

Chapter 1 : PowerShell Automatic Variables - Windows PowerShell Pocket Reference [Book]

This portable reference to Windows PowerShell summarizes the command shell and scripting language, and provides a concise reference to the many tasks that make PowerShell so useful.

A Whirlwind Tour of Windows PowerShell Introduction When learning a new technology, it is natural to feel bewildered at first by all the unfamiliar features and functionality. This perhaps rings especially true for users new to Windows PowerShell, because it may be their first experience with a fully featured command-line shell. Fortunately, these fears are entirely misguided: PowerShell is a shell that both grows with you and grows on you. PowerShell works with standard Windows commands and applications. PowerShell introduces a powerful new type of command. PowerShell commands called cmdlets share a common Verb-Noun syntax and offer many usability improvements over standard commands. Working directly with richly structured objects makes working with and combining PowerShell commands immensely easier than working in the plain-text world of traditional shells. PowerShell caters to administrators. Even with all its advances, PowerShell focuses strongly on its use as an interactive shell, and the experience of entering commands in a running PowerShell application. Using three simple commands, you can learn and discover almost anything PowerShell has to offer. PowerShell enables ubiquitous scripting. With a fully fledged scripting language that works directly from the command line, PowerShell lets you automate tasks with ease. PowerShell bridges many technologies. By letting you work with. PowerShell simplifies management of data stores. Through its provider model, PowerShell lets you manage data stores using the same techniques you already use to manage files and folders. While it supports scripting and other powerful features, its focus as a shell underpins everything. Getting started in PowerShell is a simple matter of launching PowerShell. To launch Windows PowerShell, click: Entering standard DOS-style file manipulation commands in response to the PowerShell prompt produces the same results you get when you use them with any other Windows shell PS C: Tip The pushd command is an alternative name alias to the much more descriptively named PowerShell command, Push-Location. Likewise, the cd, dir, popd, and pwd commands all have more memorable counterparts. Although navigating around the filesystem is helpful, so is running the tools you know and love, such as ipconfig and notepad. Try them both on your own machine. Structured Commands Cmdlets In addition to supporting traditional Windows executables, PowerShell introduces a powerful new type of command called a cmdlet pronounced command-let. Tip Once you know the handful of common verbs in PowerShell, learning how to work with new nouns becomes much easier. While you may never have worked with a certain object before such as a Service , the standard Get, Set, Start, and Stop actions still apply. PowerShell lets you use the Tab key to auto-complete cmdlet names and parameter names: For improved efficiency, PowerShell defines aliases for all common commands and lets you define your own. In addition to alias names, PowerShell only requires that you type enough of the parameter name to disambiguate it from the other parameters in that cmdlet. PowerShell is also case-insensitive. Using the built-in gps alias that represents the Get-Process cmdlet along with parameter shortening , you can instead type: Positional parameters let you provide parameter values in a certain position on the command line, rather than having to specify them by name. The Get-Process cmdlet takes a process name as its first positional parameter. This parameter even supports wildcards: For example, the following command generates a simple text string. Since nothing captures that output, PowerShell displays it to you: For example, you can access its Length property, which tells you how many characters are in the string. To access a property, you place a dot between the object and its property name: Length 11 All PowerShell commands that produce output generate that output as objects as well. For example, the Get-Process cmdlet generates a System. Process object, which you can store in a variable. If you have an instance of Notepad running, the following command stores a reference to it: NET Framework, you can call methods on that object to perform actions on it. This command calls the Kill method, which stops a process. To access a method, you place a dot between the object and its method name: Kill PowerShell supports this functionality more directly through the Stop-Process cmdlet, but this example demonstrates an important point about your ability to interact with these rich objects. NET Framework quickens the pulse of most users,

