

PRAKTIKUM PENGELOLAAN PROSES

DI LINUX DAN WINDOWS

Tujuan Praktikum

1. Memahami konsep pengelolaan proses pada sistem operasi Linux dan Windows.
2. Mampu melihat daftar proses yang berjalan.
3. Mampu mengirim sinyal dan menghentikan proses.
4. Mampu mengubah prioritas proses.
5. Mampu menerapkan konsep message passing melalui pipelining dan redirecting.

Catatan:

Screenshot untuk setiap hasil praktikum/soal yang dikerjakan.

I. Percobaan di Linux

1. Melihat Proses yang Berjalan

1. Jalankan terminal.
2. Gunakan perintah berikut untuk melihat daftar proses:

ps aux

```
root      725  0.1  0.6 468976 13568 ?        Ssl  07:10   0:00 /usr/libexec/udisks2/udisksd
root      763  0.0  0.1   6824  2688 ?        Ss   07:10   0:00 /usr/sbin/cron -f -P
syslog    787  0.0  0.3 222508  6144 ?        Ssl  07:10   0:00 /usr/sbin/rsyslogd -n
root      817  0.2  0.6 392096 12800 ?        Ssl  07:10   0:00 /usr/sbin/ModemManager
root      844  0.2  1.1 109676 23040 ?        Ssl  07:10   0:00 /usr/bin/python3 /usr/
root      878  0.0  0.2   6964  4736 tty1    Ss   07:10   0:00 /bin/login -p --
root     1025  0.0  0.0      0      0 ?        S    07:11   0:00 [psimon]
dipras    1027  0.7  0.5 20328 11392 ?        Ss   07:11   0:00 /usr/lib/systemd/systemd
dipras    1028  0.0  0.1  21148  3520 ?        S    07:11   0:00 (sd-pam)
dipras    1041  0.1  0.2   8656  5504 tty1    S    07:11   0:00 -bash
root     1090  0.0  0.0      0      0 ?        I<   07:11   0:00 [kworker/R-tls-s]
dipras    1092  100  0.2 10884  4480 tty1    R+   07:12   0:00 ps aux
dipras@dips:~$
```

- Gunakan perintah berikut untuk menampilkan daftar proses secara real-time:

Top

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
9	root	20	0	0	0	0	I	0.3	0.0	0:00.56	kworker/0:1-ata_sff
1093	dipras	20	0	11944	5888	3712	R	0.3	0.3	0:00.07	top
1	root	20	0	22080	13148	9436	S	0.0	0.7	0:01.96	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.01	kthreadd
3	root	20	0	0	0	0	S	0.0	0.0	0:00.00	pool_workqueue_release
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-rcu_g
5	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-rcu_p
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-slub_
7	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-netns
8	root	20	0	0	0	0	I	0.0	0.0	0:00.24	kworker/0:0-events

- Gunakan perintah berikut untuk menampilkan proses berdasarkan nama:

pgrep nano

```
dipras@dips:~$ pgrep nano
dipras@dips:~$ _
```

catatan: nano adalah text editor linux

Soal:

- Apa perbedaan output antara ps aux dan top?

```
dipras@dips:~$ 123
123: command not found
dipras@dips:~$ pgrep nano
dipras@dips:~$ ps aux | grep username
dipras      968  0.0  0.1  6544  2304 tty1      S+   07:18   0:00 grep --color=auto username
dipras@dips:~$
```

- Bagaimana cara menampilkan proses yang dimulai oleh user tertentu saja?

```
dipras@dips:~$ ps aux | grep alice
dipras      970  0.0  0.1  6544  2304 tty1      S+   07:19   0:00 grep --color=auto alice
dipras@dips:~$ _
```

- Bandingkan hasil ps aux dengan htop. Apa kelebihan htop dibanding ps aux?

ps aux:

- Outputnya berupa teks biasa yang tidak interaktif.
- Harus menggunakan kombinasi perintah seperti grep, awk, atau sort untuk memfilter data.

htop:

- Menampilkan proses dalam format visual yang lebih mudah dibaca.
- Bisa dinavigasi dengan keyboard (misalnya untuk menghentikan atau mengubah prioritas proses).

