

# Lab 06

2024학년도 2학기 리눅스시스템

---

조교 이지원

lee.jiwon@sookmyung.ac.kr



SOOKMYUNG WOMEN'S UNIVERSITY



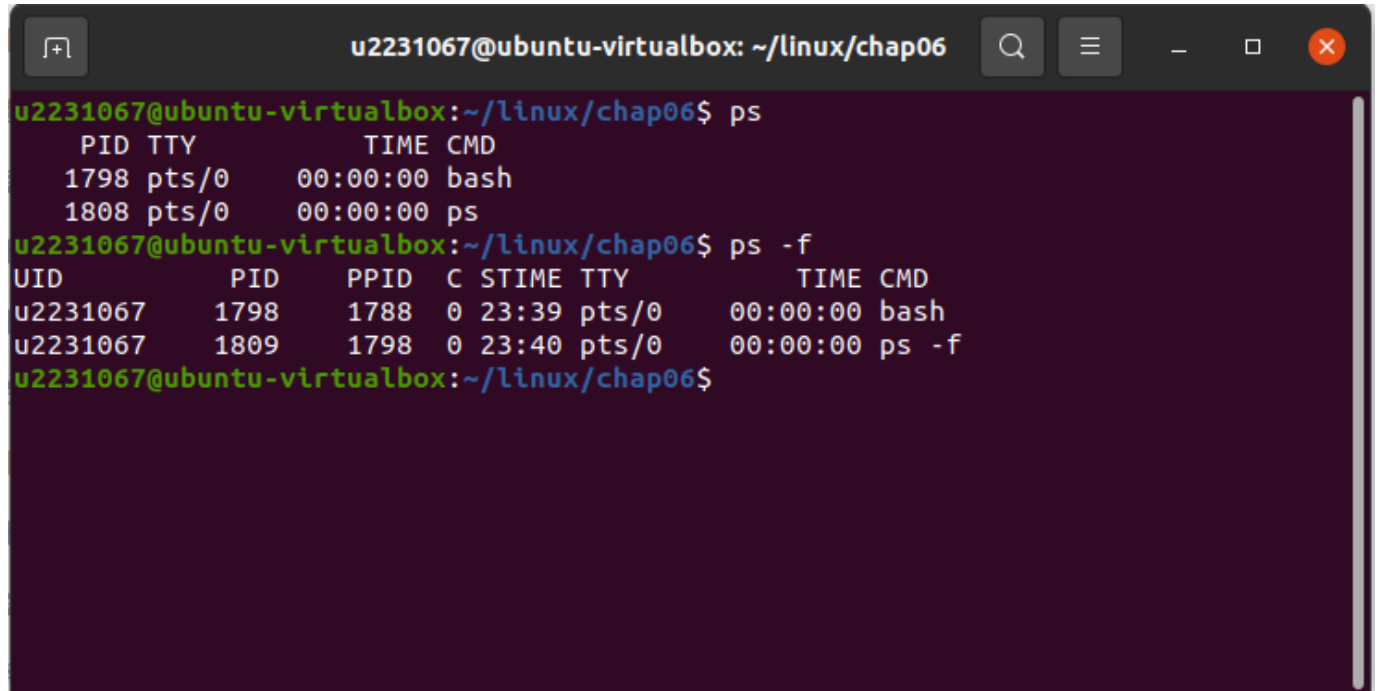
# Lab 06. 프로세스 및 작업제어 실습

1. ps 명령어
2. pgrep 명령어
3. 전면처리, 후면처리
4. 작업제어
5. 프로세스 기다리기

# 1. ps 실습

- 다음 명령을 실행하고 출력 결과에 대해 설명하시오.
  - ps 명령어가 무엇을 하는 명령어인지,
  - 각 옵션의 의미를 구체적으로 설명하기 바랍니다.

```
$ ps
$ ps -f
$ ps aux
$ ps -ef
```