PowerShell continues to focus strongly on administrative tasks. For example, PowerShell supports MB for megabyte and GB for gigabyte as some of the standard administrative constants. NET Framework for administrative tasks, though! In fact, PowerShell makes a great calendar. For example, is a leap year? PowerShell can tell you: `IsLeapYear True` Going further, how might you determine how much time remains until summer? If the second command understands the objects produced by the first command, it can operate on the results. You can chain together many commands this way, creating powerful compositions out of a few simple operations. For example, the following command gets all items in the Path1 directory and moves them to the Path2 directory: It passes those to the Where-Object cmdlet, which runs a comparison against each incoming item. For each object in which this comparison holds true, you pass the results to the Sort-Object cmdlet, asking it to sort items by their Handles property. Finally, you pass the objects to the Format-Table cmdlet to generate a table that contains the Handles, Name, and Description of the process. After all, what does this command do? How can you be sure? Let PowerShell tell you. For commands that modify data, PowerShell supports -WhatIf and -Confirm parameters that let you see what a command would do: Performing operation "Stop-Process" on Target "ctfmon ". Performing operation "Stop-Process" on Target "Ditto ". Performing operation "Stop-Process" on Target "dsamain ". Performing operation "Stop-Process" on Target "ehrecvr ". Performing operation "Stop-Process" on Target "ehSched ". Performing operation "Stop-Process" on Target "explorer ". In this interaction, using the -WhatIf parameter with the Stop-Process pipelined command lets you preview which processes on your system will be stopped before you actually carry out the operation. Note that this example is not a dare! In the words of one reviewer: Not only did it stop everything, but on Vista, it forced a shutdown with only one minute warning! It was very funny though! At least I had enough time to save everything first! Common Discovery Commands While reading through a guided tour is helpful, I find that most learning happens in an ad-hoc fashion. To find all commands that match a given wildcard, use the Get-Command cmdlet. For example, by entering the following, you can find out which PowerShell commands and Windows applications contain the word process. To see what a command such as Get-Process does, use the Get-Help cmdlet, like this: NET Framework, it provides the Get-Member cmdlet to retrieve information about the properties and methods that an object, such as a. Piping a string to the Get-Member command displays its type name and its members: String Name MemberType Definition String PadLeft Int32 tota String PadRight Int32 tot String Remove Int32 start String Replace Char oldCh String[] Split Params Cha String Substring Int32 st Char[] ToCharArray , Sys String ToLower , System String ToString , System String ToUpper , System String Trim Params Char[] String TrimEnd Params Cha String TrimStart Params C This means that your favorite cmdlets work in scripts and that your favorite scripting techniques such as the foreach statement work directly on the command line. For example, to add up the handle count for all running processes:

Chapter 2 : Windows PowerShell Pocket Reference (Pocket Reference (O'Reilly)) - Ebook pdf and epub

Windows PowerShell Pocket Reference by Lee Holmes This portable reference to Windows PowerShell summarizes the command shell and scripting language, and provides a concise reference to the many tasks that make PowerShell so useful.