- Memiliki indikator warna untuk menunjukkan penggunaan CPU dan memori.
- Lebih user-friendly dibandingkan ps aux.

2. Mengirimkan Sinyal ke Proses

1. Jalankan proses berikut di background:

sleep 100 &

```
dipras@dips:~$ sleep 100 &
[1] 977
dipras@dips:~$
```

2. Gunakan perintah berikut untuk melihat PID dari proses sleep:

ps aux | grep sleep

```
dipras@dips:~$ ps aux | grep sleep
dipras      977  0.0  0.0  5684  1920 tty1      S   07:23   0:00 sleep 100
dipras      980  0.0  0.1  6544  2304 tty1      S+  07:24   0:00 grep --color=auto sleep
dipras@dips:~$
```

3. Kirim sinyal SIGSTOP untuk menghentikan sementara proses:

kill -STOP <PID>

```
dipras@dips:~$ kill-STOP<PID>
-bash: syntax error near unexpected token `newline'
[1]+  Done                  sleep 100
dipras@dips:~$
```

4. Lanjutkan kembali proses yang dihentikan:

kill -CONT <PID>

```
dipras@dips:~$ kill-CONT <PID>
-bash: syntax error near unexpected token `newline'
dipras@dips:~$
```

Soal:

- Jika sleep 100 & dijalankan tanpa &, apakah perbedaannya?

Jika Anda menjalankan sleep 100 tanpa menggunakan &, proses akan dijalankan di **foreground**, yang berarti terminal Anda akan diblokir dan menunggu hingga sleep 100 selesai. Anda tidak dapat menjalankan perintah lain sampai sleep selesai.

- Apa perbedaan antara SIGSTOP, SIGTERM, dan SIGKILL?
 - **SIGSTOP**: Menangguhkan proses (pause). Tidak bisa diabaikan, tetapi bisa dilanjutkan dengan `SIGCONT`.
 - **SIGTERM**: Meminta proses untuk berhenti dengan cara yang baik. Proses bisa menangani atau mengabaikan sinyal ini.
 - **SIGKILL**: Memaksa menghentikan proses secara paksa. Tidak bisa ditangani atau diabaikan, proses langsung dihentikan.
- Bagaimana cara mengirim sinyal SIGINT ke proses tanpa mengetahui PID-nya?

Untuk mengirim sinyal **SIGINT** ke proses tanpa mengetahui **PID-nya**, Anda dapat menggunakan perintah **pkill** atau **killall** yang memungkinkan Anda mengirim sinyal berdasarkan nama proses.

3. Mematikan Proses

1. Jalankan gedit di terminal:

Gedit

```
dipras@dips:~$ gedit
Command 'gedit' not found, but can be installed with:
sudo snap install gedit # version 46.1, or
sudo apt install gedit # version 46.1-3
See 'snap info gedit' for additional versions.
dipras@dips:~$
```

2. Cari PID dari gedit:

pgrep gedit

```
dipras@dips:~$ pgrep gedit
dipras@dips:~$ _
```

3. Hentikan proses gedit:

kill <PID>

```
dipras@dips:~$ kill <PID>
-bash: syntax error near unexpected token `newline'
dipras@dips:~$
```

Soal:

- Apa yang terjadi jika kill digunakan tanpa PID?
- Jika perintah kill digunakan tanpa menyertakan PID (Process ID), maka perintah tersebut akan mengirimkan sinyal ke proses yang terhubung dengan ID proses default.

- Bagaimana cara menghentikan semua proses gedit tanpa menggunakan pgrep?

Untuk menghentikan semua proses gedit tanpa menggunakan pgrep, kamu bisa menggunakan perintah killall.