A terminal window titled 'u2231067@ubuntu-virtualbox: ~/linux/chap06' showing the execution of 'ps' and 'ps -f' commands. The 'ps' command output shows two processes: 'bash' (PID 1798) and 'ps' (PID 1808). The 'ps -f' command output shows the same two processes with additional details like UID, PPID, and C status.

```
u2231067@ubuntu-virtualbox:~/linux/chap06$ ps
  PID TTY          TIME CMD
 1798 pts/0        00:00:00 bash
 1808 pts/0        00:00:00 ps
u2231067@ubuntu-virtualbox:~/linux/chap06$ ps -f
UID          PID    PPID  C STIME TTY          TIME CMD
u2231067      1798    1788  0  23:39 pts/0        00:00:00 bash
u2231067      1809    1798  0  23:40 pts/0        00:00:00 ps -f
u2231067@ubuntu-virtualbox:~/linux/chap06$
```

# 1. ps 실습

- 다음 명령을 실행하고 출력 결과에 대해 설명하시오.
  - ps 명령어가 무엇을 하는 명령어인지,
  - 각 옵션의 의미를 구체적으로 설명하기 바랍니다.

```
$ ps  
$ ps -f  
$ ps aux  
$ ps -ef
```

```
u2231067@ubuntu-virtualbox: ~/linux/chap06  
u2231067@ubuntu-virtualbox:~/linux/chap06$ ps aux  
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND  
root         1   0.5   0.7 168220  7844 ?        Ss   23:35   0:01 /sbin/init sp  
root         2   0.0   0.0      0     0 ?        S    23:35   0:00 [kthreadd]  
root         3   0.0   0.0      0     0 ?        I<   23:35   0:00 [rcu_gp]  
root         4   0.0   0.0      0     0 ?        I<   23:35   0:00 [rcu_par_gp]  
root         5   0.0   0.0      0     0 ?        I<   23:35   0:00 [netns]  
root         6   0.0   0.0      0     0 ?        I    23:35   0:00 [kworker/0:0-  
root         7   0.0   0.0      0     0 ?        I<   23:35   0:00 [kworker/0:0H  
root         8   0.0   0.0      0     0 ?        I    23:35   0:00 [kworker/0:1-  
root         9   0.1   0.0      0     0 ?        I    23:35   0:00 [kworker/u4:0  
root        10   0.0   0.0      0     0 ?        I<   23:35   0:00 [mm_percpu_wq  
root        11   0.0   0.0      0     0 ?        S    23:35   0:00 [rcu_tasks_ru  
root        12   0.0   0.0      0     0 ?        S    23:35   0:00 [rcu_tasks_tr  
root        13   0.0   0.0      0     0 ?        S    23:35   0:00 [ksoftirqd/0]  
root        14   0.0   0.0      0     0 ?        I    23:35   0:00 [rcu_sched]  
root        15   0.0   0.0      0     0 ?        S    23:35   0:00 [migration/0]  
root        16   0.0   0.0      0     0 ?        S    23:35   0:00 [idle_inject/  
root        17   0.0   0.0      0     0 ?        S    23:35   0:00 [cpuhp/0]  
root        18   0.0   0.0      0     0 ?        S    23:35   0:00 [cpuhp/1]  
root        19   0.0   0.0      0     0 ?        S    23:35   0:00 [idle_inject/  
root        20   0.0   0.0      0     0 ?        S    23:35   0:00 [migration/1]  
root        21   0.0   0.0      0     0 ?        S    23:35   0:00 [ksoftirqd/1]  
root        22   0.0   0.0      0     0 ?        I    23:35   0:00 [kworker/1:0-  
root        23   0.0   0.0      0     0 ?        I<   23:35   0:00 [kworker/1:0H  
root        24   0.0   0.0      0     0 ?        S    23:35   0:00 [kdevtmpfs]  
root        25   0.0   0.0      0     0 ?        I<   23:35   0:00 [inet_frag_wq  
root        26   0.0   0.0      0     0 ?        S    23:35   0:00 [kauditd]  
root        27   0.0   0.0      0     0 ?        S    23:35   0:00 [khungtaskd]  
root        28   0.0   0.0      0     0 ?        S    23:35   0:00 [oom_reaper]  
root        29   0.0   0.0      0     0 ?        I<   23:35   0:00 [writeback]  
root        30   0.0   0.0      0     0 ?        S    23:35   0:00 [kcompactd0]  
root        31   0.0   0.0      0     0 ?        SN   23:35   0:00 [ksmd]  
root        32   0.0   0.0      0     0 ?        SN   23:35   0:00 [khugepaged]  
root        37   0.0   0.0      0     0 ?        I    23:35   0:00 [kworker/1:1-  
root        79   0.0   0.0      0     0 ?        I<   23:35   0:00 [kintegrityd]  
root        80   0.0   0.0      0     0 ?        I<   23:35   0:00 [kblockd]  
root        81   0.0   0.0      0     0 ?        I<   23:35   0:00 [blkcg_punt_b  
root        82   0.0   0.0      0     0 ?        I<   23:35   0:00 [tpm_dev_wq]  
root        83   0.0   0.0      0     0 ?        I<   23:35   0:00 [ata_sff]  
root        84   0.0   0.0      0     0 ?        I<   23:35   0:00 [md]
```

