KWANGWOON UNIVERSITY

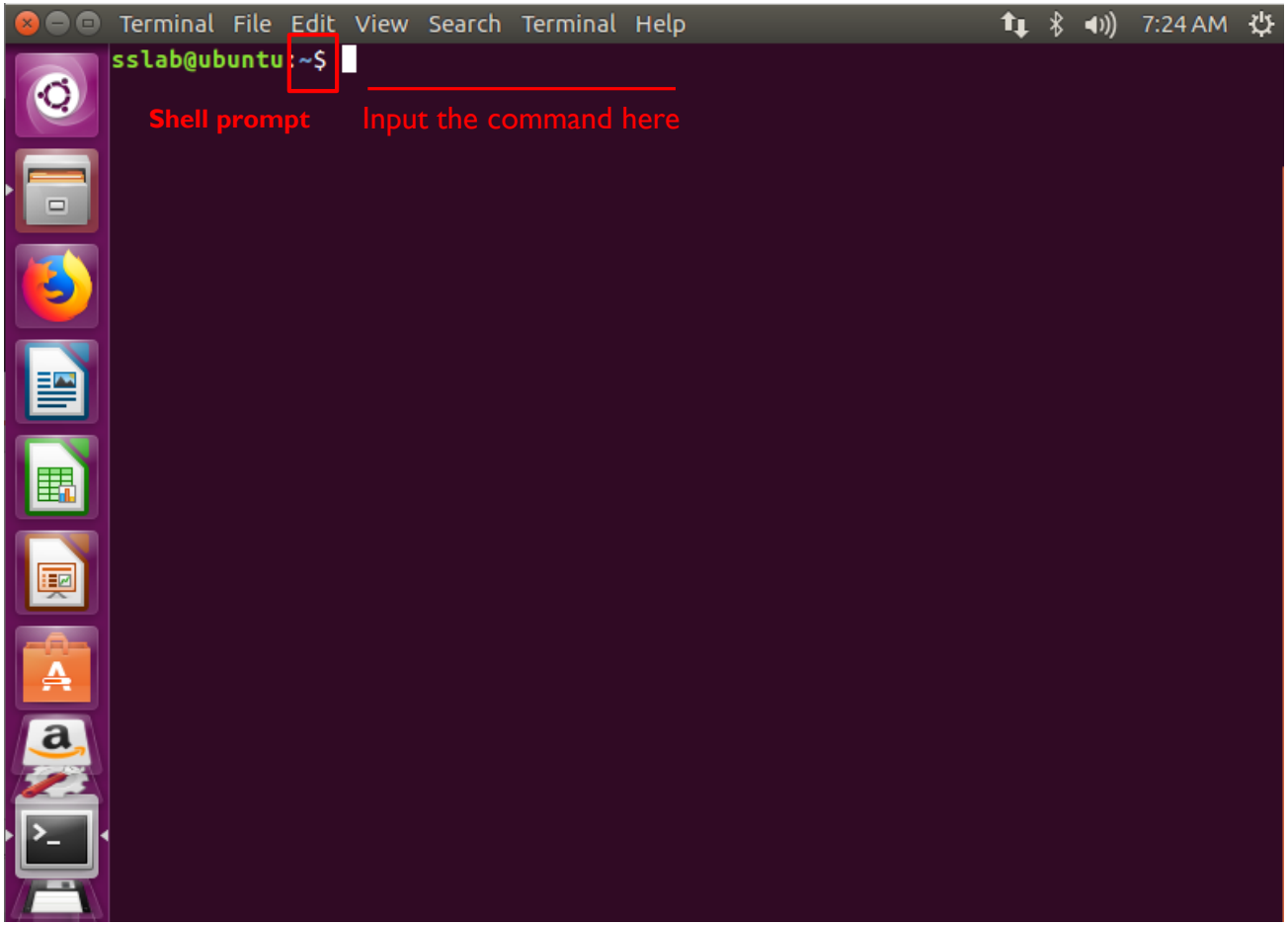
2019년 1학기 **시스템프로그래밍실습** 2주차

Unix/Linux Commands

System Software Laboratory
College of Software and Convergence
Kwangwoon Univ.

Linux Terminal

- **Command Line Interface**
  - To get access to the shell.



man (1/4)

- **format and display the on-line manual pages**
  - Usage: man [option] name …
  - E.g.
    - # man ls
    - # man –k copy //keyword search
    - # man –a write //all manuals

man (2/4)

- **# man kill**

```
KILL(1)                                  User Commands                                  KILL(1)

NAME
       kill - send a signal to a process

SYNOPSIS
       kill [options] <pid> [...]

DESCRIPTION
       The  default signal for kill is TERM.  Use -l or -L to list available signals.  Particularly useful signals include HUP, INT, KILL,
       STOP, CONT, and 0.  Alternate signals may be specified in three ways: -9, -SIGKILL or -KILL.  Negative PID values may  be  used  to
       choose  whole  process groups; see the PGID column in ps command output.  A PID of -1 is special; it indicates all processes except
       the kill process itself and init.

OPTIONS
       <pid> [...]
              Send signal to every <pid> listed.

       -<signal>
       -s <signal>
       --signal <signal>
              Specify the signal to be sent.  The signal can be specified by using name or number.  The behavior of signals  is  explained
              in signal(7) manual page.

       -l, --list [signal]
              List signal names.  This option has optional argument, which will convert signal number to signal name, or other way round.

       -L, --table
              List signal names in a nice table.

       NOTES  Your  shell  (command line interpreter) may have a built-in kill command.  You may need to run the command described here as
              /bin/kill to solve the conflict.

EXAMPLES
       kill -9 -1
              Kill all processes you can kill.

       kill -l 11
              Translate number 11 into a signal name.

       kill -L
              List the available signal choices in a nice table.

       kill 123 543 2341 3453
              Send the default signal, SIGTERM, to all those processes.

SEE ALSO
       kill(2), killall(1), nice(1), pkill(1), renice(1), signal(7), skill(1)

STANDARDS
       This command meets appropriate standards. The -L flag is Linux-specific.

AUTHOR
       Albert Cahalan (albert@users.sf.net) wrote kill in 1999 to replace a bsdutils one that was not standards compliant.  The util-linux
```

man (3/4)

