe the course timing. Extensive additional lecture, such as tangents or lengthy stories or examples, will compromise the successful delivery of the class. The course timing does not include explicit break times. However, there should be sufficient lab time for students to take brief breaks as needed throughout the day.

Day 1

Start	End	Module
9:00	9:15	Introductions and housekeeping
9:15	10:15	Module 1, Lesson 1 and Lab A
10:15	11:15	Module 1, Lesson 2 and Lab B
11:15	12:15	Module 1, Lesson 3 and Lab C
12:15	13:15	Lunch
13:15	13:45	Module 1, Lesson 4 and Lab D
13:45	14:15	Module 1, Lesson 5 and Lab E
14:15	15:00	Module 1, Lesson 6 and Lab F
15:00	15:30	Module 1, Lesson 7 and Lab G
15:30	16:30	Module 1, Lesson 8 and Lab H

Day 2

Start	End	Module
9:00	10:30	Module 2
10:30	10:45	Break
10:45	15:00	Module 3, including a 1-hour lunch
15:00	16:45	Module 4, including a short break

Day 3

Start	End	Module
9:00	11:00	Module 5 including a short break
11:00	14:15	Module 6, including a 1-hour lunch
14:15	16:30	Module 7, including a short break
16:30	17:00	Module 8

Material that was Not Included

This course excludes some advanced and intermediate content, based primarily on time constraints. The material that is included is felt to be the material that *most* administrators will get the *most* use out of in a production environment. The following list further explains some specific exclusions:

- Workflow. Not included beyond the overview module, because it is an advanced development topic with more limited production implementation. The feature is not presently in widespread use in most organizations. In many cases, DSC provides a better way to accomplish what workflow was meant, in part, to accomplish.
- DSC Resource authoring. This course focuses on using DSC with provided resources. Resource authoring is an advanced development topic that would require its own course and more substantial pre-requisites.
- Proxy functions. These are a fairly straightforward outgrowth of advanced functions, and their design and use is well documented. They have limited production applicability for most administrators.
- Retrieving information from the Internet. "Page scraping," as it is called, is a niche requirement for most organizations. PowerShell offers only limited native capability, and this topic quickly devolves into .NET Framework programming, which engages a massive set of additional pre-requisites.
- Working with SQL Server. SQL Server is obviously a good way to store data. However, working with SQL Server necessitates skill in the T-SQL language, which we do not have time to teach. We are instead teaching XML for data storage. The programming skills are similar between the two, meaning someone with existing T-SQL skills could take what they learn about XML and transfer those skills to SQL Server easily. XML is also usable in a wider variety of situations, such as when you need to write data on a server and don't have access to a SQL Server. XML is good *anywhere*, and so we felt it would offer broader use and appeal.

Delivery Options

Day 3 of the course, consisting of modules 5, 6, 7, and 8, may be taught in any order. The recommended delivery order is designed to place higher-value topics first, followed by topics that usually offer less production value in most environments. However, every class is unique, and your students' priorities may be different. Feel free to review the material in these modules, and deliver them in whatever order you feel will best meet the needs of your students. Just remember that, as the last day progresses, students will tend to become tired, and will remember less of the modules that are taught later in the day.

Lab Design Approach

Labs intended to produce complex scripts can be challenging in an instructor-led environment. Students move at varying speeds through the labs, and long labs can often create time management problems for students and instructors. Students who become lost can quickly fall behind. In addition, there is a desire to avoid inter-module dependencies. Instructor demonstrations can also fail because students spend more time copying down what the instructor is doing, instead of paying attention to the accompanying instructor explanation. To address these problems, this course's lab design includes the following features:

- In most cases, labs will require students to use 2 or more "tools." One of these tools will be used as an instructor demo and will be provided to students. Students will then use that as a model for creating 1 or more additional tools to use in the lab scenario. This ensures that students do not need to copy down what the instructor is demonstrating, because it will be given to them at the start of their lab.
- Creating tools (advanced functions) is a fundamental skill taught by this course. Students will learn to do so on the first day, and will repeat those skills over and over throughout the course. Modules on Days 2 and 3 will generally involve taking a tool (that was demonstrated by the instructor), creating a similar tool, and then using both tools to accomplish the lab scenario (e.g., generating a report, data file, etc.)
- No carry-over will be assumed between modules. That is, everything a student needs in a given module will either be provided to them at the start, or will be created by them as part of the module. There are no inter-module dependencies.
- No carry-over is *required* between the labs in a given module. That is, in most cases students *may* use their work from Lab A as a starting point for Lab B. However, a starting point will also be provided, so that students who do not complete a given lab always have a starting point for the next one.
- For longer lab assignments, the assignment will be broken down into discrete steps that involve no more than a half-dozen lines of code as the "answer." This will allow the high-level lab steps to cue students on a granular basis, without "giving away" the answer. This also allows the lab answer key to provide "just a little bit of the answer" for students seeking a clue or reminder. Instructor notes will provide granular timing estimates for each lab step, helping instructors manage long labs in increments of 5-10 minutes. This will help instructors identify students that are falling behind. In all cases, a final and working solution will be provided as the last lab answer key item.

Feedback

Any issues or feedback about the course can be reported to the Microsoft Learning courseware team through the MCT forums on the **Born to Learn website** at <http://borntolearn.mslearn.net/mct/amoc/w/wiki/default.aspx>.

Feedback that is shared on these forums will be available to the Microsoft Learning courseware team and to the MCT community. When possible, issues that are reported prior to the release of the final course will be researched and triaged by the courseware team. Microsoft Learning cannot guarantee that all reported issues will be resolved prior to release, but every effort will be made to resolve errors that are reported by MCTs.

Alternatively, you can provide your comments or feedback on the course by sending an email to support@microsoft.com with the subject line in the format, **MOC – Course XXXXy- Feedback**.