# 1. ps 실습

- 다음 명령을 실행하고 출력 결과에 대해 설명하시오.
  - ps 명령어가 무엇을 하는 명령어인지,
  - 각 옵션의 의미를 구체적으로 설명하기 바랍니다.

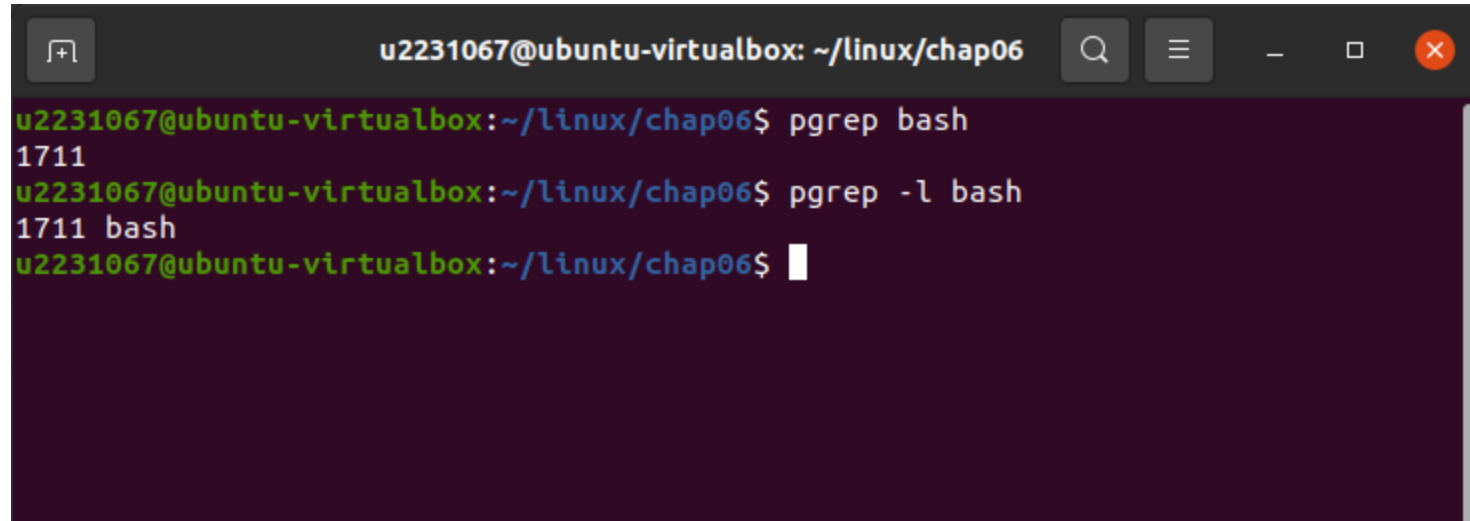
```
$ ps
$ ps -f
$ ps aux
$ ps -ef
```

```
u2231067@ubuntu-virtualbox: ~/linux/chap06
u2231067@ubuntu-virtualbox:~/linux/chap06$ ps -ef
UID        PID     PPID  C  STIME TTY          TIME CMD
root         1         0  0   23:35 ?        00:00:01 /sbin/init splash
root         2         0  0   23:35 ?        00:00:00 [kthreadd]
root         3         2  0   23:35 ?        00:00:00 [rcu_gp]
root         4         2  0   23:35 ?        00:00:00 [rcu_par_gp]
root         5         2  0   23:35 ?        00:00:00 [netns]
root         7         2  0   23:35 ?        00:00:00 [kworker/0:0H-events_highpri
root         9         2  0   23:35 ?        00:00:00 [kworker/u4:0-events_unbound
root        10         2  0   23:35 ?        00:00:00 [mm_percpu_wq]
root        11         2  0   23:35 ?        00:00:00 [rcu_tasks_rude_]
root        12         2  0   23:35 ?        00:00:00 [rcu_tasks_trace]
root        13         2  0   23:35 ?        00:00:00 [ksoftirqd/0]
root        14         2  0   23:35 ?        00:00:00 [rcu_sched]
root        15         2  0   23:35 ?        00:00:00 [migration/0]
root        16         2  0   23:35 ?        00:00:00 [idle_inject/0]
root        17         2  0   23:35 ?        00:00:00 [cpuhp/0]
root        18         2  0   23:35 ?        00:00:00 [cpuhp/1]
root        19         2  0   23:35 ?        00:00:00 [idle_inject/1]