- **Section description**
  - (1) General commands
  - (2) System calls
  - (3) C library functions
  - (4) Special files (usually devices) and drivers
  - (5) File formats and conventions
  - (6) Games and screensavers
  - (7) Miscellanea
  - (8) System administration commands and daemons

- **Examples**
  - ls(1), open(2), fopen(3)
  - write(1)/write(2)

man (4/4)

- **Manual layout**
  - NAME
    - name of the command or function
  - SYNOPSIS
    - command: how to run,
    - functions: parameter list
  - DESCRIPTION
    - description of the functioning of the command or function.
  - EXAMPLES
    - some examples of common usage.
  - SEE ALSO
    - list of related commands or functions.
  - OPTIONS, EXIT STATUS, ENVIRONMENT, KNOWN BUGS, FILES, AUTHOR, REPORTING BUGS, HISTORY and COPYRIGHT.

ls

- **List directory contents**
  - Usage: ls [OPTION]... [FILE]...
  - Useful options
    - -a: hidden file을 포함한 모든 파일을 출력
    - -F: 파일을 종류 표시 (/는 디렉토리, *는 실행파일)
    - -l: 파일의 정보를 자세하게 출력

```
sslab@ubuntu:~$ ls
Desktop    Downloads         Music     Practice  Public     Test    work
Documents  examples.desktop  Pictures  projects  Templates  Videos
sslab@ubuntu:~$ ls -a
.              .dmrc             Music                        Test
..             Documents         Pictures                     Videos
.bash_history  Downloads         Practice                     .vim
.bash_logout   examples.desktop  .profile                     .viminfo
.bashrc        .gconf            projects                     work
.cache         .gnupg            Public                       .Xauthority
.config        .ICEauthority     .sudo_as_admin_successful    .xsession-errors
Desktop        .local            Templates                    .xsession-errors.old
sslab@ubuntu:~$ ls -F
Desktop/    Downloads/        Music/     Practice/  Public/     Test*    work/
Documents/  examples.desktop  Pictures/  projects/  Templates/  Videos/
sslab@ubuntu:~$
```

pwd

- **print name of current/working directory**
  - Usage: pwd [OPTION]
  - E.g.

```
sslab@ubuntu:~$ pwd
/home/sslab
sslab@ubuntu:~$
```

cd

- **change the current directory**
  - usage : cd [-L|-P] [dir]
  - Special filenames
    - . (current directory)
    - .. (parent directory)
  - E.g.

```
sslab@ubuntu:~$ pwd
/home/sslab
sslab@ubuntu:~$ ls
Desktop    Downloads         Music     Practice  Public     Test    work
Documents  examples.desktop  Pictures  projects  Templates  Videos
sslab@ubuntu:~$ cd work
sslab@ubuntu:~/work$ pwd
/home/sslab/work
sslab@ubuntu:~/work$ cd .
sslab@ubuntu:~/work$ cd ..
sslab@ubuntu:~$ cd work
sslab@ubuntu:~/work$ cd ~
sslab@ubuntu:~$ cd -
/home/sslab/work
sslab@ubuntu:~/work$
```

$ cd ~ // is equivalent to 'cd'
$ cd - // is equivalent to $OLDPWD

cat

- **concatenate files and print on the standard output**
  - Usage: cat [OPTION] [FILE]...
  - E.g.

```
sslab@ubuntu:~/work$ ls
fileA.txt  fileB.txt
sslab@ubuntu:~/work$ cat fileA.txt
Hello This is FileA
sslab@ubuntu:~/work$ cat fileB.txt
Hello This is FileB
sslab@ubuntu:~/work$ cat fileA.txt fileB.txt
Hello This is FileA
Hello This is FileB
sslab@ubuntu:~/work$
```

File permission

- **파일 허가 지정**
  - owner, group, others 세 부류에 대해
  - read
    - file: 파일 내용 열람/복사 가능, 수정/삭제 불가능
    - directory: 디렉토리 내의 파일이름 열람 가능(ls)
  - write
    - file: 파일 내용 수정/삭제 가능, 열람/복사 불가능
    - directory: 파일을 생성하거나 삭제할 수 있는 권리
  - execute
    - file: 실행시킬 권리의 유무
    - directory: 이동 가능 여부(cd)

chmod

- **change file access permissions**
  - Usage

    % chmod [OPTION]... MODE[,MODE]... FILE...

    % chmod [OPTION]... OCTAL-MODE FILE...
  - E.g.

```
sslab@ubuntu:~/work$ ls -al
total 12
drwxrwxr-x  2 sslab sslab 4096 Mar  9 00:29 .
drwxr-xr-x 19 sslab sslab 4096 Mar  9 00:29 ..
-rw-rw-r--  1 sslab sslab   16 Mar  9 00:29 hello.txt
sslab@ubuntu:~/work$ chmod u-w,g-w,o-r hello.txt
sslab@ubuntu:~/work$ ls -l
total 4
-r--r----- 1 sslab sslab 16 Mar  9 00:29 hello.txt
sslab@ubuntu:~/work$ chmod 644 hello.txt
sslab@ubuntu:~/work$ ls -l
total 4
-rw-r--r-- 1 sslab sslab 16 Mar  9 00:29 hello.txt
sslab@ubuntu:~/work$
```

mkdir

- **make directories**
  - Usage: mkdir [OPTION] DIRECTORY...
  - E.g.

```
sslab@ubuntu:~/work$ ls -l
total 4
-rw-r--r-- 1 sslab sslab 16 Mar  9 00:29 hello.txt
sslab@ubuntu:~/work$ mkdir SP_lecture
sslab@ubuntu:~/work$ ls -l
total 8
-rw-r--r-- 1 sslab sslab   16 Mar  9 00:29 hello.txt
drwxrwxr-x 2 sslab sslab 4096 Mar  9 00:31 SP_lecture
sslab@ubuntu:~/work$
```

rmdir

- **remove empty directories**
  - Usage: rmdir [OPTION]... DIRECTORY...
  - E.g.

```
sslab@ubuntu:~/work$ ls -l
total 8
-rw-r--r-- 1 sslab sslab   16 Mar  9 00:29 hello.txt
drwxrwxr-x 2 sslab sslab 4096 Mar  9 00:34 SP_lecture
sslab@ubuntu:~/work$ rmdir SP_lecture
sslab@ubuntu:~/work$ ls -l
total 4
-rw-r--r-- 1 sslab sslab 16 Mar  9 00:29 hello.txt
sslab@ubuntu:~/work$
```

rm (1/3)

