Powershell oefeningen

Filter, Sorting, Measure

Let op !!!!

Als er een Cmdllet bestaat voor het uitvoeren van de opdrachten dien je deze te verkiezen boven een ander commando.

# Van het CmdLet Get-Process commando:

## Toon de eerste vijf lijnen

Afbeelding met tekst, schermopname, Lettertype, software

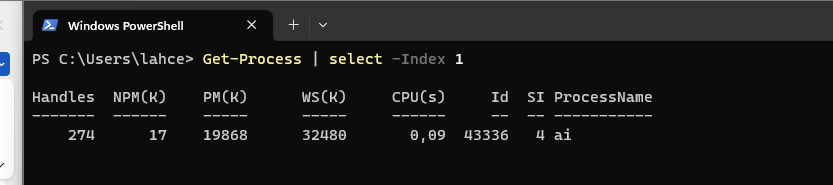
Automatisch gegenereerde beschrijving

## Toon de laatste vijf lijnen

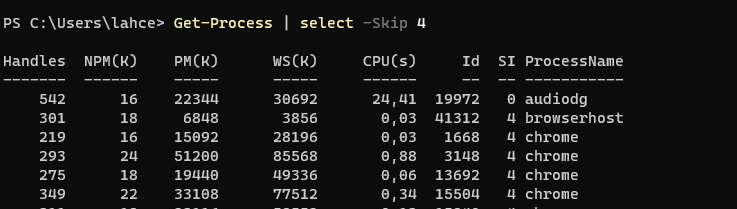
Afbeelding met tekst, schermopname, Lettertype

Automatisch gegenereerde beschrijving

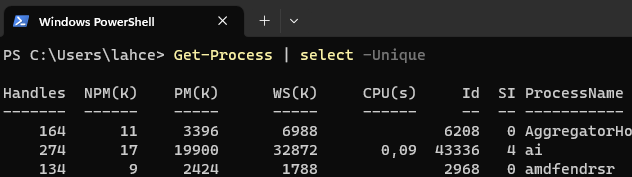
## Toon het 2de proces



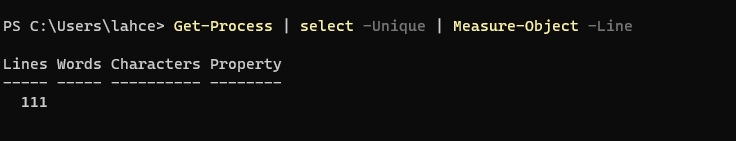
## Toon het overzicht van alle processen vanaf het vijfde proces



## Toon een overzicht van de unieke namen van de processen aanwezig op je systeem.



## Toon het aantal unieke procesnamen op je systeem.



Afbeelding met tekst, schermopname, Lettertype, zwart

Automatisch gegenereerde beschrijving

# Opdracht: Toon van de eerste 5 processen van je systeem.

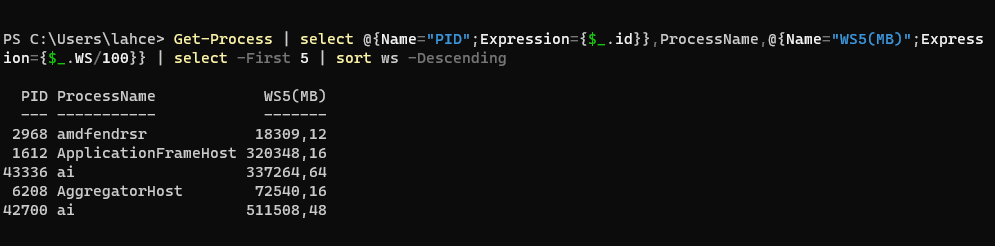
De procesnaam en vervolgens de grootte van de werkset van het proces (Property WS) in MB.  
Sorteer aflopend op de property WS  
Let op! De kolomnamen dienen overeen te komen met de verwachte output.

