```
Aufgabe 1
a. Get-Process
b. Get-Process | Format-List # Format-Table is default
c. Get-Process | Out-Host -Paging
d. Get-Process | Get-Member


Aufgabe 2
a. [services]
Get-Process | Where-Object { $_.SI -eq 0 }
b. [10MB RAM]
Get-Process | Where-Object { $_.WorkingSet -gt 10MB }
c. [CPU time]
Get-Process | Where-Object { $_.CPU -ge 0.5 }
d. [svchost]
Get-Process | Where-Object { $_.Name -eq "svchost" } | Measure-Object
e. [member]
Get-Process | Get-Member | Measure-Object | Format-Table -Property Count


Aufgabe 3
a.
[with S]
Get-Process | Where-Object { $_.Name -like "S*" }
[with S or I]
Get-Process | Where-Object { $_.Name -like "S*" -Or $_.Name -like "I*" }
[not with S]
Get-Process | Where-Object { $_.Name -notlike "S*" }
b.
[>5kB && <100kB]
Get-ChildItem C:\Windows -File | Where-Object { $_.Length -gt 5kB -And $_.Length
-lt 100kB }
[<10kB || win*]
Get-ChildItem C:\Windows -File | Where-Object { $_.Length -lt 10kB -Or $_.Name
-like "win*" }
[<10kB </> win*]
Get-ChildItem C:\Windows -File | Where-Object { $_.Length -lt 10kB -Xor $_.Name
-like "win*" }
# win.ini and WindowsUpdate.log are missing, because they satisfy both
conditions


Aufgabe 4
Get-Process -Name notepad
Stop-Process -Name notepad
Start-Process notepad # Positional Parameter -FilePath is omitted
Stop-Process -Id <PID>


Aufgabe 5
a. # List is displayed
b. Restart-Computer
c. Stop-Computer -WhatIf # WhatIf shows a textual description of the operation
without actually executing it.
```