

# **PRAKTIKUM PENGELOLAAN PROSES DI LINUX DAN WINDOWS**

**Nama: Ahmad Akhdan D**

**NIM:0920240006**

**Kelas: TRPL-1B**

## **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**

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

```

root      193  0.7  0.0  0  0 ?  I  04:00  0:01 [kworker/3:2H-kblockd]
root      210  0.0  0.0  0  0 ?  I  04:00  0:00 [kworker/R-kdmf1]
root      240  0.3  0.0  0  0 ?  I  04:00  0:00 [kworker/R-raids]
root      284  0.0  0.0  0  0 ?  S  04:00  0:00 [jbd2/dm-0-8]
root      285  0.0  0.0  0  0 ?  I  04:00  0:00 [kworker/R-ext4-]
root      317  0.1  0.0  0  0 ?  I  04:00  0:00 [kworker/3:2-events]
root      354  1.4  0.7  42280 15188 ? Ss 04:00  0:01 /usr/lib/systemd/systemd-journald
root      370  0.0  0.0  0  0 ?  I  04:00  0:00 [kworker/R-kmpat]
root      379  0.0  0.0  0  0 ?  I  04:00  0:00 [kworker/R-kmpat]
root      396  0.0  0.0  0  0 ?  I  04:00  0:00 [kworker/0:3-events]
root      397  0.0  0.0  0  0 ?  I  04:00  0:00 [kworker/0:4-events]
root      398  0.0  0.0  0  0 ?  I  04:00  0:00 [kworker/0:5-cgroup_destroy]
root      399  0.2  0.0  0  0 ?  I  04:00  0:00 [kworker/0:6-events]
root      400  0.0  0.0  0  0 ?  I  04:00  0:00 [kworker/0:7-cgwb_release]
root      401  0.0  0.0  0  0 ?  I  04:00  0:00 [kworker/0:8-events]
root      402  0.0  0.0  0  0 ?  I  04:00  0:00 [kworker/0:9-cgroup_destroy]
root      403  0.0  0.0  0  0 ?  I  04:00  0:00 [kworker/0:10]
root      410  0.0  1.3  289120 27136 ? Ssl 04:00  0:00 /sbin/multiplatform -d -s
root      421  0.2  0.0  0  0 ?  I  04:00  0:00 [kworker/0:11-ext4-rsv-conversion]
root      440  0.1  0.3  29056 7880 ? Ss 04:00  0:00 /usr/lib/systemd/systemd-udev
root      443  0.0  0.0  0  0 ?  S  04:00  0:00 [psimon]
root      528  0.0  0.0  0  0 ?  S  04:00  0:00 [irq/10-vmxgfx]
root      529  0.0  0.0  0  0 ?  I  04:00  0:00 [kworker/R-tt]
root      590  0.0  0.0  0  0 ?  S  04:00  0:00 [jbd2/sda2-8]
root      591  0.0  0.0  0  0 ?  I  04:00  0:00 [kworker/R-ext4-]
systemd+  657  0.0  0.5  21584 12800 ? Ss 04:00  0:00 /usr/lib/systemd/systemd-resolved
systemd+  663  0.0  0.3  91020 7680 ? Ssl 04:00  0:00 /usr/lib/systemd/systemd-timesyncd
root      813  0.0  0.0  0  0 ?  I  04:00  0:00 [kworker/R-cf80]
systemd+  893  0.1  0.4  18992 9472 ? Ss 04:00  0:00 /usr/lib/systemd/systemd-networkd
root      922  0.0  0.1  4236 2560 ? Ss 04:00  0:00 /usr/sbin/cron -f -P
message+  923  0.0  0.2  9772 5240 ? Ss 04:00  0:00 dbus-daemon --system --address=systemd: --no-
polkitd   928  0.0  0.4  383696 9504 ? Ssl 04:00  0:00 /usr/lib/polkit-1/polkitd --no-debug
root      936  0.0  0.4  17980 8704 ? Ss 04:00  0:00 /usr/lib/systemd/systemd-logind
root      939  0.1  0.6  46800 13312 ? Ssl 04:00  0:00 /usr/libexec/udisks2/udisksd
syslog    941  0.0  0.3  222500 6144 ? Ssl 04:00  0:00 /usr/sbin/rsyslogd -n -iNONE
root      984 15.2  0.0  0  0 ?  I  04:00  0:16 [kworker/u12:3-events_unbound]
root      986  0.0  1.1  106992 22784 ? Ssl 04:00  0:00 /usr/bin/python3 /usr/share/unattended-upgrad
root      993  0.0  0.2  6984 4736 tty1 Ss 04:00  0:00 /bin/login -p --
root     1014  0.0  0.6  392092 12416 ? Ssl 04:00  0:00 /usr/sbin/ModemManager
root     1053  0.0  0.0  0  0 ?  I  04:00  0:00 [kworker/0:2]
root     1281  0.0  0.0  0  0 ?  S  04:10  0:00 [psimon]
akhdan    1283  0.1  0.5  20340 11136 ? Ss 04:10  0:00 /usr/lib/systemd/systemd --user
akhdan    1284  0.0  0.1  21144 3644 ? S  04:10  0:00 (sd-pam)
akhdan    1310  0.0  0.2  8656 5504 tty1 S  04:10  0:00 -bash
root     1338  0.0  0.0  0  0 ?  I  04:10  0:00 [kworker/R-tls-s]
akhdan    1340  0.0  0.2  10884 4480 tty1 R+ 04:11  0:00 ps aux
akhdan@akhdan:~$