**Verwachte output: (waarden zijn verschillend afhankelijk van je systeem)**

**Afbeelding met tekst

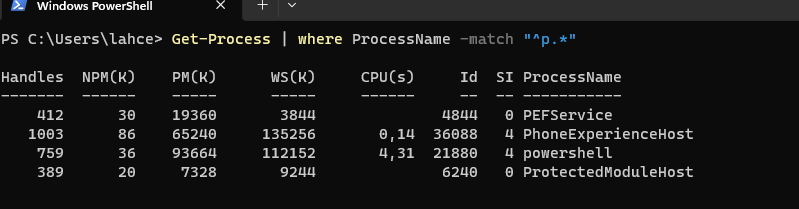
Automatisch gegenereerde beschrijving**

**Oplossing:**

****

# Toon alle processen op het host-systeem:

## Waarvan de display name begint met de p. Geef minimaal 2 mogelijke oplossingen.



Afbeelding met tekst, schermopname, Lettertype, software

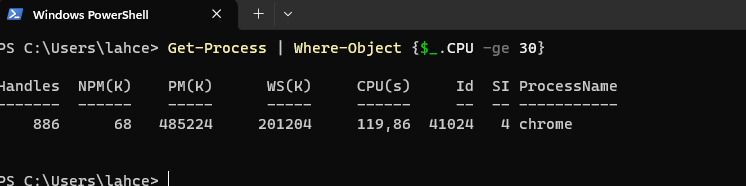
Automatisch gegenereerde beschrijving

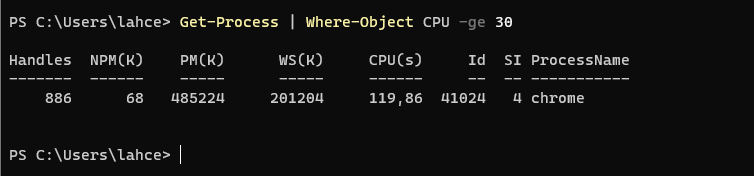
## Die een grafische window hebben

Afbeelding met tekst, schermopname, Lettertype

Automatisch gegenereerde beschrijving

## Die 30s of meer CPU tijd gebruikt hebben. Geef minimaal 2 mogelijke oplossingen.





## Geef de id en de procesnaam van alle processen, sorteer oplopend op naam en vervolgens aflopend op id.

Afbeelding met tekst, software, Multimediasoftware, schermopname

Automatisch gegenereerde beschrijving

Afbeelding met tekst, schermopname, Lettertype

Automatisch gegenereerde beschrijving

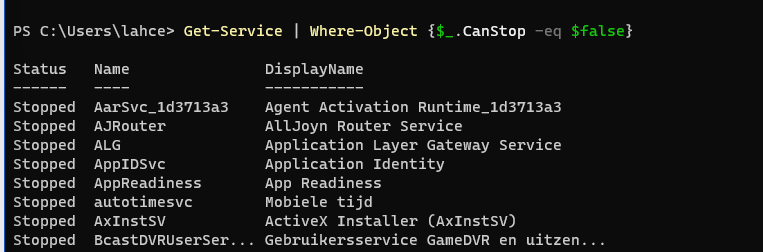
# Toon alle services:

## Die automatisch gestart worden tijdens het booten. Automatic

Afbeelding met tekst, schermopname, Lettertype

Automatisch gegenereerde beschrijving

## Die draaien op jullie host-systeem en die niet gestopt kunnen worden.



# Toon alle bestanden (files) van een map, inclusief de onderliggen mappen:

## Die 3 dagen oud of ouder zijn.

Get-ChildItem -Path "C:\Pad\Naar\Je\Map" -File -Recurse | Where-Object { $\_.LastWriteTime -le (Get-Date).AddDays(-3) }

## Die 3 dagen oud of ouder zijn en de extensie txt hebben.

Get-ChildItem -Path "C:\Pad\Naar\Je\Map" -File -Recurse | Where-Object { ($\_.LastWriteTime -le (Get-Date).AddDays(-3)) -and ($\_.Extension -eq '.txt') }

# Toon het aantal bestanden (files) van een map, inclusief de onderliggen mappen:

Afbeelding met tekst, schermopname, Lettertype

Automatisch gegenereerde beschrijving

## Die 3 dagen oud of ouder zijn en de extensie txt hebben.

(Get-ChildItem -Path "C:\Pad\Naar\Je\Map" -File -Recurse | Where-Object { ($\_.LastWriteTime -le (Get-Date).AddDays(-3)) -and ($\_.Extension -eq '.txt') }).Count

## Die als tweede letter het teken a hebben.

(Get-ChildItem -Path "C:\Users\student\Documents\" -File -Recurse | Where-Object { $\_.Name -match '^.a' }).count

# Toon alle executable items op uit de C:\Windows directory waarvan de file grootte groter of gelijk aan is aan 128 bytes

**Extra: recursief zoeken met dept 2**

**Afbeelding met tekst, schermopname, software

Automatisch gegenereerde beschrijving**

Doe eerst dita ls je wilt weten welke mogelijk heden er allemaal zijn

**get-childitem C:\Windows | Where-Object {$\_.Extension -eq ".exe" -and $\_.Length -ge 128}**

## Toon de 10 grootste files uit de C:\windows\system32 directory.

Get-ChildItem C:\Windows\System32 -File | sort Length -Descending | select -First 10

# Toon de Name, ID, de hoeveelheid wisselbaar geheugen en het niet-paginageheugen van alle processen op het host-systeem, sorteer aflopend op naam van het proces.