If you buy something we get a small commission at no extra charge to you. Is it a CLI or a scripting language? Is it proprietary or are there 3rd party extensions? All of these questions can be answered with the right learning materials. In this post I want to share the absolute best PowerShell books for everyone from complete newbies to more experienced scripters. Regardless of your knowledge level you can easily find something in this post to up your PowerShell game to the next level. It covers all the fundamentals you need to know but the writing style is very relaxed and simple to follow. This is crucial for someone with less technical knowledge who wants to learn PowerShell without the overly-confusing technical language. Windows PowerShell Programming for the Absolute Beginner Currently in its 3rd edition, Windows PowerShell Programming for the Absolute Beginner is one of my favorite books to skim through and recommend for absolute beginners. This book focuses on Windows management and some game development, so it may not be the perfect choice for everyone. But it is one of my favorite books because of how the content is presented and how easy it is to consume. Concepts are super easy to understand even with no programming background. And clocking in at pages I think this is one of the lengthier programming books you can get. The biggest drawback from studying PowerShell is the technical aspect. Many people get scared because of the innate jargon that just sounds incomprehensible to the normal human ear. But PowerShell For Beginners! This book is made for less tech-savvy individuals who desperately want to understand the power of Windows PowerShell in the real world. The writing style is super simple to understand. The book is currently in its 3rd edition totaling just over pages long. The author Ed Wilson is a recognized scripting expert by Microsoft and his name is widely known in the PowerShell community. Other more advanced topics include managing Active Directory, WMI systems, handing software services, and learning to debug your own scripts. You do not need prior experience but you do need some tenacity to work through this book. It is made for beginners but the detailed writing style mostly targets tech enthusiasts. The Windows PowerShell Cookbook is freaking massive. It totals over 1, pages long with just over unique recipes to choose from. Some recipes get into fairly mundane content like Active Directory management and the object-based pipeline features. But Holmes also gets into. Windows PowerShell in Action is currently in its 3rd edition and it easily spans over 1, pages. PowerShell in Action teaches you how to use PowerShell with real case studies and live examples. If you already feel comfortable with PowerShell then you may be using it here-and-there throughout your daily routine. But if you really want to become a power user then this book will get you there. It covers a huge variety of use cases for PowerShell ranging from IT professionals to server administrators and programmers. The chapters list is huge and this book is undeniably the best way to go from an intermediate to a super advanced PowerShell user. But the difficult part is figuring out the best workflow and how this fits into your daily activities. The book is fairly short at only pages. But it covers all the advanced topics of automation, debugging, and writing scripts with best coding practices. This includes detailed user account permissions and even complex regular expressions within your PowerShell scripts. And this book will get you there by teaching you the right way to automate scripting. This book is written by Lee Holmes, the same author of the 1, word PowerShell cookbook mentioned earlier. The tips in this pocket reference are incredibly valuable for. You can look up quick scripts or solutions for common problems with ease. Unfortunately the latest release came out in December so it only covers PowerShell 3. If you still work on legacy systems this book can be a huge asset. This includes the process of automating certain scripting problems or CRON jobs running on the server. PowerShell in Depth by Manning is one of the best books on the market with a hands-on tutorial approach to each exercise. The book has over pages with just over 40 chapters covering a wide variety of advanced PowerShell techniques. We actually published a review of this book if you want to read more about the content. The author covers typical features like. This book is mostly aimed at anyone who wants to learn the newest features of

PowerShell 5. Some users come from a technical background with Bash while others have never touched a command line prompt. However the more advanced books are only valuable on a case-by-case basis. Some people may need an IT tech reference while others may need a detailed tutorial on PowerShell for databases. You can find all of that and more from the titles in this post. Browse through this list and if anything looks appealing check it out further on Amazon. Many publishers update their tech books with newer editions so always search for the most up-to-date version before buying. Jaime Morrison Jaime is a jr. He covers general news and useful resources in the web design space.

Chapter 3 : Windows PowerShell and Other Quick Reference Guides – PowerShell Magazine

This portable reference to Windows PowerShell summarizes the command shell and scripting language, and provides a concise reference to the many tasks that make PowerShell so useful. If you're a busy Windows administrator, and don't have time to plow through huge books or search online, this is the ideal on-the-job tool.