```

top

4. Gunakan perintah berikut untuk menampilkan proses berdasarkan nama:

```

top - 04:12:54 up 4 min, 1 user, load average: 0.10, 0.33, 0.17
Tasks: 156 total, 1 running, 155 sleeping, 0 stopped, 0 zombie
%Cpu(s):  0.0 us,  0.0 sy,  0.0 ni,100.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem : 1967.2 total, 1502.3 free, 329.4 used, 283.5 buff/cache
MiB Swap: 1927.0 total, 1927.0 free,  0.0 used, 1637.8 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR  S  %CPU  %MEM     TIME+ COMMAND
    1 root        20   0   22436   13140  9300  S   0.0   0.7   0:22.18 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   0:00.00 kworker/0:0-events
    9 root        20   0      0      0      0  I   0.0   0.0   0:00.00 kworker/0:1-cgroup_destroy
   10 root        0 -20    0      0      0  I   0.0   0.0   0:00.00 kworker/0:0H
   11 root        20   0      0      0      0  I   0.0   0.0   0:00.00 kworker/u8:0-ext4-rsv-conversion
   12 root        0 -20    0      0      0  I   0.0   0.0   0:00.00 kworker/R-mm_pe
   13 root        20   0      0      0      0  I   0.0   0.0   0:00.00 rcu_tasks_kthread
   14 root        20   0      0      0      0  I   0.0   0.0   0:00.00 rcu_tasks_rude_kthread
   15 root        20   0      0      0      0  I   0.0   0.0   0:00.00 rcu_tasks_trace_kthread
   16 root        20   0      0      0      0  S   0.0   0.0   0:00.00 ksoftirqd/0
   17 root        20   0      0      0      0  I   0.0   0.0   0:00.01 rcu_preempt
   18 root        rt    0      0      0      0  S   0.0   0.0   0:00.00 migration/0
   19 root       -51   0      0      0      0  S   0.0   0.0   0:00.00 idle_inject/0
   20 root        20   0      0      0      0  S   0.0   0.0   0:00.00 cpuhp/0
   21 root        20   0      0      0      0  S   0.0   0.0   0:00.00 cpuhp/1
   22 root       -51   0      0      0      0  S   0.0   0.0   0:00.00 idle_inject/1
   23 root        rt    0      0      0      0  S   0.0   0.0   0:00.25 migration/1
   24 root        20   0      0      0      0  S   0.0   0.0   0:00.76 ksoftirqd/1
   25 root        20   0      0      0      0  I   0.0   0.0   0:00.69 kworker/1:0-events
   26 root        0 -20    0      0      0  I   0.0   0.0   0:00.09 kworker/1:0H-kblockd
   27 root        20   0      0      0      0  S   0.0   0.0   0:00.00 cpuhp/2
   28 root       -51   0      0      0      0  S   0.0   0.0   0:00.00 idle_inject/2
   29 root        rt    0      0      0      0  S   0.0   0.0   0:00.18 migration/2
   30 root        20   0      0      0      0  S   0.0   0.0   0:00.37 ksoftirqd/2
   31 root        20   0      0      0      0  I   0.0   0.0   0:00.26 kworker/2:0-cgwb_release
   32 root        0 -20    0      0      0  I   0.0   0.0   0:00.00 kworker/2:0H-events_highpri
   33 root        20   0      0      0      0  S   0.0   0.0   0:00.00 cpuhp/3
   34 root       -51   0      0      0      0  S   0.0   0.0   0:00.00 idle_inject/3
   35 root        rt    0      0      0      0  S   0.0   0.0   0:00.20 migration/3
   36 root        20   0      0      0      0  S   0.0   0.0   0:00.02 ksoftirqd/3
   37 root        20   0      0      0      0  I   0.0   0.0   0:00.00 kworker/3:0-events
   38 root        0 -20    0      0      0  I   0.0   0.0   0:00.11 kworker/3:0H-kblockd
   39 root        20   0      0      0      0  I   0.0   0.0   0:00.00 kworker/u9:0-events_power_efficient
   40 root        20   0      0      0      0  I   0.0   0.0   0:00.66 kworker/u10:0-events_unbound
   41 root        20   0      0      0      0  I   0.0   0.0   0:00.08 kworker/u11:0-events_power_efficient
   42 root        20   0      0      0      0  I   0.0   0.0   0:00.15 kworker/u12:0-events_unbound
   43 root        20   0      0      0      0  S   0.0   0.0   0:00.00 kdevtmpfs

