PowerShell Fundamentals

Jaap Brasser

AZURE
OFFICE 365
ENTERPRISE MOBILITY SUITE
OPERATIONS MANAGEMENT SUITE
AZURE STACK
HYPER-V
WINDOWS

Module 1 – PowerShell fundamentals

WINDOWS

- What is PowerShell and how did it start
- How to read the PowerShell language
- Different versions of PowerShell
- Help and command discovery in PowerShell
- What are objects in PowerShell
- The PowerShell pipeline

What is PowerShell and how did it start

- Development started in 2001
- First released in 2006
- Created by a team led by Jeffrey Snover
- Was initially codenamed Monad

Monad Manifesto

- Written by Jeffrey Snover in 2002
- Set the definition for what would be PowerShell
- The following 5 definitions:
  - *Monad Automation Model*
  - *Monad Shell*
  - *Monad Management Model*
  - *Monad Remote Scripting*
  - *Monad Management Console*

WINDOWS

PowerShell naming

- Verb-SingularNoun
- Verb has to be ‘approved’
- Noun should be descriptive, can contain a prefix

WINDOWS

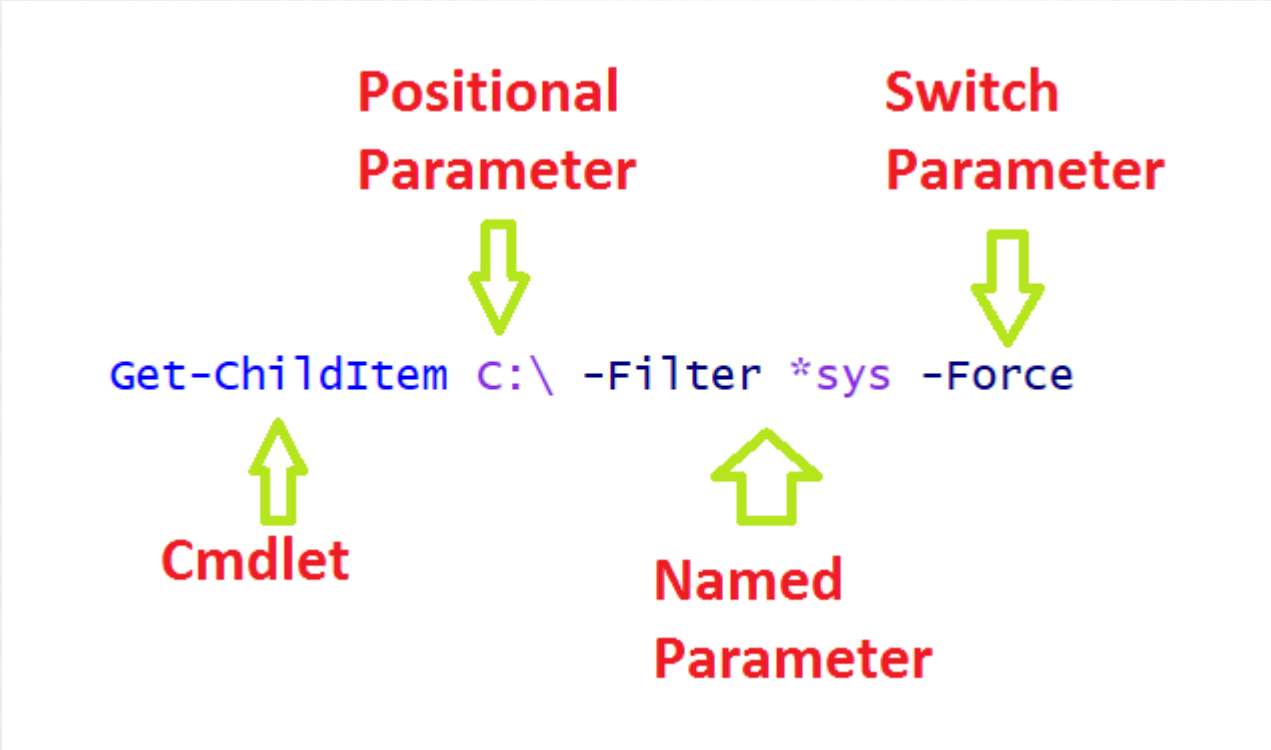
Demo Get-Verb

How to read the PowerShell language

WINDOWS

- Parameters
- Positional parameters
- Arguments

PowerShell language



WINDOWS

Demo Get-ChildItem

PowerShell tools

- Cmdlets
- Functions
- Aliases
- External tools

WINDOWS

Cmdlets & Functions

- The core of PowerShell
- Structured output
- Pipeline aware

Aliases

- Provide familiarity
- Short hand notation
- Customizability
- Not just cmdlets, parameters can also have an alias

External tools

- Can be executed in PowerShell like a command prompt
- Output can be stored and manipulated using PowerShell
- Does not adhere to PowerShell Language

WINDOWS

Help and command discovery

- Get-Help
- Get-Command
- Get-Member

WINDOWS

Demo PowerShell Help

PowerShell objects

- Everything in PowerShell is an object
- The console displays objects
- Objects can be displayed, stored, converted or exported

PowerShell pipeline

- Can be used to transfer objects
- The output from one cmdlet can directly be used in the next cmdlet
- Output is transferred on a first out-first

WINDOWS

Demo PowerShell Pipeline