Get-Process | select Name,Id,NPM,WS | sort name -Descending

# Toon processen op het host-systeem, sorteer aflopend op procesnaam en aflopend op proces id (dus 1 commando).

get-process | select ProcessName,Id | sort ProcessName,Id -Descending

# Maak een script “HowManyFilesDir.ps1”.

**Doel:**

Toon hoeveel files en directories er in de C:\Windows directory staan.

Toon hoeveel processen er runnen op je systeem.

Voorzie commentaar volgens de richtlijnen van get-help.

**Output:**

**Afbeelding met tekst, buiten

Automatisch gegenereerde beschrijving**

$directory = (Get-ChildItem C:\Windows -Directory).count

$files = (Get-ChildItem C:\Windows -File).count

$processen = (Get-Process).count

Write-Output "information of path c:\windows" "file count: $files" "directory count: $directory" "aantal processen : $processen"

# Oefening Hotfix.

## Vind een cmdlet dat informatie weergeeft over geïnstalleerde hotfixes op Windowssystemen.

Oplossing : get-help -name “hotfixes” of Get-Command -Name \*hotfix\*

## Geef met het cmdlet dat je gevonden hebt een lijst van geïnstalleerde hotfixes weer.

Oplossing : Afbeelding met tekst, schermopname, Lettertype

Automatisch gegenereerde beschrijving

## Geef een lijst van geïnstalleerde hotfixes weer

* gesorteerd op de installatiedatum
* toon enkel de volgende properties
* de installatiedatum
* de gebruiker die de hotfix heeft geïnstalleerd
* het hotfix ID.

Oplossing : Get-HotFix | Select-Object InstalledOn, InstalledBy, HotFixID | Sort-Object -Property InstalledOn

# Maak een script “MetenIsWeten.ps1”.

**Voorbereiding:**

**Maak manueel (Notepad++) een bestand “mydata.txt” aan met de volgende inhoud:**

PXL

*De school met het netwerk.*

***Passie*** *stuurt aan vanuit ons hart, geef energie en goesting. Het wakkert de nieuwsgierigheid aan en nodigt uit om te onderzoeken, mogelijkheden te ontdekken en kansen te grijpen.*

**Output:**

Afbeelding met tekst

Automatisch gegenereerde beschrijving

Voorzie commentaar volgens de richtlijnen van get-help.

**Tip: Get-Content**

Oplossing:

$lijnen = (Get-Content C:\Users\student\Documents\mydata.txt | Measure-Object -Line).Lines

$woorden = (Get-Content C:\Users\student\Documents\mydata.txt | Measure-Object -Word).Words

$letters = (Get-Content C:\Users\student\Documents\mydata.txt | Measure-Object -Character).Characters

Write-Output "het bestand mydata.txt bevat:" "$lijnen lijn(en)" "$woorden woord(en)" "$letters letter(s)"

# Maak een script “FilterWithRegExp.ps1”.

**Voorbereiding:**

**Maak manueel (Notepad++) een bestand ‘regdata.txt” aan met de volgende inhoud:**

*PXL*

*De school met het netwerk!.*

*Passie stuurt aan vanuit ons hart, geeft energie en goesting.*

*Het wakkert de nieuwsgierigheid aan en nodigt uit om te onderzoeken,*

*mogelijkheden te ontdekken en kansen te grijpen.*

*Dit is een testtekst met verschillende elementen:*

*- E-mailadressen: contact@pxl.be, info@example.com*

*- URL's: http://www.pxl.be, https://www.example.com*

*- Datums: 01/15/2023, 30-09-2023, 10.05.22*

*- IP-adressen: 192.168.0.1, 10.0.0.1, 206.10.254.210/24*

*- Belgische telefoonnummers: 012/34.56.78, 0470/12.34.56, 012 23.24.55 en+32 476/20.20.15*

*- HTML-fragmenten: <p>Dit is een alinea.</p> <a href="https://www.example.com">Link</a>*

*- c:\windows\system32\BitLockerWizard.exe*

*Wist je dat Albert Einstein geboren is op 4/03/1879?*

*A B c C*

*PXL2023*

*PXL2024*

*PXL20*

*PXL202*

*test*

*dsdstestazdsds*

**Zorg dat de inhoud van deze tekst gekoppeld wordt aan de variabele $text (tip: Get-content)**

$text = Get-Content .\regdata.txt

------EINDE OEFENINGEN------

## Filter de volgende items en toon deze op het scherm inclusief aan het aantal gevonden overeenkomsten. (Tip: matches)

### URL's

### Datum

Datums uitgaande van het jaar 1000 tot en met 2999

### IPv4-adressen

### Belgische telefoonnummers

**Output:**

**Afbeelding met tekst, schermopname

Automatisch gegenereerde beschrijving**