```

pgrep nano

```
akhdan@akhdan:~$ nano &
[1] 1359
akhdan@akhdan:~$ pgrep nano
1359

[1]+  Stopped                  nano
akhdan@akhdan:~$
```

catatan: nano adalah text editor linux

### Soal:

- Apa perbedaan output antara ps aux dan top?

**ps aux:** Menampilkan daftar semua proses yang sedang berjalan dalam bentuk statis.

**top:** Menampilkan daftar proses secara dinamis, diperbarui secara berkala

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

Menggunakan command `ps aux | grep username`

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

## 2. Mengirimkan Sinyal ke Proses

1. Jalankan proses berikut di background:

**sleep 100 &**

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

**ps aux | grep sleep**

3. Kirim sinyal SIGSTOP untuk menghentikan sementara proses:

**kill -STOP <PID>**

4. Lanjutkan kembali proses yang dihentikan:

## kill -CONT <PID>

```
akhdan@akhdan:~$ sleep 100 &
[2] 1397
akhdan@akhdan:~$ ps aux | grep sleep
akhdan      1397  0.0  0.1  5684  2048 tty1    S   04:41   0:00 sleep 100
akhdan      1399 95.2  0.1  6544  2304 tty1    S+  04:41   0:00 grep --color=auto sleep
akhdan@akhdan:~$ kill -STOP 1397
akhdan@akhdan:~$ ps aux | grep sleep
akhdan      1397  0.0  0.1  5684  2048 tty1    T   04:41   0:00 sleep 100
akhdan      1402 95.2  0.1  6544  2304 tty1    S+  04:41   0:00 grep --color=auto sleep

[2]+  Stopped                  sleep 100
akhdan@akhdan:~$ kill -CONT 1397
akhdan@akhdan:~$ ps aux | grep sleep
akhdan      1406  0.0  0.1  6544  2304 tty1    S+  04:43   0:00 grep --color=auto sleep
[2]-  Done                    sleep 100
akhdan@akhdan:~$
```

### Soal:

- Jika sleep 100 & dijalankan tanpa &, apakah perbedaannya?
  - Apa perbedaan antara SIGSTOP, SIGTERM, dan SIGKILL?
  - Bagaimana cara mengirim sinyal SIGINT ke proses tanpa mengetahui PID-nya?
- 

## 3. Mematikan Proses

1. Jalankan gedit di terminal:

gedit

2. Cari PID dari gedit:

pgrep gedit

3. Hentikan proses gedit:

kill <PID>

### Soal:

- Apa yang terjadi jika kill digunakan tanpa PID?
- Bagaimana cara menghentikan semua proses gedit tanpa menggunakan pgrep?
- Mengapa kill -9 <PID> lebih efektif dibanding kill <PID>?