root        20         2  0   23:35 ?        00:00:00 [migration/1]
root        21         2  0   23:35 ?        00:00:00 [ksoftirqd/1]
root        23         2  0   23:35 ?        00:00:00 [kworker/1:0H-kblockd]
root        24         2  0   23:35 ?        00:00:00 [kdevtmpfs]
root        25         2  0   23:35 ?        00:00:00 [inet_frag_wq]
root        26         2  0   23:35 ?        00:00:00 [kauditd]
root        27         2  0   23:35 ?        00:00:00 [khungtaskd]
root        28         2  0   23:35 ?        00:00:00 [oom_reaper]
root        29         2  0   23:35 ?        00:00:00 [writeback]
root        30         2  0   23:35 ?        00:00:00 [kcompactd0]
root        31         2  0   23:35 ?        00:00:00 [ksmd]
root        32         2  0   23:35 ?        00:00:00 [khugepaged]
root        37         2  0   23:35 ?        00:00:00 [kworker/1:1-events]
root        79         2  0   23:35 ?        00:00:00 [kintegrityd]
root        80         2  0   23:35 ?        00:00:00 [kblockd]
root        81         2  0   23:35 ?        00:00:00 [blkcg_punt_bio]
root        82         2  0   23:35 ?        00:00:00 [tpm_dev_wq]
root        83         2  0   23:35 ?        00:00:00 [ata_sff]
root        84         2  0   23:35 ?        00:00:00 [md]
root        85         2  0   23:35 ?        00:00:00 [edac-poller]
root        86         2  0   23:35 ?        00:00:00 [devfreq_wq]
root        87         2  0   23:35 ?        00:00:00 [watchdogd]
```

