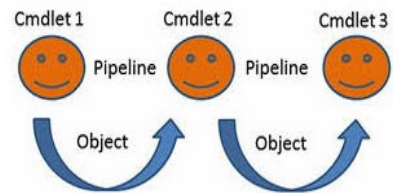


Aufgabenblatt 1

Ausgangssituation

1. Immer wieder müssen Sie überprüfen, ob bestimmte Programme auf einem Rechner laufen. Diese lästige Fleißarbeit lässt sich bestimmt mit PowerShell automatisieren.
2. Welche Updates sind wo installiert?



Aufgabe 1 – Was sind Pipes

Im „Text1 – PowerShell objects, piping and filtering“ werden Pipes beschrieben.

- a) Lesen Sie die Beschreibungen durch.
- b) Schreiben Sie mit einem Satz auf, was man unter Pipes versteht.
- c) Welches Zeichen wird bei PowerShell zur Bildung von Pipes genutzt?
- d) Schreiben Sie zwei Beispiele für Pipes auf?
- e) Was zeichnet laut dem Artikel Objekte aus?
- f) Wie kann man Cmdlets zum Filtern häufig am Namen erkennen?
- g) Was steckt hinter \$_?
- h) Wie kann man die Filterung mit Hilfe des Where-Object-Cmdlets häufig vermeiden und welchen Vorteil bietet dies?
- i) Wofür kann man das Foreach-Object-Cmdlet verwenden?

Allgemeine Recherche über den Text hinaus:

- j) Welche Cmdlets zum Filtern gibt es?
- k) Welche Vergleichsoperatoren gibt es?

Aufgabe 2 - Fingerübung Pipes

Erstellen Sie die folgenden PowerShell-Pipes. Nutzen Sie hierfür die PowerShell-ISE.

a)	Zeigen Sie alle Dienste an, die mit dem Buchstaben s beginnen und gerade laufen	aufgabe_ls2_01.ps1
b)	Zeigen Sie an, wie viele Dienste gerade laufen und wie viele nicht.	aufgabe_ls2_02.ps1
c)	Vom aktuellen Verzeichnis sollen für alle Dateien der Dateiname und die Dateigröße nach Größe sortiert in eine CSV-Datei geschrieben werden. Anschließend soll sich das Ergebnis mit Excel angesehen werden.	aufgabe_ls2_03.ps1
d)	Wie viele Dateien mit der Endung .exe weist Ihr Windows-Verzeichnis auf? Wie groß sind die Dateien durchschnittlich, maximal und minimal? Wie groß sind sie insgesamt?	aufgabe_ls2_04.ps1

e)	Schreiben Sie die ProzessId, den Prozessnamen und die verbrauchte CPU-Zeit der 10 Prozesse, welche am meisten CPU-Zeit verbraucht haben in eine HTML-Datei. Die Liste soll nach CPU-Zeit sortiert sein. Sehen Sie sich das Ergebnis mit einem Text-Editor und einem Internet-Browser an.	aufgabe_ls2_05.ps1
f)	Wie viele mp3- und mp4-Dateien gibt es im Klassenverzeichnis? Wie groß sind sie insgesamt?	aufgabe_ls2_06.ps1
g)	Wie viele Dateien mit welcher Endung weist Ihr Windows-Verzeichnis (rekursiv) auf? Die Ausgabe soll Typ, Anzahl und Gesamtgröße ausgeben.	aufgabe_ls2_07.ps1
h)	Geben Sie alle Prozesse aus, die mehr als 10 MB Arbeitsspeicher belegen. Wie viel % des gesamten Arbeitsspeichers belegen sie zusammen? Wie viel % aller Prozesse sind dies?	aufgabe_ls2_08.ps1

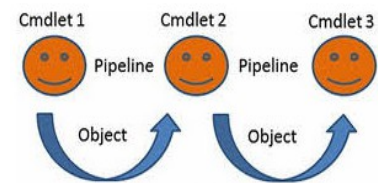
Aufgabe 3 - Befehlsübersicht

- Ergänzen Sie die Befehlsübersicht, aus LS1 um die in dieser Lernsituation neu hinzukommenden Befehle und Strukturen.
- Nehmen Sie in die Übersicht auch die wichtigsten Vergleichsoperatoren auf.



