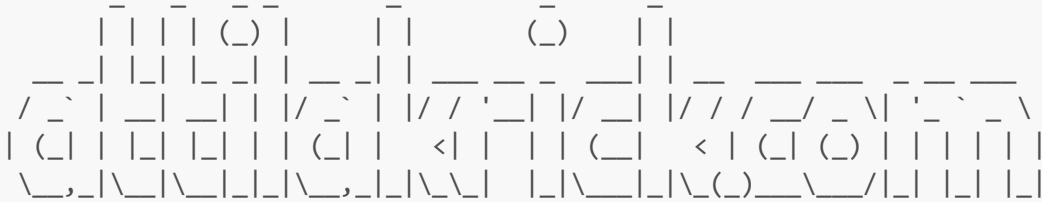


# Handout Terminologie und TOP 10-Listen

Powered by



----- HTTPS://WWW.ATTILAKRICK.COM -----

KEEP CALM and use GET-HELP!

## Terminologie der PowerShell

### Cmdlet und Alias

```

┌BZ┐──────────────────────────────────────────────────────────────────────────────────────────────────┐
Get-ChildItem -Path 'C:\Windows' -File | Select-Object 'Name', 'Length'
└C┐────────┐└PAP┐────────┐└SP┐└PL┐└C┐────────┐└PAP┐────────┐
└T┐└N┐────────┐└P┐└A┐────────┐└T┐└N┐────────┐└A┐────────┐

```

```

gci
ls
dir
...
└┐└┐

```

### PowerShell Provider & PowerShell Drive

```

Get-ChildItem -Path 'C:\'           # FileSystem (Dateisystem)
Get-ChildItem -Path 'Env:\'         # Environment (Windows Umgebungsvariable)
Get-ChildItem -Path 'HKCU:\'        # HKEY_CURRENT_USER Registry (Windows
Registrierungsdatenbank)
Get-ChildItem -Path 'HKLM:\'        # HKEY_LOCAL_MACHINE Registry (Windows
Registrierungsdatenbank)
Get-ChildItem -Path 'Cert:\'        # Certificate (Windows Zertifikatsspeicher)
Get-ChildItem -Path 'Variable:\'    # PowerShell Variable
└PD┐────────┐└PP┐────────┐

```

### Operator

```

┌W┐ ┌O┐ ┌W┐
└──┘ └──┘ └──┘
"Hallo Würzburg!" -like "*Würzburg?"
┌*┐ ┌?┐

```

## Variable

```

$Planet = "Pluto"
┌V┐ ┌O┐ ┌W┐

```

## Objektorientierte Programmierung

```

$MyNotePad = [System.Diagnostics.Process]::Start('notepad')
┌OB┐ ┌TN┐ ┌ME┐
└──┘ └──┘ └──┘
┌O┐ ┌O┐

$MyNotePad.StartTime
┌OB┐ ┌MP┐
└──┘ └──┘
┌O┐

$MyNotePad.WaitForExit(2500)
┌OB┐ ┌MM┐
└──┘ └──┘
┌O┐ ┌MMA┐

```

## Erklärung der Kürzel

KRZL	ABKÜRZUNG	BESCHREIBUNG
*, ?	Wildcards	Platzhalterzeichen
A	Argument	Objekt, das einem Parameter zugeordnet wird; P kann optional sein; => PAP
BZ	Befehlszeile	Cmdlet-Verkettungen mittels PL
C	Cmdlet	Command-Let; PowerShell-Befehl der eine kleine Teilaufgabe ausführt
L	Alias	Definiertes Kürzel für ein Cmdlet. Siehe ""Get-Alias""
ME	MemberName	Member ist der Überbegriff für Methode, Property, Event, etc.
MP	Property	Eigenschaft kann gelesen (get) und/oder geschoben (set) werden; s. ME
MM	Methode	Eine Methode kann mit Argumenten ausgeführt werden; s. ME
MMA	Methode-Argument	Argumente die einer Methode übergeben werden können; s. MM
N	Noun	Nomen, Cmdlet-Tätigkeitsbereich

