DevOpsSchool

Course curriculum · November 2026

LEVEL 04 OF 04 · ADVANCED

Powershell Advanced Training

Automate at enterprise scale with parallel remoting, JEA, signed code, DSC and cloud modules, and migrate scripts to PowerShell 7.

LIVE ONLINE **Max 8-10 hours**	CORPORATE (DAY-WISE) **5 days · Days 1-5**	SELF-PACED VIDEO **1-year LMS access**
BATCH SIZE **Max 10 learners**	NEXT BATCHES **1st of every month · weekends: first Saturday**	CERTIFICATE **Verifiable certificate of completion**

Foundation	Essential	Intermediate	Advanced
1 day · Included	2 days · Included	3 days · Included	5 days · This curriculum

INSIDE THIS CURRICULUM

What you will achieve	Outcomes and level focus
Who should attend	Audience and prerequisites
Agenda	55 topics · Foundation to Advanced
Hands-on labs	4 labs
Formats and fees	Live, 1:1, self-paced, corporate
Trainer and reviews	Rajesh Kumar
Enrolment	Batches, payment, contacts

WHY DEVOPSSCHOOL

- Rated 4.6 on Google
- 1,500+ verifiable certificates issued in the public registry
- Trainer Rajesh Kumar: 20 years in industry
- Small batches: max 10 learners, every session recorded

Course page: www.devopsschool.com/courses/powershell/powershell-advance-training.html
Talk to us: WhatsApp +91 99057 40781 · +1 (469) 756-6329 · contact@DevOpsSchool.com · reply within 1 hour

OUTCOMES

What you will achieve

By the end of the Powershell course path, you will be able to:

1

Automate with PowerShell confidently

Use the pipeline, functions, modules and .NET types to automate Windows, Linux and cloud tasks.

2

Build and test automation

Create tested, analysed and documented modules and run them in psake and CI pipelines.

3

Operate securely at scale

Run parallel remoting, JEA, code signing and DSC to manage fleets of servers safely.

WHAT THE ADVANCED LEVEL ADDS

IDE automation, parallelism, JEA, security and cloud

Hands-on: Deliver a signed operations module. Build a class-based operations module that runs parallel remote checks, expose it through a JEA endpoint, sign it and publish it to a private feed.

WHO SHOULD ATTEND

- Experienced Powershell practitioners responsible for quality, security and scale.
- Senior engineers, architects and leads who design and troubleshoot Powershell solutions.
- Teams making Powershell an organisation-wide standard.

BEFORE YOU START

- The Intermediate level, or equivalent hands-on experience.
- You can already: Use .NET from PowerShell, package script modules, automate builds with psake and CI, and test, secure and debug your scripts.
- Basic computer skills and a laptop with a stable internet connection for the live sessions and labs.
- This curriculum includes every earlier level, so it is complete on its own.

AGENDA · FOUNDATION TO ADVANCED

Powershell Advanced Training agenda

55 topics. Each level builds on the one before it, so this curriculum lists every earlier level in full. Days are for corporate (day-wise) delivery; live online batches cover the same agenda in max 8-10 hours.

DAY 1 — Foundation: PowerShell primer, help, pipelines and providers

Included from Foundation

1. Overview and PowerShell primer: Windows PowerShell 5.1 vs PowerShell 7.4 LTS and 7.5, cross-platform pwsh
2. PowerShell environments: console, Windows Terminal, VS Code with the PowerShell extension, the ISE and other hosts
3. Installation: winget, MSI, Homebrew, apt and dnf packages; running side by side with Windows PowerShell
4. Discover stuff using help: Get-Help, Update-Help, -Examples and -Online, about_ topics
5. Finding commands and making mistakes: Get-Command, verb-noun naming, Get-Verb, aliases and reading errors
6. Expressions and variables: types, strings, here-strings, arrays, hashtables and automatic variables
7. Flow control: if, elseif, switch, for, foreach, while and do loops
8. Pipelines: objects not text, Get-Member, Select-Object, Sort-Object and Measure-Object
9. Members, Where-Object and ForEach-Object: filtering and transforming objects on the pipeline
10. Location, location, location: Get-Location, Set-Location, Push-Location and Pop-Location across providers
11. PowerShell drives: FileSystem, Registry, Env and Cert providers and New-PSDrive
12. Cutting corners in Windows Explorer: Invoke-Item to open files and folders, Start-Process
13. Scripts and execution policy: .ps1 files, Set-ExecutionPolicy, RemoteSigned, Unblock-File and running scripts