## 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>**

**Soal:**

- Apa yang terjadi jika nilai nice lebih kecil dari -20?
- Bisakah user biasa mengubah prioritas menjadi -10? Mengapa?
- Bagaimana cara meningkatkan prioritas semua proses yang dimiliki oleh user tertentu?

## 5. Message Passing (Pipelining dan Redirecting)

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

**ps aux | head -n 5**

2. Redirect output daftar proses ke dalam file:

**ps aux > daftar\_proses.txt**

**ls daftar\_proses.txt**

```
akhdan@akhdan:~$ ps aux | head -n 5
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1   1.2   0.6 22436 13140 ?        Ss   04:08   0:30 /sbin/init
root         2   0.0   0.0      0     0 ?        S    04:08   0:00 [kthreadd]
root         3   0.0   0.0      0     0 ?        S    04:08   0:00 [pool_workqueue_release]
root         4   0.0   0.0      0     0 ?        I<   04:08   0:00 [kworker/R-rcu_g]
akhdan@akhdan:~$ ps aux > daftar_proses.txt
akhdan@akhdan:~$ ls daftar_proses.txt
daftar_proses.txt
akhdan@akhdan:~$
```

**Soal:**

- Bagaimana cara menyimpan output ps aux ke file sekaligus tetap menampilkannya di layar?
- Jelaskan perbedaan antara > dan >> dalam redirection.
- Buat perintah untuk menampilkan daftar proses, menyaring yang memiliki nama bash, dan menyimpannya ke file.

## II. Latihan Linux

1. Seorang user menjalankan perintah berikut di background:

**ping google.com > hasil.txt &**

- Apa fungsi **&** dalam perintah tersebut? **&** digunakan untuk menjalankan perintah di **background**, sehingga terminal tetap bisa digunakan tanpa harus menunggu perintah selesai.
  - Bagaimana cara menghentikan proses ini tanpa mengetahui PID-nya?  
Menggunakan command **"kill %1"**
2. Anda ingin memastikan bahwa hanya satu instance nano yang berjalan. Jika ada lebih dari satu, matikan yang lain. Bagaimana perintahnya?  
  
Menggunakan perintah command yaitu **"pkill -o nano"**
  3. Sebuah server mengalami masalah karena load CPU terlalu tinggi. Bagaimana cara menemukan proses yang menghabiskan CPU terbanyak? Untuk menemukan proses yang menghabiskan CPU terbanyak gunakan perintah berikut **"ps -eo pid,comm,%cpu --sort=-%cpu | head -n 10"**
  4. Anda menjalankan sleep 500 &, lalu menutup terminal. Apakah proses tetap berjalan? Tidak, proses akan **terhenti** ketika terminal ditutup
  5. Bagaimana cara memastikan bahwa proses tetap berjalan setelah terminal ditutup? Gunakan perintah berikut untuk memastikan tetap berjalan nohup **"sleep 500 & disown"**
  6. Anda ingin melihat daftar proses yang telah berjalan lebih dari 1 jam. Bagaimana perintahnya? Gunakan perintah berikut **ps -eo pid,etime,cmd --sort=etime**

## III. Percobaan di Windows

### 1. Melihat Proses yang Berjalan

1. Buka Command Prompt atau PowerShell.
2. Gunakan perintah berikut untuk menampilkan daftar proses:

**powershell**

**tasklist**

```
C:\Users\AXI00>tasklist
```

Image Name	PID	Session Name	Session#	Mem Usage
System Idle Process	0	Services	0	8 K
System	4	Services	0	4.868 K
Secure System	168	Services	0	48.916 K
Registry	248	Services	0	93.676 K
smss.exe	636	Services	0	1.108 K
csrss.exe	916	Services	0	5.328 K
wininit.exe	424	Services	0	6.304 K
services.exe	920	Services	0	9.908 K
LsaIso.exe	1044	Services	0	3.472 K
lsass.exe	1052	Services	0	21.836 K
svchost.exe	1168	Services	0	44.300 K
fontdrvhost.exe	1196	Services	0	3.096 K
WUDFHost.exe	1204	Services	0	16.960 K
svchost.exe	1308	Services	0	20.520 K
svchost.exe	1356	Services	0	9.184 K
WUDFHost.exe	1408	Services	0	13.176 K
WUDFHost.exe	1588	Services	0	7.260 K
svchost.exe	1740	Services	0	5.184 K
svchost.exe	1792	Services	0	11.900 K
svchost.exe	1804	Services	0	8.596 K
svchost.exe	1820	Services	0	11.544 K
svchost.exe	1984	Services	0	7.228 K
svchost.exe	2012	Services	0	16.344 K
svchost.exe	684	Services	0	9.920 K
svchost.exe	700	Services	0	10.496 K
svchost.exe	896	Services	0	11.408 K
svchost.exe	1016	Services	0	16.408 K