KRZL	ABKÜRZUNG	BESCHREIBUNG
O	Operator	Ein O da kein Cmdlet vorangestellt ist; O muss nicht mit - beginnen; => P
OB	Object	Mit einem OB kann interagiert werden.
P	Parameter	Ein P da ein Cmdlet vorangestellt ist; => PAP
PD	PowerShell Drive	Ein Laufwerk in dem Default-cmdlets anwendung finde, z.B.: <code>Get-ChildItem</code>
PAP	Parameter-Argument-Paar	Zu einem P gehört immer ein A; Ausnahme SP
PL	Pipeline	Cmdlet-Ausgabe-Objekt (li. vom PL) wird zum Cmdlet-Eingabe-Objekt (re. von PL)
PD	PowerShell Provider	Eine Schnittstelle zwischen div. DB ähnliche Strukturen und PowerShell; s. PD
SP	Switch-Parameter	SP wird gesetzt, um etwas einzuschalten; SP enthalten keine A.
T	Verb	Cmdlet-Tätigkeit
TN	TypeName	Type ist der Überbegriff für Class, Enum, etc.
W	Value	Wert, Text, oder Objekt

## TOP 10

### TOP 10 PowerShell-Cmdlets

CMDLET	BESCHREIBUNG / BEISPIEL
<code>Find-Command</code>	Cmdlets in <a href="http://www.PowerShellGallery.com">www.PowerShellGallery.com</a> finden: <code>Find-Command Get-EuroExchange</code>
<code>Find-Module</code>	Module in <a href="http://www.PowerShellGallery.com">www.PowerShellGallery.com</a> finden: <code>Find-Module AKPT</code>
<code>ForEach-Object</code>	Bearbeiten von Pipeline-Objekten: <code>1..10   foreach {"192.168.0.\$_"}</code>
<code>Get-Command</code>	Installierte Cmdlets, Functions, Aliase und Applications finden (Alias <code>gcm</code> ): <code>Get-Command -Noun Object ; Get-Command -Module ScheduledTasks</code>
<code>Get-Help</code>	Anzeige von Hilfe für Cmdlet, <code>about_</code> : <code>Get-Help about_if ; Get-Help Get-Member -ShowWindow</code>
<code>Get-Member</code>	Objekt-Analyse (Alias <code>gm</code> ): <code>Get-Process   Get-Member</code>
<code>Get-Module</code>	Installierte bzw. geladene Module listen: <code>Get-Module -List ; Get-Module</code>

<b>CMDLET</b>	<b>BESCHREIBUNG / BEISPIEL</b>
Group-Object	Pipeline-Objekte gruppieren: <code>Get-ChildItem C:\windows -File   Group-Object Extension</code>
Measure-Object	Anzahl, Summe, Min, Max und Durchschnitt von Pipeline-Objekte-Eigenschaften ermitteln.: <code>Get-ChildItem C:\windows -File   measure -Sum -Min -Max -Avg</code>
Out-GridView	Pipeline-Objekte in einer GUI ausgeben: <code>Get-Process   Out-GridView</code>
Select-Object	<i>Taschenmesser</i> für Pipeline-Objekte: <code>Get-Process Name, WorkingSet64 -Skip 2 -First 5</code>
Show-Command	Cmdlets über eine GUI finden und bedienen: <code>Show-Command ; Show-Command Get-Command</code>
Sort-Object	Pipeline-Objekte sortieren.: <code>Get-Process   Sort-Object WorkingSet64 -Desc</code>
Where-Object	Filtern von Pipeline-Objekten: <code>Get-Process   where Company -like "Microsoft*"</code>

