

Lab 05

리눅스 시스템

조교 : 최정민

miniymay@sookmyung.ac.kr

대표문자 실습

- Lab04 실습시간에 만든 home/계정/linux 디렉터리로 이동
 - Lab04의 작업환경이 없을 경우, 아래의 명령어를 이용하여 다음과 같이 디렉터리와 파일을 생성한다.

```
$ cd ~
$ mkdir linux
$ cd linux
$ mkdir ch05
$ cd ch05
$ touch input1.txt
$ touch list1.txt
$ touch list2.txt
$ touch list3.txt
$ touch names.txt
$ touch out1.txt
$ touch out1.c
$ touch out2.c
$ touch out3.c
```

```
centos7@localhost
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[centos7@localhost ~]$ mkdir linux
[centos7@localhost ~]$ cd linux
[centos7@localhost linux]$ mkdir ch05
[centos7@localhost linux]$ cd ch05
[centos7@localhost ch05]$ touch input1.txt
[centos7@localhost ch05]$ touch list1.txt
[centos7@localhost ch05]$ touch list2.txt
[centos7@localhost ch05]$ touch list3.txt
[centos7@localhost ch05]$ touch names.txt
[centos7@localhost ch05]$ touch out1.txt
[centos7@localhost ch05]$ touch out1.c
[centos7@localhost ch05]$ touch out2.c
[centos7@localhost ch05]$ touch out3.c
[centos7@localhost ch05]$ █
```

대표문자 실습

- 다음 명령들을 이용하여 특정 디렉터리에 있는 파일들을 복사 한 후에, 그 결과를 설명하시오. 이들 사이의 차이점은 무엇인가?

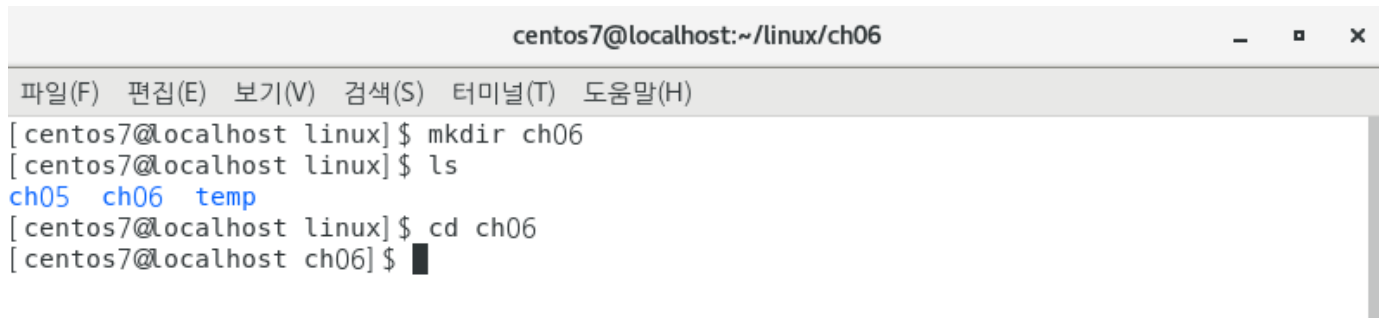
```
$ cp 디렉터리/*.c temp  
$ cp 디렉터리/?.c temp  
$ cp 디렉터리/* temp
```

```
[centos7@localhost ch05]$ cd ..  
[centos7@localhost linux]$ mkdir temp  
[centos7@localhost linux]$ ls ch05  
input1.txt  list2.txt  names.txt  out1.txt  out3.c  
list1.txt  list3.txt  out1.c     out2.c  
[centos7@localhost linux]$ cp ch05/*.c temp  
[centos7@localhost linux]$ ls temp  
out1.c  out2.c  out3.c  
[centos7@localhost linux]$ cp ch05/?.c temp  
cp: cannot stat `ch05/?.c': 그런 파일이나 디렉터리가 없습니다  
[centos7@localhost linux]$ touch ch05/a.c  
[centos7@localhost linux]$ cp ch05/?.c temp  
[centos7@localhost linux]$ ls temp  
a.c  out1.c  out2.c  out3.c  
[centos7@localhost linux]$ cp ch05/* temp  
[centos7@localhost linux]$ ls temp  
a.c          list1.txt  list3.txt  out1.c    out2.c  
input1.txt  list2.txt  names.txt  out1.txt  out3.c  
[centos7@localhost linux]$ █
```