- Mengapa kill -9 <PID> lebih efektif dibanding kill <PID>?

kill -9 lebih "efektif" dalam hal menghentikan proses yang tidak merespons, meskipun bisa berisiko kehilangan data atau merusak status internal proses jika tidak digunakan dengan hati-hati.

4. Meningkatkan Prioritas Proses

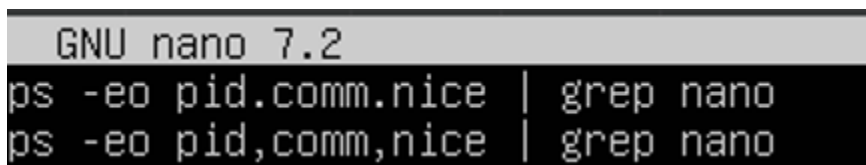
1. Jalankan proses gedit dengan prioritas rendah:

nice -n 10 nano



2. Lihat prioritasnya dengan:

ps -eo pid,comm,nice | grep nano



3. Tingkatkan prioritas proses nano:

renice -5 -p <PID>

```
renice -5 -p <pid>  
renice -5 -p <PID>
```

Soal:

- Apa yang terjadi jika nilai nice lebih kecil dari -20?

Jika nilai nice lebih kecil dari -20, itu tidak diizinkan oleh sistem dan akan diberikan batasan. Artinya, kamu tidak bisa memberikan nilai nice yang lebih rendah dari -20, dan jika mencoba melakukannya, sistem akan menampilkan pesan kesalahan (error) yang menunjukkan bahwa nilai yang diberikan tidak valid.

- Bisakah user biasa mengubah prioritas menjadi -10? Mengapa?

user biasa tidak dapat mengubah prioritas proses menjadi nilai negatif (seperti -10). karena mengatur prioritas negatif memberikan prioritas lebih tinggi kepada proses, yang bisa berdampak besar terhadap kinerja sistem jika digunakan secara sembarangan oleh pengguna biasa.

- Bagaimana cara meningkatkan prioritas semua proses yang dimiliki oleh user tertentu?

Untuk meningkatkan prioritas semua proses yang dimiliki oleh user tertentu, kamu bisa menggunakan perintah renice bersama dengan ID pengguna (UID) atau nama pengguna.

5. Message Passing (Pipelining dan Redirecting)

1. Gunakan pipelining untuk menampilkan hanya 5 proses pertama dari daftar proses:

ps aux | head -n 5

```
dipras@dips:~$ ps aux | head -n 5  
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND  
root         1  0.0  0.6 22064 13164 ?        Ss   17:31   0:01 /sbin/init  
root         2  0.0  0.0      0     0 ?        S    17:31   0:00 [kthreadd]  
root         3  0.0  0.0      0     0 ?        S    17:31   0:00 [pool_workqueue_release]  
root         4  0.0  0.0      0     0 ?        I<   17:31   0:00 [kworker/R-rcu_g]  
dipras@dips:~$
```

2. Redirect output daftar proses ke dalam file:

ps aux > daftar_proses.txt

ls daftar_proses.txt

```
dipras@dips:~$ ps aux > daftar_proses.txt
dipras@dips:~$ ls daftar_proses.txt
daftar_proses.txt
dipras@dips:~$
```

Soal:

- Bagaimana cara menyimpan output ps aux ke file sekaligus tetap menampilkannya di layar?

Anda bisa menggunakan perintah tee untuk mencapai ini. Perintah tee akan menampilkan output ke layar dan juga menyimpannya ke file.

- Jelaskan perbedaan antara > dan >> dalam redirection.

>: Digunakan untuk mengarahkan output ke file dan **menimpa** file tersebut jika file sudah ada.

>>: Digunakan untuk mengarahkan output ke file dan **menambahkan** output ke akhir file tersebut tanpa menimpa isinya.