## 2. pgrep 실습

- 다음 명령을 실행하고 출력 결과에 대해 설명하시오.
  - pgrep 명령어가 무엇을 하는 명령어인지,
  - 각 옵션의 의미를 구체적으로 설명하기 바랍니다.

```
$ pgrep bash  
$ pgrep -l bash
```

A terminal window titled 'u2231067@ubuntu-virtualbox: ~/linux/chap06' with search, menu, and window control icons. It shows the execution of 'pgrep bash' resulting in '1711' and 'pgrep -l bash' resulting in '1711 bash'.

```
u2231067@ubuntu-virtualbox: ~/linux/chap06  
u2231067@ubuntu-virtualbox:~/linux/chap06$ pgrep bash  
1711  
u2231067@ubuntu-virtualbox:~/linux/chap06$ pgrep -l bash  
1711 bash  
u2231067@ubuntu-virtualbox:~/linux/chap06$
```

### 3. 전면처리, 후면처리 실습

- 다음 명령들을 실행하고 출력 결과에 대해 설명하시오.
  - 하단의 6개의 명령들이 각각 어떤 의미를 가지고 있는지
  - 실행 후 어떤 변화가 생겼는지 결과에 대해 구체적으로 설명하기 바랍니다.
- 출력된 프로세스들의 부모-자식 관계를 설명하시오.

```
$ sleep 100 &  
$ sleep 100 &  
$ jobs  
$ ps -f  
$ kill %1  
$ ps -f  
$ jobs
```

```
u2231067@ubuntu-virtualbox: ~/linux/chap06  
u2231067@ubuntu-virtualbox:~/linux/chap06$ sleep 100 &  
[1] 2521  
u2231067@ubuntu-virtualbox:~/linux/chap06$ sleep 100 &  
[2] 2522  
u2231067@ubuntu-virtualbox:~/linux/chap06$ jobs  
[1]- 실행 중          sleep 100 &  
[2]+ 실행 중          sleep 100 &  
u2231067@ubuntu-virtualbox:~/linux/chap06$ ps -f  
UID        PID     PPID  C STIME TTY          TIME CMD  
u2231067    1798     1788  0  9월30 pts/0    00:00:00 bash  
u2231067    2521     1798  0 00:12 pts/0    00:00:00 sleep 100  
u2231067    2522     1798  0 00:12 pts/0    00:00:00 sleep 100  
u2231067    2523     1798  0 00:12 pts/0    00:00:00 ps -f  
u2231067@ubuntu-virtualbox:~/linux/chap06$ kill %1  
u2231067@ubuntu-virtualbox:~/linux/chap06$ ps -f  
UID        PID     PPID  C STIME TTY          TIME CMD  
u2231067    1798     1788  0  9월30 pts/0    00:00:00 bash  
u2231067    2522     1798  0 00:12 pts/0    00:00:00 sleep 100  
u2231067    2524     1798  0 00:12 pts/0    00:00:00 ps -f  
[1]- 종료됨          sleep 100  
u2231067@ubuntu-virtualbox:~/linux/chap06$
```

## 4. 작업제어 실습

- 다음 명령을 실행하고 출력 결과에 대해 설명하시오.
  - 하단의 4개의 명령들이 각각 어떤 의미를 가지고 있는지
  - 실행 후 어떤 변화가 생겼는지 결과에 대해 구체적으로 설명하기 바랍니다.

```
$ (echo START1; sleep 30; echo DONE1) &  
$ fg %1  
^Z  
$ bg %1  
$ jobs
```

```
u2231067@ubuntu-virtualbox:~/linux/chap06$ (echo START1; sleep 30; echo DONE1) &  
[1] 1748  
START1  
u2231067@ubuntu-virtualbox:~/linux/chap06$ fg %1  
( echo START1; sleep 30; echo DONE1 )  
^Z  
[1]+  멈춤                ( echo START1; sleep 30; echo DONE1 )  
u2231067@ubuntu-virtualbox:~/linux/chap06$ bg %1  
[1]+ ( echo START1; sleep 30; echo DONE1 ) &  
u2231067@ubuntu-virtualbox:~/linux/chap06$ DONE1  
  
[1]+  완료                ( echo START1; sleep 30; echo DONE1 )  
u2231067@ubuntu-virtualbox:~/linux/chap06$ jobs  
u2231067@ubuntu-virtualbox:~/linux/chap06$
```