* ch06 디렉터리 생성

- 편의를 위해 ch06 디렉터를 생성한다

```
$ mkdir ch06  
$ ls  
$ cd ch06
```

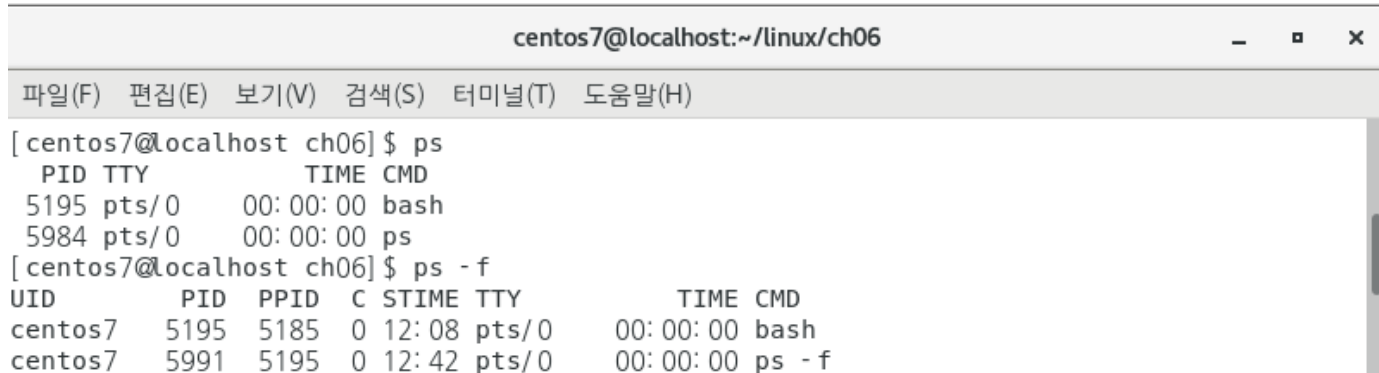


```
centos7@localhost:~/linux/ch06  
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)  
[centos7@localhost linux]$ mkdir ch06  
[centos7@localhost linux]$ ls  
ch05 ch06 temp  
[centos7@localhost linux]$ cd ch06  
[centos7@localhost ch06]$
```

1. ps 실습

- 다음 명령을 실행하고 출력 결과에 대해 설명하시오.

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



centos7@localhost:~/linux/ch06

파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)

```
[centos7@localhost ch06]$ ps  
  PID TTY          TIME CMD  
 5195 pts/0    00:00:00 bash  
 5984 pts/0    00:00:00 ps  
[centos7@localhost ch06]$ ps -f  
UID          PID    PPID  C STIME TTY          TIME CMD  
centos7      5195    5185  0 12:08 pts/0    00:00:00 bash  
centos7      5991    5195  0 12:42 pts/0    00:00:00 ps -f
```

1. ps 실습

- 다음 명령을 실행하고 출력 결과에 대해 설명하시오.

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

```
[centos7@localhost ch06]$ ps aux
```