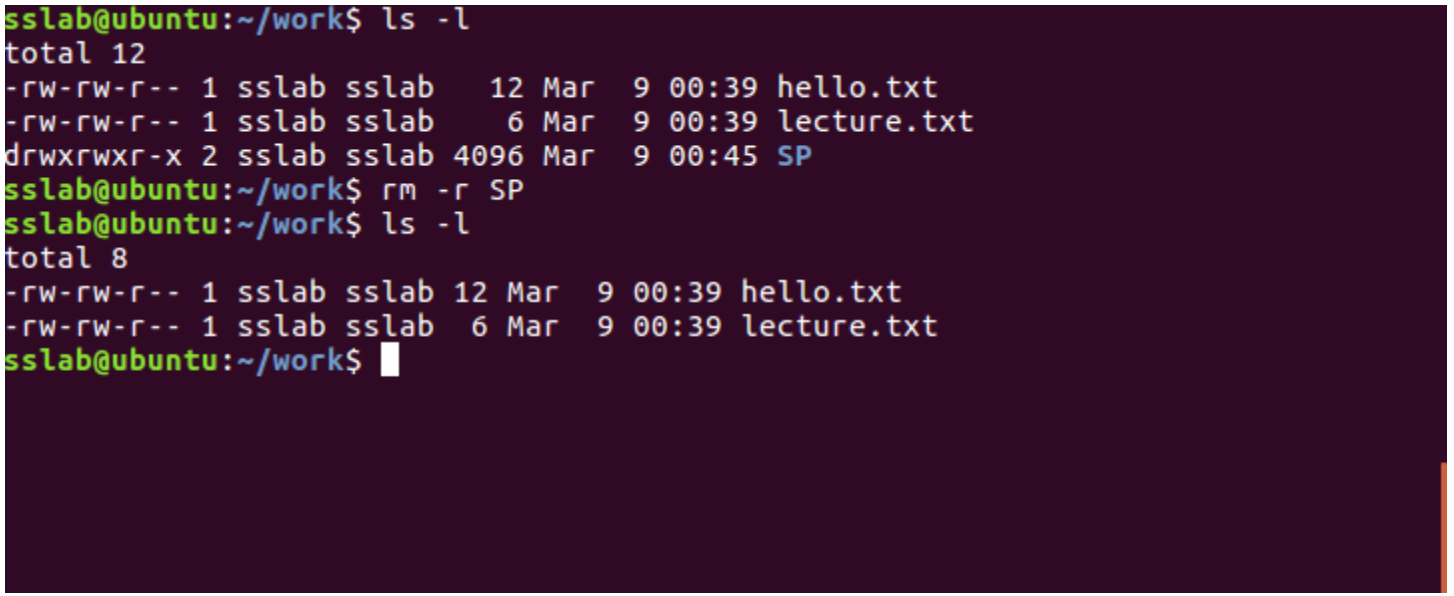
- **remove files or directories**
  - Usage: rm [OPTION]... FILE...
  - E.g.

```
sslab@ubuntu:~/work$ ls -l
total 12
-rw-rw-r-- 1 sslab sslab  4 Mar  9 00:39 enjoy.txt
-rw-rw-r-- 1 sslab sslab 12 Mar  9 00:39 hello.txt
-rw-rw-r-- 1 sslab sslab  6 Mar  9 00:39 lecture.txt
sslab@ubuntu:~/work$ rm enjoy.txt
sslab@ubuntu:~/work$ ls -l
total 8
-rw-rw-r-- 1 sslab sslab 12 Mar  9 00:39 hello.txt
-rw-rw-r-- 1 sslab sslab  6 Mar  9 00:39 lecture.txt
sslab@ubuntu:~/work$
```

rm (2/3)

- E.g.

-r remove the contents of directory recursively



rm (3/3)

- E.g.

-i prompt before every removal

```
sslab@ubuntu:~/work$ ls -l
total 12
-rw-rw-r-- 1 sslab sslab    6 Mar  9 00:50 file.txt
-rw-rw-r-- 1 sslab sslab    5 Mar  9 00:49 hello.txt
drwxrwxr-x 2 sslab sslab 4096 Mar  9 00:49 html
sslab@ubuntu:~/work$ rm -i *
rm: remove regular file 'file.txt'? y
rm: remove regular file 'hello.txt'? n
rm: cannot remove 'html': Is a directory
sslab@ubuntu:~/work$ ls -l
total 8
-rw-rw-r-- 1 sslab sslab    5 Mar  9 00:49 hello.txt
drwxrwxr-x 2 sslab sslab 4096 Mar  9 00:49 html
sslab@ubuntu:~/work$
```

cp

- **copy files and directories**
  - Usage: cp [OPTION]... SOURCE DEST
    cp [OPTION]... SOURCE... DIRECTORY
  - E.g.

```
sslab@ubuntu:~/work$ ls -l
total 8
-rw-rw-r-- 1 sslab sslab    5 Mar  9 00:49 hello.txt
drwxrwxr-x 2 sslab sslab 4096 Mar  9 00:58 html
sslab@ubuntu:~/work$ cp hello.txt hello_copy.txt
sslab@ubuntu:~/work$ ls -l
total 12
-rw-rw-r-- 1 sslab sslab    5 Mar  9 01:00 hello_copy.txt
-rw-rw-r-- 1 sslab sslab    5 Mar  9 00:49 hello.txt
drwxrwxr-x 2 sslab sslab 4096 Mar  9 00:58 html
sslab@ubuntu:~/work$ cp html/* .
sslab@ubuntu:~/work$ ls -l
total 24
-rw-rw-r-- 1 sslab sslab    4 Mar  9 01:00 file1.txt
-rw-rw-r-- 1 sslab sslab    5 Mar  9 01:00 file2.txt
-rw-rw-r-- 1 sslab sslab    4 Mar  9 01:00 file3.txt
-rw-rw-r-- 1 sslab sslab    5 Mar  9 01:00 hello_copy.txt
-rw-rw-r-- 1 sslab sslab    5 Mar  9 00:49 hello.txt
drwxrwxr-x 2 sslab sslab 4096 Mar  9 00:58 html
sslab@ubuntu:~/work$
```