## TOP 10 PowerShell-Operatoren & -Sonderzeichen

<b>OPERATOR</b>	<b>BESCHREIBUNG: BEISPIEL</b>
= += -= ...	Zuweisung: <code>\$a = @() ; \$a += 'Hallo Welt!'</code>
+ - * / % ...	Arithmetik: <code>10 + 4</code>
-lt -le -eq -ne -ge -gt	Vergleich: <code>5 -gt 4</code>
-like -notLike	Wildcard-Muster-Vergleich (*, ?, ...): <code>'Report_21.log' -like 'Report_*.log'</code>
-match -notmatch	Regex-Muster-Vergleich: <code>'Report_21.log' -match '^Report_[0-9]{2,2}\.log\$'</code>
-in -notin -contains -notContains	Element(e) in/enthält Array: <code>10 -in 12,10,12</code>
. &	.PS1-Ausführung: <code>&amp; '\\srv00\Projekt X\MachWas.ps1'</code>
\$objekt.MemberName	. trennt Objekt vom Member: <code>\$PSVersionTable.PSVersion</code>
[Type]::MemberName	:: trennt Type und statischen Member: <code>[DateTime]::Now</code>
-and -or ...	Logik: <code>1 -gt 0 -or 9 -lt 10</code>
-split	Text in Array-Elemente aufteilen: <code>'a-b-c' -split '-'</code>
-join	Array-Elemente zu einem String verbinden: <code>10,11,12 -join ';' </code>

OPERATOR	BESCHREIBUNG: BEISPIEL
-f	Formatierung: 'Hallo {1}! {0:dd.MMM.yy}' -f (Get-Date), 'Würzburg'
n..m	Zahlenbereich von >= n und <= m erzeugen: 10..20 ; 9..2
[n]m -as	ObjektA umwandeln in Type X: [datetime]'2020-03-13' ; '2020-03-13' -as [datetime]
-is	Ist ein Objekt A vom Typ X: 'Würzburg!' -is [string]

## TOP 10 PowerShell-Typen

TYPENAME	[ALIAS /] BESCHREIBUNG / BEISPIEL
[ADSI]	Active Directory Services Interface: [ADSI]'LDAP://cn=lustig, ou=ak, dc=abc, dc=local'
[Array]	@() / Auflistung von Objekten: \$ar = @(10, 12, 13) ; \$ar[1]
[Boolean]	[bool] / Boolean: \$true ; \$false
[Byte]	Ganzzahl, \$[0, 255]\$: 255
[CultureInfo]	Kulturelle Besonderheiten bzgl. Datum, Zeit und Zahlen: [CultureInfo]'de-DE' ; [CultureInfo]\$PsUiCulture
[DateTime]	Datum und Uhrzeit: [DateTime]'2020-03-15' ; [DateTime]::Today
[Decimal]	Dezimalzahl ; \$[-79.228.162.514.264.337.593.543.950.335, 2^{96}]\$: [Decimal]::Round(0.987665, 2)
[GUID]	Globally Unique Identifier: [Guid]::NewGuid() ; [Guid]'daed6ca3-8d91-4dc4-81b5-9cc122645b87'
[HashTable]	Wörterbuch das aus Name-Wert-Paaren besteht: @{Label='LengthKB' ; Expression={ \$_.Length / 1KB }}
[Int32]	[int] / 32 Bit-Ganzzahlen; \$[-2.147.483.648, +2.147.483.647]\$: 12 ; -4096
[Int64]	[long] / 64 Bit-Ganzzahlen; \$[-9.223.372.036.854.775.808, +9.223.372.036.854.775.807]\$: 4567 ; -5437485
[IPAddress]	IPv4 oder IPv6 Adresse: [IPAddress]'192.168.178.20'
[MailAddress]	E-Mail-Adresse: [MailAddress]'Attila Krick<info@attilakrick.com>'
[PhysicalAddress]	MAC-Adresse: [PhysicalAddress]'B8-31-B5-3D-17-EC'
[PSCustomObject]	Individuell erstellbares Objekt: [PSCustomObject]@{ Author = 'Attila Krick' }

<b>TYPENAME</b>	<b>[ALIAS /] BESCHREIBUNG / BEISPIEL</b>
[ScriptBlock]	PowerShell-Codeblock um Zeitpunkt der Erstellung von der Ausführung entkoppeln: <code>\$cb = { Get-Process }</code>
[SecureString]	Ein String, der verschlüsselt (Benutzer) im RAM liegt: <code>\$pwd = Read-Host -Prompt 'Passwort?' -AsSecureString</code>
[String]	Text: <code>'Hallo' ; "Hello";@'</code> Text 1 Text 2 '@
[URI]	Uniform Resource Identifier: <code>[URI]'https://www.attilakrick.com'</code>
[void]	Unterdrückt die Ausgaben, schneller als Out-Null: <code>[Void](1..1000)</code>
[Xml]	XML: <code>[xml]'&lt;vom&gt;2020-03-12&lt;/vom&gt;'</code>