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	1	0.1	0.5	128268	5976	?	Ss	12:05	0:02	/usr/lib/systemd/systemd
root	2	0.0	0.0	0	0	?	S	12:05	0:00	[kthreadd]
root	3	0.0	0.0	0	0	?	S	12:05	0:00	[ksoftirqd/0]
root	5	0.0	0.0	0	0	?	S<	12:05	0:00	[kworker/0:0H]
root	7	0.0	0.0	0	0	?	S	12:05	0:00	[migration/0]

```
centos7 5195 0.0 0.3 116752 3548 pts/0 Ss 12:08 0:00 bash
```

root	5593	0.0	0.0	0	0	?	S	12:16	0:00	[kworker/0:2]
root	5689	0.0	0.0	0	0	?	S	12:20	0:00	[kworker/u2:1]
root	5713	0.0	0.0	0	0	?	S	12:21	0:00	[kworker/0:0]
root	5870	0.0	0.0	0	0	?	R	12:34	0:00	[kworker/0:1]
root	5946	0.0	0.0	0	0	?	S	12:39	0:00	[kworker/0:3]
root	5976	0.0	0.0	107952	616	?	S	12:41	0:00	sleep 60
centos7	5998	0.0	0.1	155360	1900	pts/0	R+	12:42	0:00	ps aux

1. ps 실습

- 다음 명령을 실행하고 출력 결과에 대해 설명하시오.

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

```
[centos7@localhost ch06]$ ps -ef
```

UID	PID	PPID	C	STIME	TTY	TIME	CMD
root	1	0	0	12:05	?	00:00:02	/usr/lib/systemd/systemd --switched-roo
root	2	0	0	12:05	?	00:00:00	[kthreadd]
root	3	2	0	12:05	?	00:00:00	[ksoftirqd/0]
root	5	2	0	12:05	?	00:00:00	[kworker/0:0H]
root	7	2	0	12:05	?	00:00:00	[migration/0]

```
centos7 5195 5185 0 12:08 pts/0 00:00:00 bash
```

root	5689	2	0	12:20	?	00:00:00	[kworker/u2:1]
root	5713	2	0	12:21	?	00:00:00	[kworker/0:0]
root	5870	2	0	12:34	?	00:00:00	[kworker/0:1]
root	6102	2	0	12:51	?	00:00:00	[kworker/0:4]
root	6144	2	0	12:56	?	00:00:00	[kworker/0:2]
root	6160	2825	0	12:58	?	00:00:00	sleep 60
centos7	6168	5195	0	12:59	pts/0	00:00:00	ps -ef

2. pgrep 실습

- 다음 명령을 실행하고 출력 결과에 대해 설명하시오.

```
$ pgrep -l sshd  
$ pgrep -l ssh  
$ pgrep -ln sshd  
$ pgrep -ln ssh  
$ pgrep -x sshd  
$ pgrep -x ssh
```

```
[centos7@localhost ~]$ pgrep -l sshd  
3186 sshd  
[centos7@localhost ~]$ pgrep -l ssh  
3186 sshd  
4230 ssh-agent  
[centos7@localhost ~]$ pgrep -ln sshd  
3186 sshd  
[centos7@localhost ~]$ pgrep -ln ssh  
4230 ssh-agent  
[centos7@localhost ~]$ pgrep -x sshd  
3186  
[centos7@localhost ~]$ pgrep -x ssh
```


3. 전면처리, 후면처리 실습

- 다음 명령을 실행하고 출력 결과에 대해 설명하시오.
- 출력된 프로세스들의 부모-자식 관계를 설명하시오.

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

```
[centos7@localhost ch06]$ sleep 100 &
[1] 6317
[centos7@localhost ch06]$ sleep 100 &
[2] 6324
[centos7@localhost ch06]$ jobs
[1]-  Running                  sleep 100 &
[2]+  Running                  sleep 100 &
[centos7@localhost ch06]$ ps -f
UID          PID  PPID  C  STIME TTY          TIME CMD
centos7      5195  5185  0  12:08 pts/0        00:00:00 bash
centos7      6317  5195  0  13:03 pts/0        00:00:00 sleep 100
centos7      6324  5195  0  13:03 pts/0        00:00:00 sleep 100
centos7      6337  5195  0  13:03 pts/0        00:00:00 ps -f
```

프로세스 ID == PID
부모 프로세스 ID == PRID

3. 전면처리, 후면처리 실습

- 다음 명령을 실행하고 출력 결과에 대해 설명하시오.
- 출력된 프로세스들의 부모-자식 관계를 설명하시오.