- Buat perintah untuk menampilkan daftar proses, menyaring yang memiliki nama bash, dan menyimpannya ke file.

```
dipras@dips:~$ ps aux | grep bash > daftar_bash.txt
dipras@dips:~$ ps aux | grep bash >> daftar_bash.txt
dipras@dips:~$
```

II. Latihan Linux

1. Seorang user menjalankan perintah berikut di background:

ping google.com > hasil.txt &

```
dipras@dips:~$ ping google.com > hasil.txt&
[1] 1005
dipras@dips:~$ _
```

- Apa fungsi **&** dalam perintah tersebut?

Fungsi dari & dalam perintah ping google.com > hasil.txt & adalah untuk menjalankan perintah di **background**. Dengan menggunakan &, perintah tersebut tidak akan menghalangi terminal Anda untuk digunakan kembali setelah dijalankan.

- o Bagaimana cara menghentikan proses ini tanpa mengetahui PID-nya?

```
dipras@dips:~$ jobs
[1]+  Running                  ping google.com > hasil.txt &
dipras@dips:~$ kill %1
-bash: kill: %1: arguments must be process or job IDs
dipras@dips:~$
```

2. Anda ingin memastikan bahwa hanya satu instance nano yang berjalan. Jika ada lebih dari satu, matikan yang lain. Bagaimana perintahnya?

```
dipras@dips:~$ pgrep -x nano | tail -n +2 | xargs kill

Usage:
kill [options] <pid> [...]

Options:
<pid> [...]          send signal to every <pid> listed
-s, --signal <signal> specify the <signal> to be sent
-q, --queue <value>  integer value to be sent with the signal
-l, --list=[<signal>] list all signal names, or convert one to a name
-L, --table           list all signal names in a nice table

-h, --help           display this help and exit
-V, --version        output version information and exit

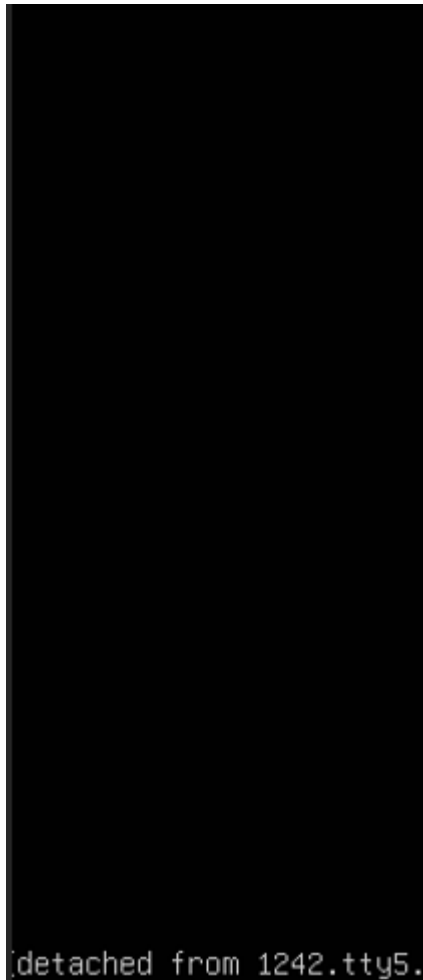
For more details see kill(1).
dipras@dips:~$
```

3. Sebuah server mengalami masalah karena load CPU terlalu tinggi. Bagaimana cara menemukan proses yang menghabiskan CPU terbanyak?

```
top - 01:23:51 up 40 min, 1 user, load average: 0.00, 0.00, 0.00
Tasks: 98 total, 1 running, 97 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.3 us, 0.3 sy, 0.0 ni, 99.0 id, 0.0 wa, 0.0 hi, 0.3 si, 0.0 st
MiB Mem : 1968.2 total, 1599.0 free, 286.2 used, 229.1 buff/cache
MiB Swap: 2048.0 total, 2048.0 free, 0.0 used, 1682.0 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1	root	20	0	21964	13180	9468	S	0.0	0.7	0:00.98	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	20	0	0	0	0	S	0.0	0.0	0:00.00	pool_workqueue_release
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-rcu_g
5	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/R-rcu_p

4. Anda menjalankan sleep 500 &, lalu menutup terminal. Apakah proses tetap berjalan? Bagaimana cara memastikan bahwa proses tetap berjalan setelah terminal ditutup?



5. Anda ingin melihat daftar proses yang telah berjalan lebih dari 1 jam. Bagaimana perintahnya?