## TOP 10 automatische PowerShell-Variablen

<b>VARIABLE</b>	<b>BESCHREIBUNG</b>
<code>\$_</code>	Das Pipeline-Objekt in einem {}-Block in der Pipeline-Verarbeitung
<code>\$?</code>	Status des letzten Befehls
<code>\$env:...</code>	Umgebungsvariablen; s. Autovervollständigung
<code>\$Error</code>	Enthält die letzten 256 aufgetretenen Fehler
<code>\$ErrorActionPreference</code>	Gibt die Aktion an, die ausgeführt wird, wenn eine Fehlermeldung übermittelt wird
<code>\$ErrorView</code>	Legt den Ansichtsmodus fest, der beim Anzeigen von Fehlern verwendet werden soll
<code>\$false</code>	Boolean False
<code>\$HOME</code>	Ordner mit dem Profil des aktuellen Benutzers
<code>\$Host</code>	Ein Verweis auf den Host des aktuellen Runspace
<code>\$null</code>	Verweise auf die Nullvariable geben immer den Nullwert zurück. Zuweisungen haben keine Auswirkung.
<code>\$PSHome</code>	Übergeordneter Ordner der Hostanwendung des aktuellen Runspace
<code>\$PSCulture</code>	Kultur der aktuellen PowerShell-Sitzung
<code>\$PSEdition</code>	Editions-Informationen für die aktuelle PowerShell-Sitzung
<code>\$PSVersionTable</code>	Versionsinformationen für die aktuelle PowerShell-Sitzung
<code>\$PROFILE</code>	Enthält den Pfad der aktuellen Profil-Datei

VARIABLE	BESCHREIBUNG
<code>\$true</code>	Boolean True
<code>\$WarningPreference</code>	Gibt die Aktion an, die ausgeführt wird, wenn eine Warnmeldung übermittelt wird

## TOP 10 PowerShell-Datei-Erweiterung

Alle PowerShell-Datei-Typen sind textbasiert und sollten UTF8BOM kodiert sein.

EXTENSION	BEDEUTUNG	Beschreibung
.PS1	PowerShell Script Version 1.0	Ausführbare Skriptdatei die den Ausführungsrichtlinien unterliegen.
.PS1XML	PowerShell Format- and Typedefinition	Beschreibt Typen und deren Standard-Ausgabe-Aussehen
.PSC1	PowerShell Console Version 1.0	Exportierte PowerShell Host Konfigurationseinstellungen
.PSD1	PowerShell Definition Version 1.0	Modul-Manifest-Datei mit Eckdaten eines Modules
.PSM1	PowerShell Module Version 1.0	Modul-Skriptdatei, die beim Laden eines Modules ausgeführt wird, soweit es die Ausführungsrichtlinien es zulassen.
.PSRC	PowerShell Role Capability	Rollenfunktions-Datei die für die Sicherheit der PowerShell benötigt wird (JEA)
.PSSC	PowerShell Session Configuration	Sitzungskonfigurations-Datei die für die Sicherheit der PowerShell benötigt wird (JEA)

## Verweise, Links

- [Monatliche Herausforderungen](#) und Rätsel rund um PowerShell meistern von Iron Scripter
- [PowerShell ORG](#)
- [What's New in PowerShell 7.0](#)
- Attila Krick
  - [PowerShell Fachartikel und weitere Guide](#)
  - [PowerShell-Videos im YouTube-Kanal](#)
  - [Schulung, Workshop und Training](#)
  - [Twitter](#) und [LinkedIn](#)
- [Offizielle Microsoft PowerShell Produktdokumentation](#)
- [Wikipedia PowerShell Artikel](#)