

Windows PowerShell Scripting and Toolmaking

Description

Découvrez comment implémenter un Système de Management de la Sécurité de l'Information conforme à la norme ISO/IEC 27001

Ce cours de cinq jours est destiné aux professionnels de l'informatique qui souhaitent développer leurs compétences dans Windows PowerShell et l'automatisation administrative. Le cours suppose une connaissance de base de PowerShell en tant que shell de ligne de commande interactif, et enseigne aux apprenants les schémas et les pratiques corrects pour construire des unités d'automatisation réutilisables et à portée limitée.

Nous vous permettons de gagner en autonomie avec Windows PowerShell et avec l'automatisation administrative

Cette formation vous prépare efficacement afin que vous puissiez entre autres créer des fonctions hautement modulaires, gérer les données dans une variété de formats et finalement prendre en main les outils de débogage.

Prix de l'inscription en Présentiel (CHF)

3900

Prix de l'inscription en Virtuel (CHF)

3650

Contenu du cours

Module 1 : Conception d'outils

- Leçon 1 : Les outils font une chose
- Leçon 2 : Les outils sont flexibles
- Leçon 3 : Les outils semblent natifs

Module 2 : Commencer avec une commande

- Leçon 1 : Pourquoi commencer avec une commande?
- Leçon 2 : Découverte et expérimentation

Module 3 : Construire une fonction de base et un module

- Leçon 1 : Commencer avec une fonction de base
- Leçon 2 : Créer un module de script
- Leçon 3 : Vérifier les prérequis
- Leçon 4 : Exécuter la nouvelle commande

Module 4 : Ajout de CmdletBinding et paramétrage

- Leçon 1 : A propos de CmdletBinding et paramètres communs
- Leçon 2 : Accepter l'entrée de pipeline
- Leçon 3 : Obligation-ness
- Leçon 4 : Validation des paramètres

- Leçon 5 : Alias de paramètre

Module 5 : Emission d'objets en sortie

- Leçon 1 : Assembler l'information
- Leçon 2 : Construire et émettre une sortie
- Leçon 3 : Tests rapides

Module 6 : Interlude - Changer votre approche

- Leçon 1 : Examiner un script
- Leçon 2 : Critiquer un script
- Leçon 3 : Réviser le script

Module 7 : Utilisation de verbose, d'avertissement et de sortie d'informations

- Leçon 1 : Connaître les six canaux
- Leçon 2 : Ajout de sortie verbose et d'avertissement
- Leçon 3 : En faire plus avec une sortie verbose
- Leçon 4 : Sortie d'information

Module 8 : Aide basée sur les commentaires

- Leçon 1 : Où mettre votre aide
- Leçon 2 : Commencer
- Leçon 3 : Aller plus loin avec l'aide basée sur les commentaires
- Leçon 4 : Aide brisée

Module 9 : Erreurs de manipulation

- Leçon 1 : Comprendre les erreurs et les exceptions
- Leçon 2 : Mauvaise manipulation
- Leçon 3 : Deux raisons pour la gestion des exceptions
- Leçon 4 : Gérer les exceptions dans notre outil
- Leçon 5 : Capture de l'exception réelle
- Leçon 6 : Gestion des exceptions pour les non-commandes
- Leçon 7 : Aller plus loin avec la gestion des exceptions
- Leçon 8 : Gestion des exceptions obsolète

Module 10 : Débogage de base

- Leçon 1 : Deux types de bugs
- Leçon 2 : L'objectif ultime du débogage
- Leçon 3 : Développer des hypothèses
- Leçon 4 : Write-Debug
- Leçon 5 : Set-PSBreakpoint
- Leçon 6 : L'ISE PowerShell

Module 11 : Aller plus loin avec les paramètres

- Leçon 1 : Positions de paramètre
- Leçon 2 : Validation
- Leçon 3 : Ensembles de paramètres multiples

- Leçon 4 : Valeur des arguments restants
- Leçon 5 : Messages d'aide
- Leçon 6 : Alias
- Leçon 7 : Plus de CmdletBinding

Module 12 : Écrire une aide complète

- Leçon 1 : Aide externe
- Leçon 2 : Utilisation de PlatyPs
- Leçon 3 : Aide en ligne
- Leçon 4 : "À propos" des sujets
- Leçon 5 : Rendre votre aide modifiable

Module 13 : Unité testant votre code

- Leçon 1 : Esquisser le test
- Leçon 2 : Faire quelque chose à tester
- Leçon 3 : Expansion du test
- Leçon 4 : Aller plus loin avec Pester

Module 14 : Extension des types de sortie

- Leçon 1 : Comprendre les types
- Leçon 2 : Le système de type extensible
- Leçon 3 : Extension d'un objet
- Leçon 4 : Utilisation de Update-TypeInfo

Module 15 : Analyser votre script

- Leçon 1 : Effectuer une analyse de base
- Leçon 2 : Analyser l'analyse

Module 16 : Publication de vos outils

- Leçon 1 : Commencez avec un manifeste
- Leçon 2 : Publication à la galerie PowerShell
- Leçon 3 : Publication dans des référentiels privés

Module 17 : Contrôleurs de base: Scripts d'automatisation et menus

- Leçon 1 : Construire un menu
- Leçon 2 : Utiliser UIChoice
- Leçon 3 : Ecrire un contrôleur de processus

Module 18 : Fonctions de proxy

- Leçon 1 : Un exemple de proxy
- Leçon 2 : Création de la base de proxy
- Leçon 3 : Modifier le proxy
- Leçon 4 : Ajouter ou supprimer des paramètres

Module 19 : Travailler avec des données XML

- Leçon 1 : Simple: CliXML

- Leçon 2 : Importation de XML natif
- Leçon 3 : ConvertTo-XML
- Leçon 4 : Créer du XML natif à partir de zéro

Module 20 : Utilisation des données JSON

- Leçon 1 : Conversion en JSON
- Leçon 2 : Conversion à partir de JSON

Module 21 : Utilisation des données SQL Server

- Leçon 1 : Terminologie SQL Server et faits
- Leçon 2 : Connexion au serveur et à la base de données
- Leçon 3 : Ecrire une requête
- Leçon 4 : Exécuter une requête
- Leçon 5 : Invoke-SqlCmd
- Leçon 6 : Penser aux modèles de conception d'outils

Module 22 : Examen final

- Leçon 1 : Problème de laboratoire
- Leçon 2 : Briser le problème
- Leçon 3 : Faire le design
- Leçon 4 : Testez les commandes
- Leçon 5 : Coder l'outil

Lab / Exercices

- Laboratoires Microsoft en ligne

Documentation

- Support de cours numérique inclus

Profils des participants

- Administrateurs dans un environnement centré sur Microsoft qui souhaitent créer des unités d'automatisation réutilisables, automatiser les processus métier et permettre à des collègues moins techniques d'accomplir des tâches administratives

Connaissances Préalables

- Expérience à l'administration Windows de base
- Expérience de l'utilisation de Windows PowerShell pour interroger et modifier les informations système
- Expérience avec Windows PowerShell pour découvrir les commandes et leur utilisation
- Expérience en utilisant WMI et / ou CIM pour interroger les informations système

Objectifs

- Décrire les modèles corrects pour construire des outils modulaires dans Windows PowerShell
- Créer des fonctions hautement modulaires conformes aux modèles PowerShell natifs
- Construire des scripts de contrôleurs qui exposent les interfaces utilisateur et automatisent les processus métier
- Gérer les données dans une variété de formats
- Écrire des tests automatisés pour les outils

- Outil de débogage

Niveau

Intermédiaire

Durée (Nombre de Jours)

5

Reference

55039BC