```
error: unknown user-defined format specifier "com"

Usage:
ps [options]

Try 'ps --help <simple|list|output|threads|misc|all>'
or 'ps --help <s|l|o|t|m|a>'
for additional help text.

For more details see ps(1).
```

III. Percobaan di Windows

1. Melihat Proses yang Berjalan

1. Buka Command Prompt atau PowerShell.

- Gunakan perintah berikut untuk menampilkan daftar proses:

powershell

tasklist

```
svchost.exe      4972 Services      0      10,240 K
svchost.exe      3616 Console       1      16,352 K
dllhost.exe      1636 Console       1      15,824 K
WmiPrvSE.exe     1956 Services      0       9,728 K
cmd.exe          2296 Console       1       4,580 K
conhost.exe       820 Console       1      21,412 K
powershell.exe   4804 Console       1      58,124 K
MicrosoftEdgeUpdate.exe 3980 Services      0       1,648 K
MicrosoftEdgeUpdate.exe 4600 Services      0      10,844 K
tasklist.exe     5080 Console       1       8,872 K
PS C:\Users\Administrator>
```

- Gunakan perintah berikut untuk melihat detail proses tertentu:

powershell

tasklist | findstr notepad

```
PS C:\Users\Administrator> tasklist | findstr notepad
notepad.exe      4080 Console
PS C:\Users\Administrator>
```

2. Mengirimkan Sinyal ke Proses

- Jalankan Notepad.
- Cari PID dari Notepad dengan:

powershell

tasklist | findstr notepad

```
PS C:\Users\Administrator> tasklist | findstr notepad
notepad.exe      4080 Console
PS C:\Users\Administrator>
```

3. Hentikan proses Notepad:

powershell

taskkill /PID <PID>

```

PS C:\Users\Administrator> taskkill/PID<PID>
taskkill/PID<PID> : The term 'taskkill/PID<PID>' is not recognized as
operable program. Check the spelling of the name, or if a path was in
again.
At line:1 char:1
+ taskkill/PID<PID>
+ ~~~~~
+ CategoryInfo          : ObjectNotFound: (taskkill/PID<PID>:Stri
+ FullyQualifiedErrorId : CommandNotFoundException

PS C:\Users\Administrator>

```

3. Meningkatkan Prioritas Proses

1. Jalankan Notepad.
2. Ubah prioritasnya menjadi tinggi:

powershell

wmic process where name="notepad.exe" call setpriority 128

```

PS C:\Users\Administrator> wmic process where name="
ERROR:
Description = Invalid query
PS C:\Users\Administrator>

```

4. Message Passing (Pipelining dan Redirecting)

1. Gunakan pipelining untuk menampilkan hanya 5 proses pertama dari daftar proses:

powershell

tasklist | Select-Object -First 5

```

PS C:\Users\Administrator> tasklist | Select-Object -First 5

```

Image Name	PID	Session Name	Session#	Mem Usage
System Idle Process	0	Services	0	8 K
System	4	Services	0	140 K

```

PS C:\Users\Administrator>

```

2. Redirect output daftar proses ke dalam file:

powershell

tasklist > daftar_proses.txt

```
PS C:\Users\Administrator> tasklist > daftar_proses.txt
PS C:\Users\Administrator>
```