3. Gunakan perintah berikut untuk melihat detail proses tertentu:

**powershell**

**tasklist | findstr notepad**

```
C:\Users\AXI00>tasklist | findstr Notepad
Notepad.exe                9496 Console                8      101.536 K
```

## 2. Mengirimkan Sinyal ke Proses

1. Jalankan Notepad.
2. Cari PID dari Notepad dengan:

**powershell**

**tasklist | findstr notepad**

## 3. Hentikan proses Notepad:

**powershell**

**taskkill /PID <PID>**

### **3. Meningkatkan Prioritas Proses**

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

**powershell**

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

### **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\AXI00> 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	4.888 K

```
PS C:\Users\AXI00>
```

2. Redirect output daftar proses ke dalam file:

**powershell**

**tasklist > daftar\_proses.txt**

### **IV. Latihan di Windows**

1. Tampilkan hanya proses yang dijalankan oleh user tertentu.



```
C:\Users\AXI00>tasklist /FI "USERNAME eq AXI00"
```

Image Name	PID	Session Name	Session#	Mem Usage
=====	=====	=====	=====	=====
ipf_helper.exe	11868	Console	8	8.972 K
sihost.exe	4368	Console	8	39.312 K
svchost.exe	7484	Console	8	21.164 K
svchost.exe	12120	Console	8	40.916 K
taskhostw.exe	21136	Console	8	19.464 K
explorer.exe	6284	Console	8	217.356 K
svchost.exe	6560	Console	8	27.160 K
Widgets.exe	10760	Console	8	35.260 K
SearchHost.exe	704	Console	8	384.832 K
StartMenuExperienceHost.e	9372	Console	8	85.808 K
RuntimeBroker.exe	11640	Console	8	28.892 K
RuntimeBroker.exe	8400	Console	8	55.696 K
svchost.exe	21336	Console	8	10.920 K
dllhost.exe	8568	Console	8	17.540 K
ctfmon.exe	20572	Console	8	26.984 K
LockApp.exe	15068	Console	8	86.876 K
RuntimeBroker.exe	6316	Console	8	43.688 K
svchost.exe	4092	Console	8	15.080 K
PhoneExperienceHost.exe	13048	Console	8	136.876 K
FnKey.exe	17848	Console	8	81.096 K
SecurityHealthSystray.exe	9068	Console	8	10.496 K
RtkAudUService64.exe	17688	Console	8	13.456 K
steam.exe	9584	Console	8	60.464 K
steamwebhelper.exe	19156	Console	8	142.604 K
steamwebhelper.exe	2436	Console	8	11.896 K
steamwebhelper.exe	9984	Console	8	91.120 K
steamwebhelper.exe	16680	Console	8	25.660 K
steamwebhelper.exe	21320	Console	8	17.504 K
steamwebhelper.exe	13636	Console	8	346.756 K
Lively.exe	10676	Console	8	134.948 K
ms-teams.exe	5748	Console	8	29.208 K
RuntimeBroker.exe	10416	Console	8	11.712 K