DAY 2 — Essential: Functions, profiles, prompts, native tools and modules

Included from Essential

14. Functions: param blocks, parameters, output and the Push-Project demo function
15. Advanced functions: CmdletBinding, parameter validation attributes, Write-Verbose and -WhatIf support
16. Supporting pipeline operations in functions: ValueFromPipeline and begin, process and end blocks
17. Profiles and hosts: $PROFILE paths, per-host profiles, loading functions and aliases at startup, Push-Project summary
18. Customising the PowerShell prompt: the prompt function, a source-control aware prompt, posh-git and oh-my-posh
19. Files and data: Get-Content, Set-Content, Import-Csv, Export-Csv, ConvertFrom-Json and ConvertTo-Json
20. Text processing: -match, -replace, -split, Select-String and regular expressions
21. Error handling: try, catch and finally, -ErrorAction, $ErrorActionPreference, terminating vs non-terminating errors
22. PowerShell commands vs console applications: using console applications in PowerShell, $LASTEXITCODE and arguments
23. Capturing commands as PowerShell functions: wrapping native tools such as git, hg and dotnet, Crescendo basics
24. A toast to PowerShell modules: where you find PowerShell modules, the PowerShell Gallery and module discovery
25. Installing modules: Install-PSResource and Install-Module, manual module installation and PSModulePath
26. Useful modules for software developers: PSReadLine, posh-git, Pester, PSScriptAnalyzer, Az and Microsoft.Graph
27. Remoting basics: Enter-PSSession and Invoke-Command over WinRM and SSH

DAY 3 — Intermediate: .NET interop, script modules, psake builds and testing

Included from Intermediate

28. PowerShell and .NET: referencing types, creating objects with New-Object and ::new()

29. Using static and instance members: [Math], [DateTime] and [IO.Path] methods and properties
30. Handling events: Register-ObjectEvent, more event handling with FileSystemWatcher and timers
31. Loading assemblies: Add-Type, loading DLLs and compiling inline C#
32. Sandbagging automation: the ConvertTo-Sandbags.ps1 script and revisiting it with .NET types
33. Script modules: .psm1 files, .psd1 manifests, exported functions, versioning and module summary
34. Automating bisections: the Find-HgChangeset function, the Hg PowerShell module and git bisect run equivalents
35. Build automation with psake: the nature of the build, installing psake, running and documenting build scripts
36. psake for .NET: building Visual Studio projects, why psake over raw MSBuild, the PackageZip task, configuring the build
37. CI integration: running psake builds in TeamCity, plus pwsh steps in GitHub Actions and Azure Pipelines
38. Testing with Pester 5: Describe, It, Should, mocks and code coverage
39. Code quality: PSScriptAnalyzer rules, consistent formatting and comment-based help
40. Secrets: SecretManagement and SecretStore modules instead of plain-text credentials
41. Troubleshooting: Set-PSBreakpoint, VS Code debugging, Set-PSDebug, transcripts and $Error inspection

DAYS 4-5 **Advanced: IDE automation, parallelism, JEA, security and cloud** **New at this level**

42. Automating Visual Studio with StudioShell: exploring Visual Studio features and configuring window layouts
43. Extending Visual Studio: adding psake build menu items and capturing changes in a solution module
44. Conditional refactoring: overview, strategy and scaling refactoring across large solutions
45. Classes and enums: PowerShell classes, inheritance, custom types and format and type files
46. Parallelism: ForEach-Object -Parallel, ThreadJob, runspaces and throttling
47. Remoting at scale: fan-out Invoke-Command, persistent sessions, SSH remoting to Linux and the double-hop problem
48. Just Enough Administration: role capability files, session configurations and constrained endpoints
49. Security hardening: code signing, AllSigned policy, script block and module logging, Constrained Language Mode
50. Configuration management: Microsoft DSC v3 resources and configuration documents, Azure Machine Configuration
51. Cloud automation: Az and Microsoft.Graph modules, Azure Automation runbooks and managed identities
52. Publishing modules: private repositories with PSResourceGet, semantic versioning and release pipelines
53. Performance: Measure-Command, pipeline overhead, efficient collections and string building
54. Migration: moving scripts from Windows PowerShell 5.1 to PowerShell 7, the compatibility layer and breaking changes
55. Capstone: build, sign, test and publish an operations module with JEA endpoints and CI release

PRACTICE

Hands-on labs

You build these yourself during the course, guided by the trainer.