IV. Latihan di Windows

1. Tampilkan hanya proses yang dijalankan oleh user tertentu.

Image Name	PID	Session Name	Session#	Mem Usage
sihost.exe	2088	Console	1	27,256 K
svchost.exe	2644	Console	1	52,676 K
taskhostw.exe	2972	Console	1	16,172 K
ctfmon.exe	2428	Console	1	19,976 K
explorer.exe	1012	Console	1	98,356 K
smartscreen.exe	1324	Console	1	30,460 K
svchost.exe	3276	Console	1	17,468 K
StartMenuExperienceHost.e	3572	Console	1	72,176 K
RuntimeBroker.exe	3680	Console	1	27,880 K
SearchApp.exe	3804	Console	1	84,704 K
RuntimeBroker.exe	3972	Console	1	36,084 K
SecurityHealthSystray.exe	4224	Console	1	8,948 K
msedge.exe	4488	Console	1	89,344 K
msedge.exe	4616	Console	1	7,380 K
msedge.exe	4852	Console	1	32,328 K
msedge.exe	4860	Console	1	24,244 K
msedge.exe	4896	Console	1	17,664 K
RuntimeBroker.exe	5996	Console	1	13,468 K
SkypeApp.exe	6120	Console	1	29,552 K
RuntimeBroker.exe	4316	Console	1	11,976 K
OneDrive.exe	5048	Console	1	93,096 K
WWAHost.exe	5384	Console	1	43,112 K
ApplicationFrameHost.exe	984	Console	1	25,812 K

2. Cari PID dari proses chrome.exe.

```
PS C:\Users\Administrator> tasklist /FI "IMAGENAME eq chrome.exe"

Image Name          PID Session Name        Session#    Mem Usage
-----
chrome.exe           768 Console              1          145,828 K
chrome.exe           4492 Console              1           8,252 K
chrome.exe           4448 Console              1          36,896 K
chrome.exe           2964 Console              1          18,552 K
chrome.exe           4836 Console              1          48,564 K
chrome.exe           4612 Console              1          79,756 K
chrome.exe           2780 Console              1          67,912 K
chrome.exe           4192 Console              1          61,456 K
chrome.exe           4508 Console              1          29,276 K
PS C:\Users\Administrator>
```

3. Hentikan semua proses chrome.exe.

```
PS C:\Users\Administrator> taskkill /F /IM chrome.exe
SUCCESS: The process "chrome.exe" with PID 768 has been terminated.
SUCCESS: The process "chrome.exe" with PID 4492 has been terminated.
SUCCESS: The process "chrome.exe" with PID 4448 has been terminated.
SUCCESS: The process "chrome.exe" with PID 2964 has been terminated.
SUCCESS: The process "chrome.exe" with PID 4836 has been terminated.
SUCCESS: The process "chrome.exe" with PID 4612 has been terminated.
SUCCESS: The process "chrome.exe" with PID 2780 has been terminated.
SUCCESS: The process "chrome.exe" with PID 4192 has been terminated.
SUCCESS: The process "chrome.exe" with PID 4508 has been terminated.
PS C:\Users\Administrator>
```

4. Hentikan proses firefox.exe secara paksa.

```
PS C:\Users\Administrator> taskkill /F /IM firefox.exe
SUCCESS: The process "firefox.exe" with PID 2444 has been terminated.
SUCCESS: The process "firefox.exe" with PID 3272 has been terminated.
SUCCESS: The process "firefox.exe" with PID 2724 has been terminated.
ERROR: The process "firefox.exe" with PID 2924 could not be terminated.
Reason: There is no running instance of the task.
ERROR: The process "firefox.exe" with PID 3672 could not be terminated.
Reason: There is no running instance of the task.
ERROR: The process "firefox.exe" with PID 4644 could not be terminated.
Reason: There is no running instance of the task.
ERROR: The process "firefox.exe" with PID 2040 could not be terminated.
Reason: There is no running instance of the task.
ERROR: The process "firefox.exe" with PID 2704 could not be terminated.
Reason: There is no running instance of the task.
ERROR: The process "firefox.exe" with PID 2056 could not be terminated.
Reason: There is no running instance of the task.
ERROR: The process "firefox.exe" with PID 264 could not be terminated.
Reason: There is no running instance of the task.
PS C:\Users\Administrator>
```