Mostly, it occurs when the new readers cease using the eBooks as they are unable to utilize all of them with the appropriate and effective fashion of reading these books. There present variety of reasons behind it due to which the readers stop reading the eBooks at their first most attempt to make use of them. Nonetheless, there exist some techniques that could help the readers to really have a good and effective reading experience. Someone ought to adjust the suitable brightness of display before reading the eBook. Because of this they suffer from eye sores and headaches. The very best alternative to overcome this serious issue is to reduce the brightness of the displays of eBook by making specific changes in the settings. It is proposed to keep the brightness to possible minimum amount as this can help you to raise the time you could spend in reading and provide you great comfort onto your eyes while reading. A great eBook reader ought to be set up. It will be useful to really have a great eBook reader to be able to truly have a great reading experience and high quality eBook display. You can even make use of free software that can offer the readers with many functions to the reader than simply a simple platform to read the wanted eBooks. Besides offering a place to save all your valuable eBooks, the eBook reader software even provide you with a lot of attributes in order to improve your eBook reading experience in relation to the standard paper books. You may also enhance your eBook reading encounter with help of choices supplied by the software program including the font size, full display mode, the specific variety of pages that need to be exhibited at once and also alter the color of the background. You need to take appropriate breaks after specific intervals while reading. Nevertheless, this does not mean that you ought to step away from the computer screen every now and then. Constant reading your eBook on the computer screen for a long time without taking any rest can cause you headache, cause your neck pain and suffer with eye sores and in addition cause night blindness. So, it is essential to give your eyes rest for a little while by taking rests after specific time intervals. This will help you to prevent the problems that otherwise you may face while reading an eBook constantly. While reading the eBooks, you must prefer to read enormous text. So, boost the size of the text of the eBook while reading it at the monitor. It is suggested that never use eBook reader in full screen mode. It is recommended not to go for reading the eBook in full-screen mode. Even though it might look simple to read with full screen without turning the page of the eBook fairly often, it place ton of pressure in your eyes while reading in this mode. Consistently favor to read the eBook in exactly the same length that will be similar to the printed book. This is so, because your eyes are used to the span of the printed book and it would be comfortable for you to read in the same way. By using different techniques of page turn you can additionally improve your eBook encounter. Check out whether you can turn the page with some arrow keys or click a specific portion of the display, apart from utilizing the mouse to handle everything. Try using the mouse if you are comfy sitting back. Lesser the movement you need to make while reading the eBook better will be your reading experience. Technical problems One problem on eBook readers with LCD screens is that it is not going to take long before you try your eyes from reading. This will definitely help make reading easier. By using each one of these powerful techniques, you can surely boost your eBook reading experience to a terrific extent. This advice will help you not only to prevent particular risks which you may face while reading eBook often but also ease you to relish the reading experience with great relaxation. The download link provided above is randomly linked to our ebook promotions or third-party advertisements and not to download the ebook that we reviewed. We recommend to buy the ebook to support the author. Thank you for reading. Search a Book Search Recommended Books.

Chapter 4 : Windows PowerShell Pocket Reference by Lee Holmes

This portable reference to Windows PowerShell summarizes both the command shell and scripting language, and

DOWNLOAD PDF WINDOWS POWERSHELL POCKET REFERENCE

provides a concise reference to the major tasks that make PowerShell so successful. It's an ideal on-the-job tool for Windows administrators who don't have time to plow through huge books or search online.

Chapter 5 : A Whirlwind Tour of Windows PowerShell - Windows PowerShell Pocket Reference [Book]

Find helpful customer reviews and review ratings for Windows PowerShell Pocket Reference (Pocket Reference (O'Reilly)) at blog.quintoapp.com Read honest and unbiased product reviews from our users.

Chapter 6 : Windows Powershell Pocket Reference - O'Reilly Media

Windows PowerShell Pocket Reference by Lee Holmes Stay ahead with the world's most comprehensive technology and business learning platform. With Safari, you learn the way you learn best.

Chapter 7 : Windows PowerShell Pocket Reference, 2nd Edition : Books

Stay ahead with the world's most comprehensive technology and business learning platform. With Safari, you learn the way you learn best. Get unlimited access to videos, live online training, learning paths, books, tutorials, and more.

Chapter 8 : 10 Best Windows PowerShell Books

To solve that problem, this Pocket Reference summarizes the Windows PowerShell command shell and scripting language, while also providing a concise reference for the major tasks that.

Chapter 9 : Windows PowerShell Pocket Reference, 2nd Edition - PDF Free Download - Fox eBook

This portable reference to Windows PowerShell summarizes both the command shell and scripting language, and provides a concise reference to the major tasks that make PowerShell so successful.