# 5. 작업제어 실습

- 다음 명령을 실행하고 출력 결과에 대해 설명하시오.
  - kill 명령어 설명의 경우, 각 옵션들의 의미에 대해 설명하기 바랍니다.

```
$ (echo START1; sleep 50; echo DONE1) &  
$ (echo START2; sleep 50; echo DONE2) &  
$ kill -KILL 첫 번째 프로세스  
$ kill -STOP 두 번째 프로세스  
$ kill -CONT 두 번째 프로세스  
$ jobs
```

```
u2231067@ubuntu-virtualbox: ~/linux/chap06  
u2231067@ubuntu-virtualbox:~/linux/chap06$ (echo START1; sleep 50; echo DONE1) &  
[1] 1796  
START1  
u2231067@ubuntu-virtualbox:~/linux/chap06$ (echo START2; sleep 50; echo DONE2) &  
[2] 1798  
START2  
u2231067@ubuntu-virtualbox:~/linux/chap06$ kill -KILL 1796  
u2231067@ubuntu-virtualbox:~/linux/chap06$  
[1]-  죽었음          ( echo START1; sleep 50; echo DONE1 )  
u2231067@ubuntu-virtualbox:~/linux/chap06$ kill -STOP 1798  
u2231067@ubuntu-virtualbox:~/linux/chap06$  
[2]+  멈춤           ( echo START2; sleep 50; echo DONE2 )  
u2231067@ubuntu-virtualbox:~/linux/chap06$ kill -CONT 1798  
u2231067@ubuntu-virtualbox:~/linux/chap06$  
u2231067@ubuntu-virtualbox:~/linux/chap06$ jobs  
[2]+  실행중         ( echo START2; sleep 50; echo DONE2 ) &  
u2231067@ubuntu-virtualbox:~/linux/chap06$  
u2231067@ubuntu-virtualbox:~/linux/chap06$ DONE2  
[2]+  완료           ( echo START2; sleep 50; echo DONE2 )  
u2231067@ubuntu-virtualbox:~/linux/chap06$ jobs  
u2231067@ubuntu-virtualbox:~/linux/chap06$
```

## 6. 프로세스 기다리기 실습

- 다음 명령을 실행하고 출력 결과에 대해 설명하시오.
  - 마지막 명령의 출력 결과가 어떤 이유로 다음과 같이 나왔는지에 대해 설명하기 바랍니다.

```
$ (echo START1; sleep 50; echo DONE1) &  
$ (echo START2; sleep 50; echo DONE2) &  
$ echo 1; wait; echo 2;
```


```
u2231067@ubuntu-virtualbox: ~/linux/chap06  
u2231067@ubuntu-virtualbox:~/linux/chap06$ (echo START1; sleep 50; echo DONE1) &  
[1] 1806  
START1  
u2231067@ubuntu-virtualbox:~/linux/chap06$ (echo START2; sleep 50; echo DONE2) &  
[2] 1808  
START2  
u2231067@ubuntu-virtualbox:~/linux/chap06$ echo 1; wait; echo 2  
1  
  
DONE1  
[1]- 완료 ( echo START1; sleep 50; echo DONE1 )  
DONE2  
[2]+ 완료 ( echo START2; sleep 50; echo DONE2 )  
2  
u2231067@ubuntu-virtualbox:~/linux/chap06$  
u2231067@ubuntu-virtualbox:~/linux/chap06$
```



# Lab 06. 과제 설명

# 과제 제출

## ■ 과제 제출 양식 (보고서)

1. 자신의 학번으로 된 계정으로 로그인하여 실습을 진행한 후 캡처 
2. 각 문제별 실행 화면 캡처 및 설명을 보고서로 작성
3. 보고서에 실습번호, 분반, 학과, 학번, 이름을 적을 것 (보고서 기본 형식 유지)
4. 워드로 보고서를 작성한 뒤 PDF로 변환하여 제출
5. 파일명: 실습번호\_분반\_학번\_이름 (Lab06\_001\_2300000\_눈송이.pdf)

## ■ 과제 제출 방법

- Snowboard Lab06에 pdf 파일 제출
- 제출 기한 : 1분반) 10월 15일 화요일 오후 11:59  
2분반) 10월 16일 수요일 오후 11:59
- 제출 기한 이후 24시간 이내 추가 제출 가능 (20% 감점)

## + ) 과제 질문 방법

- lee.jiwon@sookmyung.ac.kr (조교 메일로 질문 보내기)
- 질문 답변 시간: 월-목 10:00-17:00
- 질문 시 주의사항

- 충분히 고민 후 질문 (질문하기 전 구글링 필수!)
- 메일에 반드시 과목, 분반, 전공, 학번, 이름 명시
- 몇 번 과제에서 어떤 부분이 막혔는지, 어떤 과정이 문제인지 **명확한 설명 첨부**
- 코드 질문 시 화면 캡처가 아닌 실제 코드 파일 첨부 (출력 결과는 캡처 가능)
- 답장이 늦을 수 있으니 여유 있게 미리 질문 (특히 과제 제출 마지막 날 유의!)