2. Cari PID dari proses chrome.exe.
3. Hentikan semua proses chrome.exe.
4. Hentikan proses firefox.exe secara paksa.

```

C:\Users\AXI00>tasklist | findstr chrome.exe
chrome.exe             11616 Console             8      176.632 K
chrome.exe             10608 Console             8         9.320 K
chrome.exe             18224 Console             8     158.380 K
chrome.exe             10188 Console             8     45.900 K
chrome.exe             19328 Console             8     20.024 K
chrome.exe             18936 Console             8    191.076 K
chrome.exe             10424 Console             8     21.020 K
chrome.exe             18856 Console             8     18.088 K
chrome.exe              1288 Console             8     29.876 K

C:\Users\AXI00>taskkill /IM chrome.exe /F
SUCCESS: The process "chrome.exe" with PID 11616 has been terminated.
SUCCESS: The process "chrome.exe" with PID 10608 has been terminated.
SUCCESS: The process "chrome.exe" with PID 18224 has been terminated.
SUCCESS: The process "chrome.exe" with PID 10188 has been terminated.
SUCCESS: The process "chrome.exe" with PID 19328 has been terminated.
SUCCESS: The process "chrome.exe" with PID 18936 has been terminated.
SUCCESS: The process "chrome.exe" with PID 10424 has been terminated.
SUCCESS: The process "chrome.exe" with PID 18856 has been terminated.
SUCCESS: The process "chrome.exe" with PID 1288 has been terminated.

C:\Users\AXI00>taskkill /IM firefox.exe /F
SUCCESS: The process "firefox.exe" with PID 13036 has been terminated.
SUCCESS: The process "firefox.exe" with PID 3984 has been terminated.
SUCCESS: The process "firefox.exe" with PID 3492 has been terminated.
SUCCESS: The process "firefox.exe" with PID 13028 has been terminated.
SUCCESS: The process "firefox.exe" with PID 19512 has been terminated.

```

5. Atur prioritas cmd.exe menjadi rendah.
6. Tampilkan prioritas dari semua proses yang berjalan.

```

C:\Users\AXI00>wmic process where name="cmd.exe" CALL setpriority 64
Executing (\\DESKTOP-0USIBFH\ROOT\CIMV2:Win32_Process.Handle="17772")->setpriority()
Method execution successful.
Out Parameters:
instance of __PARAMETERS
{
    ReturnValue = 0;
};

C:\Users\AXI00>wmic process get name,priority
Name                                Priority
System Idle Process                 0
System                              8
Secure System                       8
Registry                            8
smss.exe                             11
csrss.exe                           13
wininit.exe                         13
services.exe                        9
lsass.exe                           8
svchost.exe                         8
fontdrvhost.exe                     8
WUDFHost.exe                        13
svchost.exe                         8
svchost.exe                         8

```

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

8. Gabungkan dua perintah: tasklist dan findstr chrome.

```
C:\Users\AXI00>dir C:\Windows > output.txt

C:\Users\AXI00>tasklist | findstr chrome
chrome.exe             15408 Console           8      184.980 K
chrome.exe              380 Console             8         9.200 K
chrome.exe             17200 Console           8     156.008 K
chrome.exe              8836 Console            8      43.996 K
chrome.exe             19196 Console           8      19.856 K
chrome.exe             17904 Console           8     194.324 K
chrome.exe             16852 Console           8      17.896 K
chrome.exe             11168 Console           8      16.852 K
chrome.exe              5772 Console            8      28.548 K
```

9. Anda menjalankan Notepad dan ingin memastikan bahwa tidak ada proses Notepad yang berjalan lebih dari 5 menit. Bagaimana cara mengeceknya? Gunakan perintah wmic berikut ***"wmic process where (name="notepad.exe" and CreationDate<="DATE-5min") get ProcessId,Name,CreationDate"***

10. Sebuah proses bernama svchost.exe menggunakan terlalu banyak memori. Bagaimana cara menghentikan hanya proses svchost.exe tertentu yang menghabiskan lebih dari 500MB RAM? Gunakan perintah ini ***"wmic process where (name="svchost.exe" and WorkingSetSize>524288000) call terminate"***

**524288000 bytes = 500MB**

11. Anda ingin menampilkan semua proses yang memiliki nama mengandung "chrome" dan menyimpan hasilnya ke file. Bagaimana caranya? Gunakan perintah tasklist berikut ***"tasklist / findstr chrome > chrome\_processes.txt"***

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

Ini perintahnya ***"wmic process where (priority=13) get Name,ProcessId,Priority"***

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

Penyebabnya adalah Proses mengalami "hang" atau "not responding". Proses memiliki hak akses lebih tinggi (misalnya berjalan sebagai SYSTEM). Proses dikunci oleh layanan lain atau kernel solusinya bisa hentikan menggunakan perintah wmic ***"wmic process where name='nama\_aplikasi.exe' call terminate"***