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

프로세스 ID == PID
부모 프로세스 ID == PRID

```
[centos7@localhost ch06]$ kill %1
[1]- 종료됨                sleep 100
[centos7@localhost ch06]$ ps -f
```

UID	PID	PPID	C	STIME	TTY	TIME	CMD
centos7	5195	5185	0	12:08	pts/0	00:00:00	bash
centos7	6324	5195	0	13:03	pts/0	00:00:00	sleep 100
centos7	6350	5195	0	13:03	pts/0	00:00:00	ps -f

4. 작업제어 실습

- 다음 명령을 실행하고 출력 결과에 대해 설명하시오.

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

```
[centos7@localhost ch06]$ (echo START1; sleep 30; echo DONE1) &
[1] 6429
START1
[centos7@localhost ch06]$ fg %1
( echo START1; sleep 30; echo DONE1 )
^Z
[1]+  Stopped                  ( echo START1; sleep 30; echo DONE1 )
[centos7@localhost ch06]$ bg %1
[1]+ ( echo START1; sleep 30; echo DONE1 ) &
[centos7@localhost ch06]$ DONE1

[1]+  Done                    ( echo START1; sleep 30; echo DONE1 )
[centos7@localhost ch06]$ jobs
```


6. 프로세스 기다리기 실습

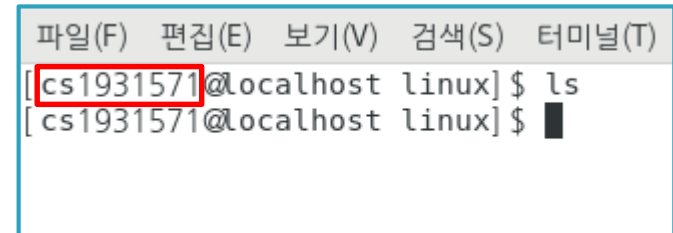
- 다음 명령을 실행하고 출력 결과에 대해 설명하시오.

```
$ (echo START1; sleep 30; echo DONE1) &  
$ (echo START2; sleep 30; echo DONE2) &  
$ echo 1; wait; echo 2
```

```
[centos7@localhost ch06]$ (echo START1; sleep 30; echo DONE1) &  
[1] 6767  
START1  
[centos7@localhost ch06]$ (echo START2; sleep 30; echo DONE2) &  
[2] 6775  
START2  
[centos7@localhost ch06]$ echo 1; wait; echo 2  
1  
DONE1  
[1]- Done ( echo START1; sleep 30; echo DONE1 )  
DONE2  
[2]+ Done ( echo START2; sleep 30; echo DONE2 )  
2  
[centos7@localhost ch06]$ █
```

과제 제출 양식

- 실습 자료에 나와있는 각 내용 별 실행 화면 캡처 및 간략한 설명을 보고서로 작성
- 캡처화면에 사용자(학번) 이름이 보이도록 캡처
 - root 권한이 필요한 경우는, root 그대로 진행
- **양식** : 스노우보드에서 다운로드
- 최종 보고서 스노우보드에 업로드



```
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T)
[cs1931571@localhost linux]$ ls
[cs1931571@localhost linux]$
```

- 파일 명 : Lab05_학번_이름.docx
- 제출기한 : 10월 15일 오후 11:55까지

궁금한 점 + 질문

- 조교 : 최정민
- 이메일 : miniymay@sookmyung.ac.kr

[질문 시 주의!]

- ✓ 꼭 과목, 분반, 전공, 이름, 학번을 알려주세요!
- ✓ 그냥 코드만 보내면 (X)
- ✓ 몇 번 과제에서, 어떤 부분이 막혔는지 or 어떤 과정인지 설명과 함께
- ✓ 그 외에 출석, 과제 제출 확인 혹은 과목과 관련 없는 진로 등등 질문도 환영!
- ✓ 실습 마지막 제출 날 당일에는 답장이 늦을 수 있다는 점 고려해주세요 ☺
- ✓ Office Hour
: 메일로 시간 약속 잡기 – 연구실 : 명신관 316C호