mv

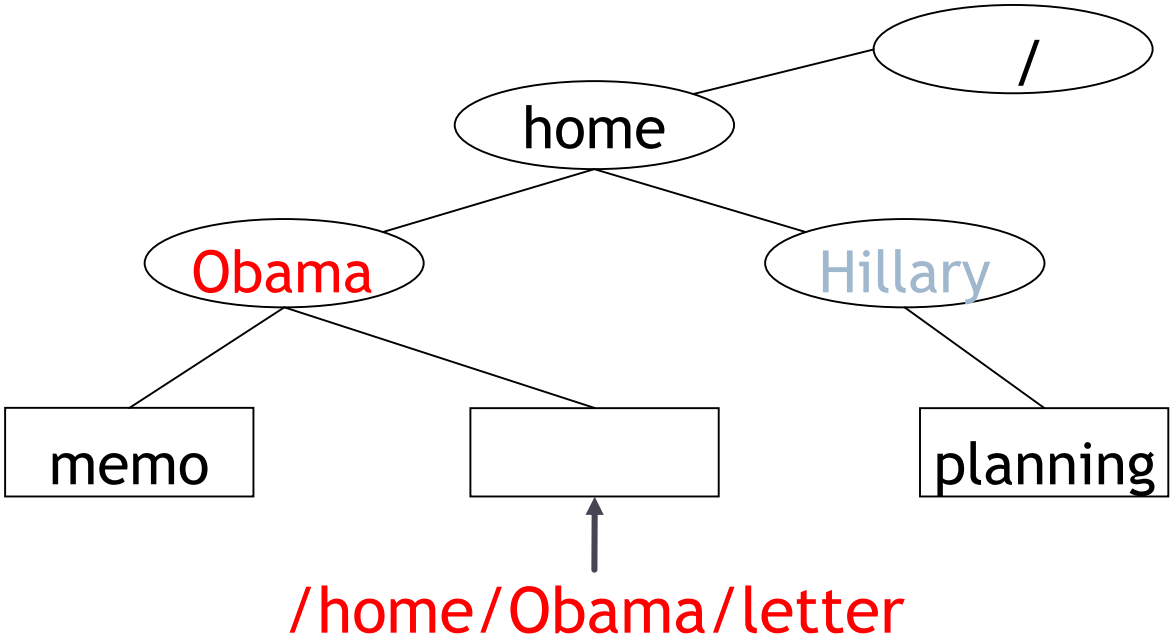
- **move (rename) files**
  - Usage: mv [OPTION]... SOURCE DEST
  - E.g

```
sslab@ubuntu:~/work$ ls
empty.txt  file2.txt  file.txt  text.txt
ex         file3.txt  html      world.txt
sslab@ubuntu:~/work$ mv world.txt /home/sslab/work/ex
sslab@ubuntu:~/work$ ls
empty.txt  ex  file2.txt  file3.txt  file.txt  html  text.txt
sslab@ubuntu:~/work$ cd ex
sslab@ubuntu:~/work/ex$ ls
file  world.txt
sslab@ubuntu:~/work/ex$ cd ..
sslab@ubuntu:~/work$ mv ex LINUX
sslab@ubuntu:~/work$ ls
empty.txt  file2.txt  file3.txt  file.txt  html  LINUX  text.txt
sslab@ubuntu:~/work$
```

ln (1/3)

- **make links between files**
  - Usage: ln [OPTION]... TARGET [LINK_NAME]
  - E.g. (working directory is /home/Hillary)