Aufgabe 4 - Prozess und Dienste-Übersicht

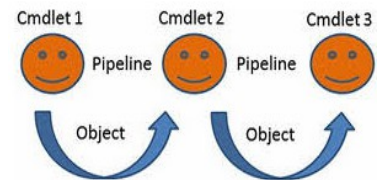
Immer wieder müssen Sie überprüfen, ob bestimmte Programme auf einem Rechner laufen. Diese lästige Fleißarbeit lässt sich bestimmt mit PowerShell automatisieren.



1. **Programm 1:** Erstellen Sie ein PowerShell-Skript, welches den Benutzer zeigt, welche Dienste laufen und welche nicht. Speichern Sie das Skript unter dem Namen **aufgabe_ls2_11.ps1** ab.
2. **Programm 2:** In zwei Textdateien stehen zeilenweise die Namen von Prozessen bzw. Diensten. Erstellen Sie ein PowerShell-Skript, welches prüft, ob die in der Textdatei aufgelisteten Prozesse und Dienste auf dem Rechner laufen. Speichern Sie das Skript unter dem Namen **aufgabe_ls2_12.ps1** ab.

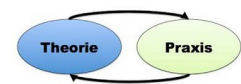
**Aufgabe 5 - Update-Übersicht**

Welche Updates sind wo installiert?



1. **Ausbaustufe 1:** Erstellen Sie ein PowerShell-Skript, welches den Benutzer nach einem Update fragt und anschließend prüft, ob dieses auf dem Rechner installiert ist. Speichern Sie das Skript unter dem Namen **aufgabe_ls2_13.ps1** ab.
2. **Ausbaustufe 2:** In einer Textdatei stehen zeilenweise die Namen von Updates. Erstellen Sie ein PowerShell-Skript, welches prüft, ob die in der Textdatei aufgelisteten Updates auf dem Rechner installiert sind. Speichern Sie das Skript unter dem Namen **aufgabe_ls2_14.ps1** ab.



(Theoretische Vertiefung)**Aufgabe 6 - Hintergrundwissen Prozesse, Dienste, Hintergrundprozesse, Programme**

Laufen auf dem PC Prozesse oder Programme? Was sind die geheimnisvollen Services? Hintergrundprozesse?

- Schreiben Sie kurze Arbeitsdefinitionen der 4 Begriffe auf.
- Was unterscheidet die Begriffe voneinander?
- Welche Beziehungen gibt es zwischen den Begriffen?

Orientieren Sie sich zuerst an der Windows-Welt. Wer will, kann die Begriffsthematik auf andere System-Welten übertragen.

Aufgabe 7 - Hintergrundwissen HTML, XML

- Was versteht man unter HTML? Wie sieht eine einfache HTML-Datei aus? Wo werden HTML-Dateien genutzt? Wie kann man aus PowerShell heraus HTML-Dateien erzeugen?
- Was versteht man unter XML? Wie sieht eine einfache XML-Datei aus? Wo werden XML-Dateien in PowerShell genutzt?

Aufgabe 8 - Hintergrundwissen PowerShell-Versionen

Seit wann gibt es PowerShell? Welche PowerShell-Version gehört standardmäßig zu welcher Windows-Version? Wie kann man updaten und welche Voraussetzungen müssen gegeben sein?

Aufgabe 9 - Microsoft Update-Philosophie

Wie müssen Sie vorgehen, wenn sie ein Window-Client-System auf dem neusten Stand halten müssen? Welche Arten von Updates gibt es? In welchem Rhythmus werden sie angeboten?

(Zusatzaufgaben)**Aufgabe 10 -**

to be done

**Aufgabe 11 -**

to be done