FOUNDATION

Explore a system with the pipeline

Use help and the pipeline to report the top processes, stopped services and largest files on a machine, then save it as a script.

ESSENTIAL

Build a developer shell profile

Create a profile with a source-control aware prompt, a Push-Project navigation function and wrappers for console tools that accept pipeline input.

INTERMEDIATE

Build and test with psake

Package helper functions as a script module, write Pester tests, and run a psake build that tests, zips and publishes it from a CI job.

ADVANCED

Deliver a signed operations module

Build a class-based operations module that runs parallel remote checks, expose it through a JEA endpoint, sign it and publish it to a private feed.

HOW YOU CAN LEARN

Formats and fees

The same agenda in four formats. Fees are in Indian rupees as of November 2026; the course page shows them in your currency.

FORMAT	DURATION	WHAT YOU GET	FEE
Live & interactive (online)	Max 8-10 hours	Instructor-led public batch of max 10 learners with demos, labs, Q&A and assignment review. Weekday batches start on the 1st of every month, weekend batches on the first Saturday of every month.	**INR 34,999**
1:1 live training (online)	Max 8-10 hours, one-to-one	The same agenda, taught one-to-one by the trainer. Starts immediately, without waiting for the next batch; sessions planned around your working hours, at your pace.	**INR 99,999**
Self-paced video	1-year LMS access	All four Powershell levels, plus 20+ more courses and 50+ tools, at your own pace.	**INR 833/month** billed yearly at INR 9,996
Corporate (day-wise)	5 days for this level	A private batch for your team, on demand. Foundation 1 day, Essential 2, Intermediate 3, Advanced 5 (cumulative).	**Quote on request**

ASSESSMENT AND CERTIFICATE

Your progress is checked through hands-on labs and assignments at every level. On completion you receive a DevOpsSchool certificate of completion with a unique credential ID that anyone can verify at devopsschool.com/certificates.

INCLUDED WITH EVERY LIVE BATCH

- Every live session is recorded and stays in your LMS for a year.
- Can't attend your batch? Move to next month's batch.
- Free demo class before you pay.
- Free 10-15 minute call with an advisor or the trainer to check the course fits you.

YOUR TRAINER

Learn from Rajesh Kumar

Principal Architect, trainer and mentor in DevOps, SRE, DevSecOps and platform engineering, with 20 years across 8+ software MNCs, including Staff DevOps/SRE Engineer for a leading Japanese mobile-payment platform and Principal DevOps Architect and Manager in Europe.

Profile: www.rajeshkumar.xyz · Videos: youtube.com/@TheDevOpsSchool

20 years in industry	**200+** corporate trainings
10,000+ engineers mentored	**70+** organisations coached

REVIEWS

What learners say

Rated 4.6 on Google · more reviews at devopsschool.com/testimonials and in our video testimonials.

> “Rajesh is very good trainer. Rajesh was able to resolve our queries and question effectively. We really liked the hands-on examples covered during this training program.”
>
> Indrayani, India

> “The training was very useful and interactive. Rajesh helped develop the confidence of all.”
>
> Abhinav Gupta, Pune

START YOUR POWERSHELL TRAINING

Join the next batch

Weekday batches start on the 1st of every month and weekend batches on the first Saturday of every month, with max 10 learners each. 1:1 training can start immediately; corporate batches run on demand. We reply within 1 hour.

Pay by UPI, credit and debit cards, net banking or PayPal. Before you pay, read our refund policy. All levels of this course: www.devopsschool.com/courses/powershell/

ENROL OR RESERVE A SEAT
Powershell Advanced Training

WHATSAPP · INDIA
+91 99057 40781

WHATSAPP · USA
+1 (469) 756-6329

EMAIL
contact@DevOpsSchool.com