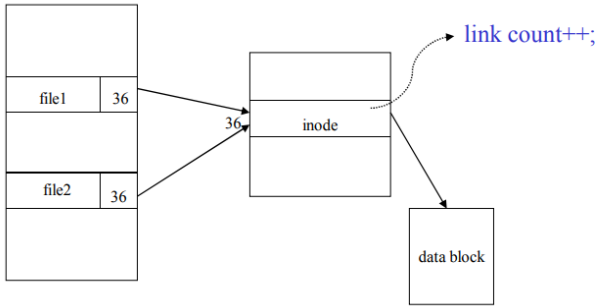
    % ln /home/Obama/letter draft

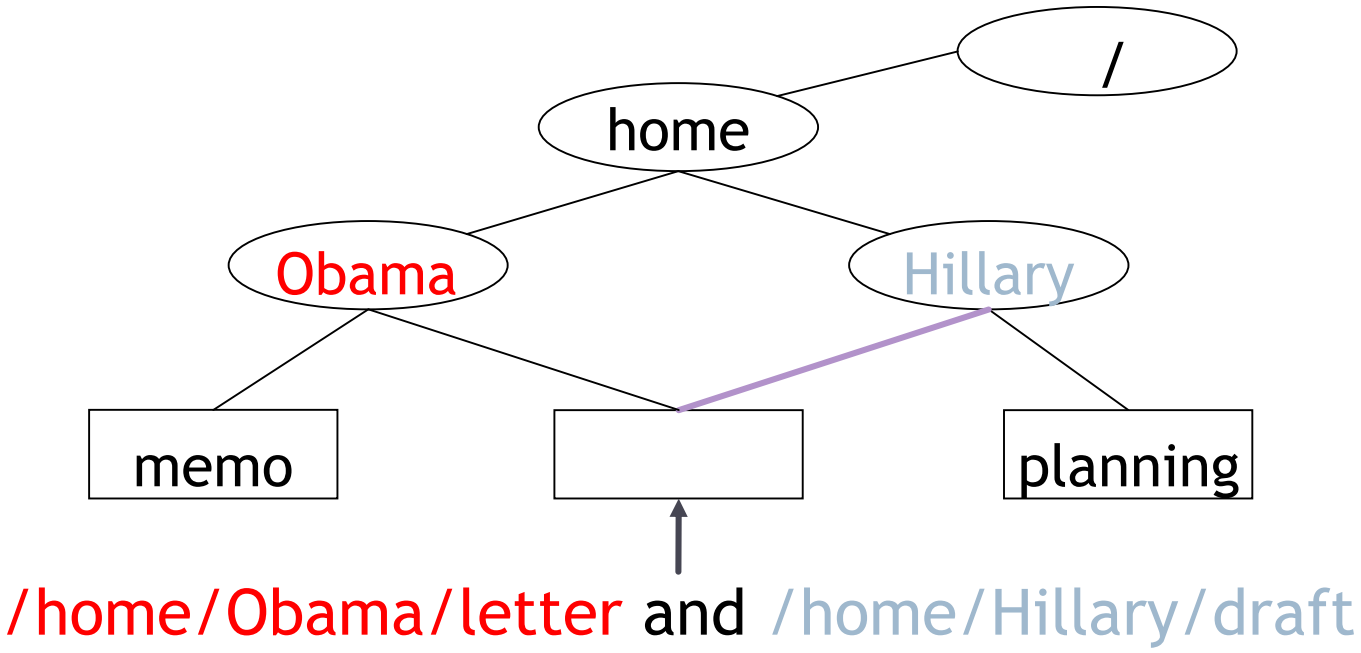


ln (1/3)

- **make links between files**
  - Usage: ln [OPTION]... TARGET [LINK_NAME]
  - E.g. (working directory is /home/Hillary)

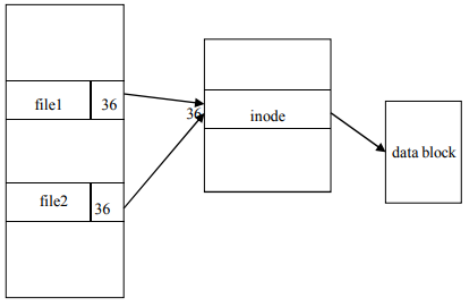
    % ln /home/Obama/letter draft



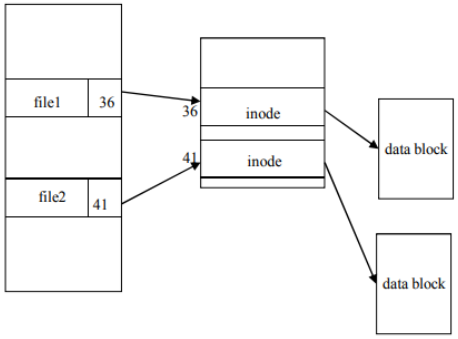


ln (2/3)

- **ln vs. cp**



$ ln file1 file2



$ cp file1 file2

```
sslab@ubuntu:~/work/LINUX$ cat file_a
This is file A.
sslab@ubuntu:~/work/LINUX$ ln file_a file_b
sslab@ubuntu:~/work/LINUX$ cat file_b
This is file A.
sslab@ubuntu:~/work/LINUX$ vi file_b
sslab@ubuntu:~/work/LINUX$ cat file_b
This is file B after the change.
sslab@ubuntu:~/work/LINUX$ cat file_a
This is file B after the change.
sslab@ubuntu:~/work/LINUX$
```

```
sslab@ubuntu:~/work/LINUX$ cat file_c
This is file C.
sslab@ubuntu:~/work/LINUX$ cp file_c file_d
sslab@ubuntu:~/work/LINUX$ cat file_d
This is file C.
sslab@ubuntu:~/work/LINUX$ vi file_d
sslab@ubuntu:~/work/LINUX$ cat file_c
This is file C.
sslab@ubuntu:~/work/LINUX$ cat file_d
This is file D after the change.
sslab@ubuntu:~/work/LINUX$
```

ln (3/3)

- **Symbolic link**
  - special file that contains another file's path
  - E.g.

```
sslab@ubuntu:~/work$ cat file_a.txt
hello everyone! I am file_a
sslab@ubuntu:~/work$ ln -s file_a.txt file_b.txt
sslab@ubuntu:~/work$ ls -l
total 8
-rw-rw-r-- 1 sslab sslab   28 Mar  9 01:15 file_a.txt
lrwxrwxrwx 1 sslab sslab   10 Mar  9 01:16 file_b.txt -> file_a.txt
drwxrwxr-x 2 sslab sslab 4096 Mar  9 01:13 html
sslab@ubuntu:~/work$ cat file_b.txt
hello everyone! I am file_a
sslab@ubuntu:~/work$ rm file_a.txt
sslab@ubuntu:~/work$ cat file_b.txt
cat: file_b.txt: No such file or directory
sslab@ubuntu:~/work$
```

Symbolic link의 필요성
- 디렉토리에 대한 hard link는 superuser만 만들 수 있다.
- 다른 파일시스템 사이에서는 hard link를 사용할 수 없다.

touch

- **Make an empty file or change filestamps**
  - Usage : touch [OPTION]… FILE…
  - E.g.

```
sslab@ubuntu:~/work$ ls
file1.txt  file2.txt  file3.txt  html
sslab@ubuntu:~/work$ touch empty.txt
sslab@ubuntu:~/work$ ls
empty.txt  file1.txt  file2.txt  file3.txt  html
sslab@ubuntu:~/work$ ls -l
total 16
-rw-rw-r-- 1 sslab sslab    0 Mar  9 01:23 empty.txt
-rw-rw-r-- 1 sslab sslab   11 Mar  9 01:21 file1.txt
-rw-rw-r-- 1 sslab sslab    7 Mar  9 01:21 file2.txt
-rw-rw-r-- 1 sslab sslab    8 Mar  9 01:21 file3.txt
drwxrwxr-x 2 sslab sslab 4096 Mar  9 01:13 html
sslab@ubuntu:~/work$ touch empty.txt
sslab@ubuntu:~/work$ ls -l
total 16
-rw-rw-r-- 1 sslab sslab    0 Mar  9 01:25 empty.txt
-rw-rw-r-- 1 sslab sslab   11 Mar  9 01:21 file1.txt
-rw-rw-r-- 1 sslab sslab    7 Mar  9 01:21 file2.txt
-rw-rw-r-- 1 sslab sslab    8 Mar  9 01:21 file3.txt
drwxrwxr-x 2 sslab sslab 4096 Mar  9 01:13 html
sslab@ubuntu:~/work$
```