5. Atur prioritas cmd.exe menjadi rendah.

```
PS C:\Users\Administrator> wmic process where "name='cmd.exe'" call setpriority 64
Executing (\\WIN-RLI7R5NMNSJ\ROOT\CIMV2:Win32_Process.Handle="5880")->setpriority()
Method execution successful.
Out Parameters:
instance of __PARAMETERS
{
    ReturnValue = 0;
};
PS C:\Users\Administrator>
```

6. Tampilkan prioritas dari semua proses yang berjalan.

WmiPrvSE.exe	0:00:00 N/A	5384 Services	0	23,400 K	Ur
E	0:00:01 N/A				
TrustedInstaller.exe	3764 Services	0	8,056 K	Ur	
TiWorker.exe	0:00:00 N/A	6136 Services	0	29,672 K	Ur
svchost.exe	0:00:03 N/A	3188 Services	0	11,352 K	Ur
tasklist.exe	0:00:00 N/A	4320 Console	1	9,292 K	Ur
or	0:00:00 N/A				

```
PS C:\Users\Administrator>
```

7. Jalankan dir C:\Windows lalu alihkan outputnya ke file output.txt.

```
PS C:\Users\Administrator> dir C:\Users > output.txt
PS C:\Users\Administrator> _
```

8. Gabungkan dua perintah: tasklist dan findstr chrome.

```
PS C:\Users\Administrator> tasklist | findstr chrome
PS C:\Users\Administrator> _
```

9. Anda menjalankan Notepad dan ingin memastikan bahwa tidak ada proses Notepad yang berjalan lebih dari 5 menit. Bagaimana cara mengeceknya?

```
PS C:\Users\Administrator> Get-Process notepad | Where-Object { $_.StartTime -lt (Get-Date).AddMinutes(-5)}
Handles      NPM(K)      PM(K)      WS(K)      CPU(s)      Id  SI ProcessName
-----
410          24       7000      31564       0.91     4080  1 notepad
PS C:\Users\Administrator> _
```

10. Sebuah proses bernama svchost.exe menggunakan terlalu banyak memori. Bagaimana cara menghentikan hanya proses svchost.exe tertentu yang menghabiskan lebih dari 500MB RAM?

```
PS C:\Users\Administrator> Get-Process svchost | Where-Object { $_.PrivateMemorySize -gt 500MB } | ForEach-Object { Stop-Process -Id $_.Id }
PS C:\Users\Administrator> _
```

11. Anda ingin menampilkan semua proses yang memiliki nama mengandung "chrome" dan menyimpan hasilnya ke file. Bagaimana caranya?

```
PS C:\Users\Administrator> tasklist | findstr chrome > hasil_proses_chrome.txt
PS C:\Users\Administrator> _
```

12. Anda menemukan bahwa Windows berjalan lambat karena banyak proses latar belakang. Buat perintah untuk menampilkan hanya proses dengan prioritas "Tinggi".

```
PS C:\Users\Administrator> Get-Process | Where-Object { $_.PriorityClass -eq "High" }
Handles      NPM(K)      PM(K)      WS(K)      CPU(s)      Id  SI ProcessName
-----
406          15       3548      20668       0.25     340  1 ctfmon
770          40      29436      55760      22.08     864  1 dwm
269          13       2660      13116       0.17     520  1 winlogon
PS C:\Users\Administrator> _
```

13. Sebuah aplikasi tidak merespons. Anda mencoba menghentikannya dengan taskkill, tetapi tidak berhasil. Apa kemungkinan penyebabnya dan bagaimana solusi lainnya?

```
PS C:\Users\Administrator> Stop-Process -Name "notepad" -Force
PS C:\Users\Administrator> _
```