ps

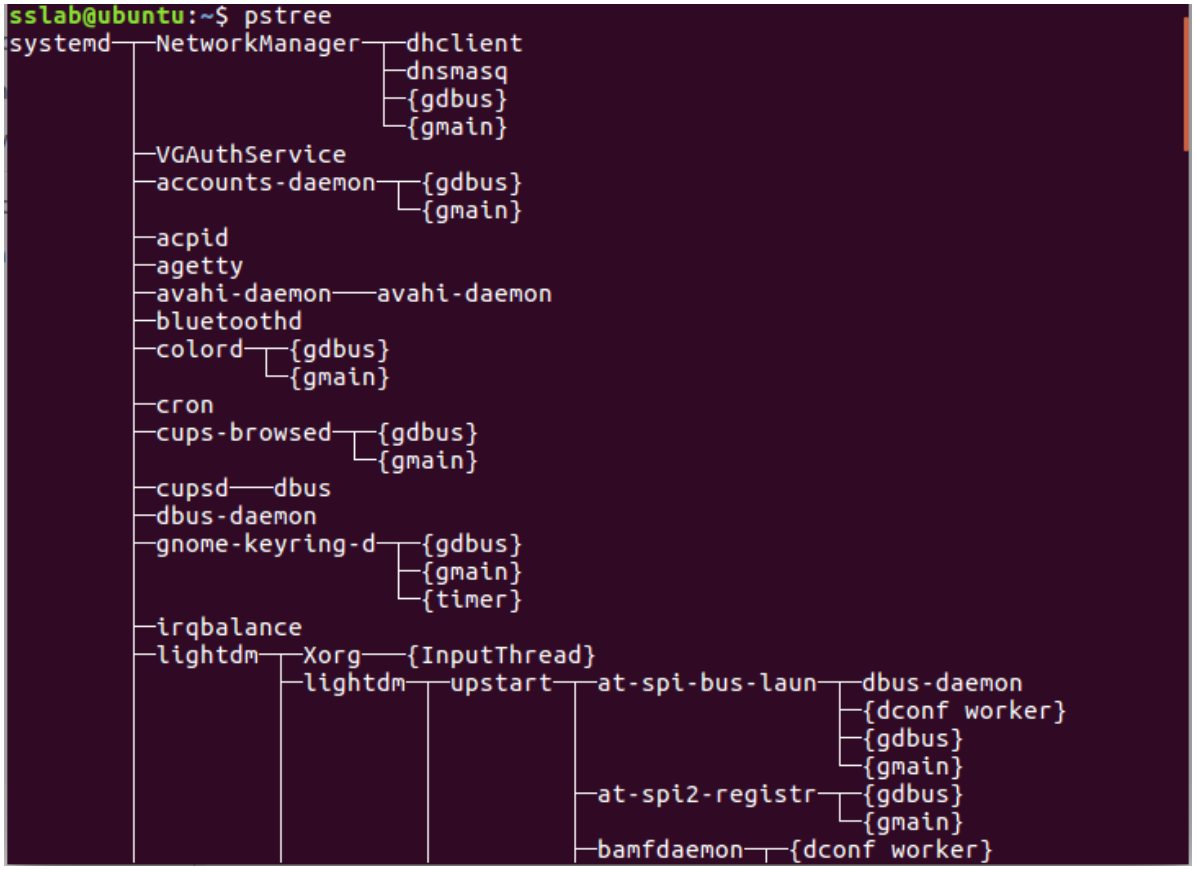
- **report process status**
  - Usage: ps [options]
  - E.g.

```
sslab@ubuntu:~$ ps
   PID TTY          TIME CMD
  2240 pts/12   00:00:00 bash
  3500 pts/12   00:00:00 ps
sslab@ubuntu:~$ ps -ef
UID          PID    PPID  C STIME TTY          TIME CMD
root           1       0  0 Mar08 ?        00:00:01 /sbin/init auto noprompt
root           2       0  0 Mar08 ?        00:00:00 [kthreadd]
root           4       2  0 Mar08 ?        00:00:00 [kworker/0:0H]
root           6       2  0 Mar08 ?        00:00:00 [mm_percpu_wq]
root           7       2  0 Mar08 ?        00:00:00 [ksoftirqd/0]
root           8       2  0 Mar08 ?        00:00:00 [rcu_sched]
root           9       2  0 Mar08 ?        00:00:00 [rcu_bh]
root          10       2  0 Mar08 ?        00:00:00 [migration/0]
root          11       2  0 Mar08 ?        00:00:00 [watchdog/0]
root          12       2  0 Mar08 ?        00:00:00 [cpuhp/0]
root          13       2  0 Mar08 ?        00:00:00 [cpuhp/1]
root          14       2  0 Mar08 ?        00:00:00 [watchdog/1]
root          15       2  0 Mar08 ?        00:00:00 [migration/1]
root          16       2  0 Mar08 ?        00:00:00 [ksoftirqd/1]
root          18       2  0 Mar08 ?        00:00:00 [kworker/1:0H]
root          19       2  0 Mar08 ?        00:00:00 [kdevtmpfs]
root          20       2  0 Mar08 ?        00:00:00 [netns]
```

-e : select all processes
-f : full format listing

pstree

- **Display a tree of processes**
  - Usage: pstree [options]
  - E.g.



exit

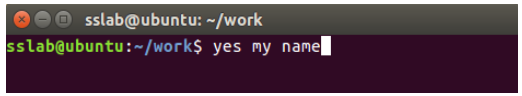
- **Cause the shell to exit**
  - Usage: exit
  - E.g.

```
sslab@ubuntu:~$ sudo apt-get install csh
[sudo] password for sslab:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  csh
0 upgraded, 1 newly installed, 0 to remove and 94 not upgraded.
```

```
sslab@ubuntu:~$ ps
   PID TTY          TIME CMD
  2240 pts/12   00:00:00 bash
  3979 pts/12   00:00:00 ps
sslab@ubuntu:~$ csh
% ps
   PID TTY          TIME CMD
  2240 pts/12   00:00:00 bash
  3980 pts/12   00:00:00 csh
  3981 pts/12   00:00:00 ps
% exit
% exit
sslab@ubuntu:~$ ps
   PID TTY          TIME CMD
  2240 pts/12   00:00:00 bash
  3982 pts/12   00:00:00 ps
sslab@ubuntu:~$
```

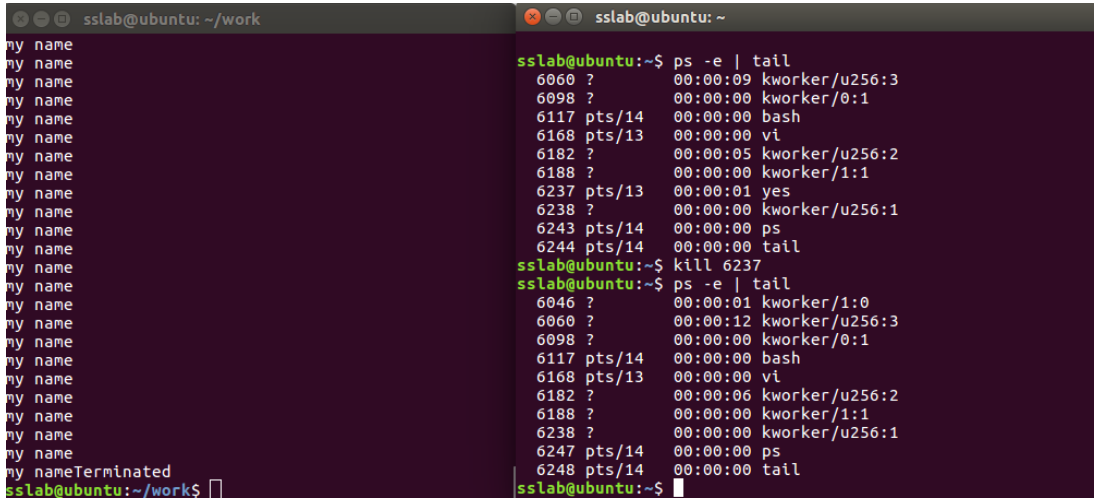
kill

- **send a signal to a process**
  - Usage: kill [-s signal | -p] [-a] [--] pid ...
  - The default signal for kill is TERM. (i.e. Terminate process)
  - E.g.



| : 이전 명령어의 output을 다음 명령어의 Input으로 연결
tail : 파일의 끝 부분부터 10개의 행 출력

string 출력
무한 반복



다른 terminal 상의 process를 종료

kill

- **KILL signal (-9)**
  - -9 : SIGKILL (process 강제 종료)
  - E.g

Ctrl + Z

```
sslab@ubuntu:~$ ps
   PID TTY          TIME CMD
  6117 pts/14   00:00:00 bash
  6280 pts/14   00:00:00 ps
sslab@ubuntu:~$ vi hello

[1]+  Stopped                 vi hello
sslab@ubuntu:~$ ps
   PID TTY          TIME CMD
  6117 pts/14   00:00:00 bash
  6283 pts/14   00:00:00 vi
  6288 pts/14   00:00:00 ps
sslab@ubuntu:~$ kill -9 6283
sslab@ubuntu:~$ ps
   PID TTY          TIME CMD
  6117 pts/14   00:00:00 bash
  6289 pts/14   00:00:00 ps
[1]+  Killed                  vi hello
sslab@ubuntu:~$
```

time

- **Time a simple command or give resource usage**
  - Usage: time [options] command [arguments…]
  - E.g

```
sslab@ubuntu:~$ time ps
   PID TTY          TIME CMD
  5645 pts/13   00:00:00 bash
  5875 pts/13   00:00:00 ps

real    0m0.008s
user    0m0.000s
sys     0m0.007s
sslab@ubuntu:~$
```

real : 실제 CPU 소요 시간
user : user영역에서 소비된 CPU 시간
sys : 커널에서 소비된 CPU 시간

passwd

- **Update a user's authentication tokens**
  - Usage: passwd [options]
  - E.g

```
sslab@ubuntu:~$ passwd
Changing password for sslab.
(current) UNIX password:
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
sslab@ubuntu:~$
```

uname

- **Display system information**
  - Usage: uname [options]
  - E.g

```
sslab@ubuntu:~$ uname
Linux
sslab@ubuntu:~$ uname -r
4.15.0-46-generic
sslab@ubuntu:~$ uname -m
x86_64
sslab@ubuntu:~$ uname -a
Linux ubuntu 4.15.0-46-generic #49~16.04.1-Ubuntu SMP Tue Feb 12 17:45:24 UTC 2019 x86_64
x86_64 x86_64 GNU/Linux
```

-r : print the kernel release
-m : print the machine hardware name
-a : print all information

wc

- **Print newline, word, and byte count for each file**
  - Usage: wc [options]… [FILE]…
  - E.g

```
sslab@ubuntu:~/work$ cat file.txt
2019 system programming course
command test
sslab@ubuntu:~/work$ wc file.txt
 2  6 46 file.txt
sslab@ubuntu:~/work$ wc -c file.txt
46 file.txt
sslab@ubuntu:~/work$ wc -w file.txt
6 file.txt
sslab@ubuntu:~/work$ wc -l file.txt
2 file.txt
sslab@ubuntu:~/work$
```

more

- **more is a filter for paging through text one screenful at a time**
  - Usage: more [-options] [-num] [+/ pattern] [+ linenum] [file …]
  - E.g

```
sslab@ubuntu:~$ more file.txt
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
this is file !
--More--(56%)
```

echo

- **Display a line of text**
  - Usage: echo [OPTION]… [STRING]…
  - Display environment variable
  - E.g

```
sslab@ubuntu:~$ echo helloworld
helloworld
sslab@ubuntu:~$ echo $HOME
/home/sslab
sslab@ubuntu:~$ echo ~
/home/sslab
sslab@ubuntu:~$
```

alias

- **alias is a command which enables a replacement of a word by another string.**
  - E.g. alias myls='ls –al'

```
sslab@ubuntu:~/work$ myls
No command 'myls' found, did you mean:
 Command 'tyls' from package 'terminology' (universe)
 Command 'mmls' from package 'sleuthkit' (universe)
myls: command not found
sslab@ubuntu:~/work$ alias myls='ls -al'
sslab@ubuntu:~/work$ myls
total 28
drwxrwxr-x  3 sslab sslab 4096 Mar  9 05:21 .
drwxr-xr-x 19 sslab sslab 4096 Mar  9 05:21 ..
-rw-rw-r--  1 sslab sslab    0 Mar  9 01:25 empty.txt
-rw-rw-r--  1 sslab sslab    7 Mar  9 01:21 file2.txt
-rw-rw-r--  1 sslab sslab    8 Mar  9 01:21 file3.txt
-rw-rw-r--  1 sslab sslab   46 Mar  9 04:52 file.txt
drwxrwxr-x  2 sslab sslab 4096 Mar  9 01:13 html
-rw-rw-r--  1 sslab sslab   43 Mar  9 05:21 text.txt
sslab@ubuntu:~/work$ alias
alias alert='notify-send --urgency=low -i "$([ $? = 0 ] && echo terminal || echo er
ror)" "$(history|tail -n1|sed -e '\''s/^\s*[0-9]\+\s*//;s/[;&|]\s*alert$//'\'')"'
alias egrep='egrep --color=auto'
alias fgrep='fgrep --color=auto'
alias grep='grep --color=auto'
alias l='ls -CF'
alias la='ls -A'
alias ll='ls -alF'
alias ls='ls --color=auto'
alias myls='ls -al'
sslab@ubuntu:~/work$
```

find

- **Search for files in a directory hierarchy**
  - Find [-H] [-L] [-P] [path…] [expression]
  - E.g

```
sslab@ubuntu:~/work$ cd html/
sslab@ubuntu:~/work/html$ ls
file1.txt  file2.txt  file3.txt  hello_copy.txt
sslab@ubuntu:~/work/html$ cd ..
sslab@ubuntu:~/work$ ls
empty.txt  file2.txt  file3.txt  html
sslab@ubuntu:~/work$ find -name '*.txt'
./file3.txt
./html/file3.txt
./html/file1.txt
./html/file2.txt
./html/hello_copy.txt
./empty.txt
./file2.txt
sslab@ubuntu:~/work$
```

grep

- **Searches the named input FILEs(or standard input if no files are named, or the file name is given) for lines containing a match to the given PATTERN**
  - grep [options] [PATTEN] [FILE…]
  - E.g

Unix commands

File and file system management	cat · cd · chmod · chown · chgrp · cmp · cp · du · df · file · fsck · ln · ls · mkdir · mount · mv · pwd · rm · rmdir · touch
Process management	exit · kill · killall · nice · ps · pstree · sleep · time · top · wait
User management /environment	finger · mesg · passwd · su · sudo · uname · w · wall · who · whoami · write
Text processing	awk · comm · ed · ex · head · less · more · sed · sort · tail · uniq · wc · xargs
Shell programming	alias · echo · expr · false · printf · test · true · unset
Communications	inetd · netstat · ping · rlogin · traceroute
Searching	find · grep · strings
Miscellaneous	dd · lp · man · size · yes

KWANGWOON UNIVERSITY

2019년 1학기 **시스템프로그래밍실습** 과제 1차

Linux 기초

System Software Laboratory
College of Software and Convergence
Kwangwoon Univ.

Contents

1-1. Ubuntu Installation

- **과제 내용**
  - Ubuntu를 설치하는 과정을 캡쳐하고 설명

- **요구 사항**
  - 설치 하는 방법(multi-booting, virtual machine, …)은 무관
    - 단, Virtual machine을 사용할 경우, tool(VMWare, …) 설치 과정은 과제에서 제외
  - Ubuntu 계정 생성 시, 계정 ID는 “sp학번”으로 할 것
    - ex) sp2017202000
  - 두 장 이내로 작성

1-2. Usage of Linux Commands

- **과제 내용**
  - 실습 시간에 배운 Linux 명령어를 사용하고, 이를 캡쳐하고 설명

- **요구 사항**
  - 2주차 강의 자료에서 다룬 아래의 명령어를 모두 사용하고, 과정을 캡쳐 및 설명
    - Man, cat, pwd, cd, ls, chmod, mkdir, rmdir, rm, cp, mv, ln, touch, exit, kill, ps, pstree, time, passwd, uname, wc, more, echo, alias, find, grep
  - 3주차 강의 자료에서 다룬 아래의 명령어를 모두 사용하고, 과정을 캡쳐 및 설명
    - vi
      - 기본 명령어(삽입, 삭제, 데이터 저장, 검색, 패턴에 의한 치환) 각각에서 하나의 명령어만 수행
        - e.g. 삽입 명령어는 'i' 만, 삭제 명령어는 'dd'만, …
    - make, gdb
  - 캡쳐한 내용에 필히 한 줄 이상의 설명을 작성할 것

Report Requirements

- **표지**
  - 다음의 내용은 필히 기록
    - 과제 이름
      - e.g. 1차 과제 – Linux 기초
    - 분반 (요일, 담당 교수님)
    - 본인 인적 사항 (학번, 이름)

- **과제 내용**
  - 아래의 내용을 하위 과제(1-1, 1-2)마다 작성
    - Introduction : 5줄 이하
      - 과제 소개.
      - Background 제외
    - Result
      - 수행한 내용을 캡쳐 이미지와 함께 설명
    - Reference
      - 과제하면서 참고한 내용을 구체적으로 기록
      - 강의 자료만 이용한 경우 생략 가능
      - e.g. 친구 도움, 책, 인터넷 사이트 주소

Report Requirements (cont'd)

- **Softcopy Upload**
  - 보고서를 pdf로 변환하여 제출
  - 보고서 이름은 *실습 요일_학번_이름* 으로 수정
    - e.g. 월1,2 ➔ **mon**_2017202000_홍길동.**pdf**
    - e.g. 화3,4 ➔ **tue**_2017202000_홍길동.**pdf**
    - e.g. 금5,6 ➔ **fri**_2017202000_홍길동.**pdf**

  - U-Campus의 과제 제출에 3월 29일(금) 23:59:59까지 제출
  - 미리 공지한 바와 같이, delay 받지 않음.

- **과제 관련 질문**
  - 해당 과제 출제 담당 조교에게 이메일로 문의 (**남건욱** 조교 / ngotic@kw.ac.kr)
  - 과제 제출 마감 당일에는 오후 4시까지 도착한 질문 메일에만 답변

- **1차 퀴즈**
  - 3월 30일(토), 시간 및 장소는 U-campus 및 Q&A 게시판에 추후 공지