

Chapter 02 디렉터리와 파일 사용법

목차

00 개요

01 리눅스의 파일과 디렉터리

02 디렉터리 관련 명령

03 파일 관련 명령

학습목표

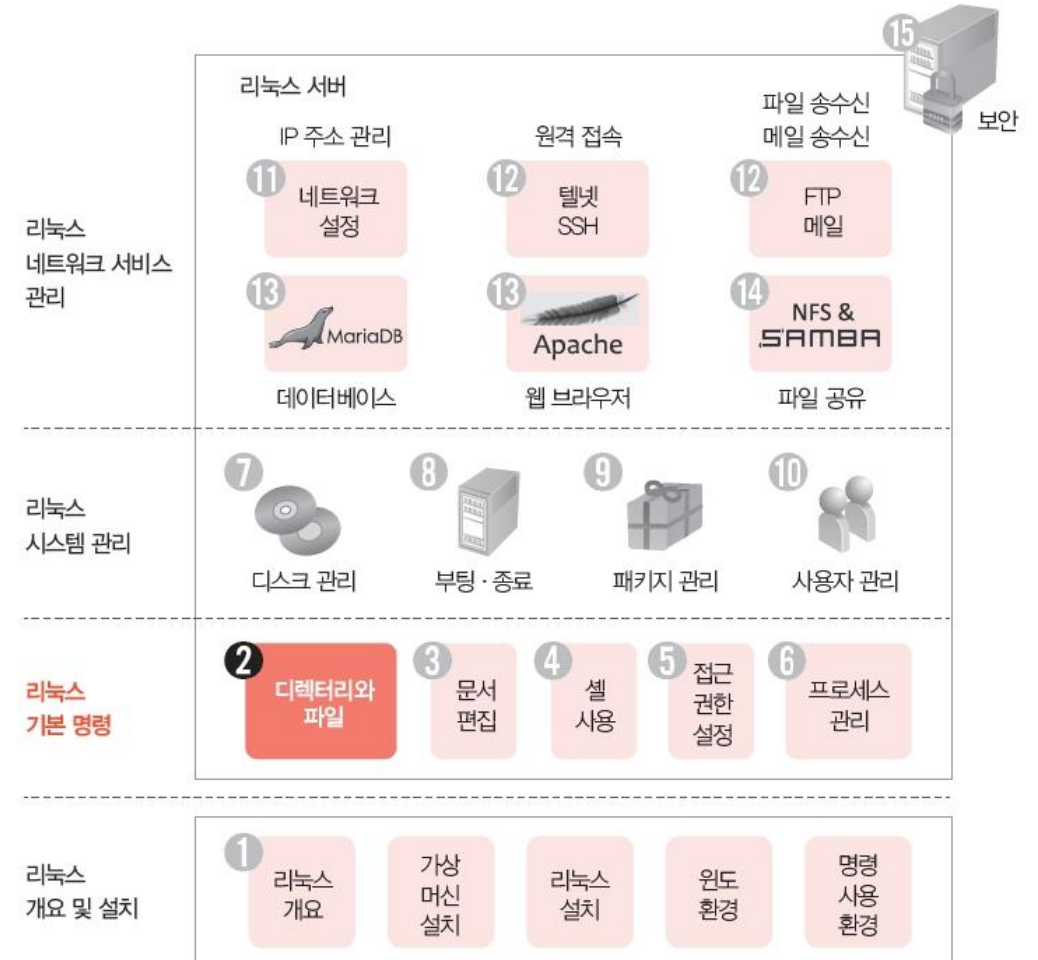


- 리눅스 파일의 종류와 특징을 설명할 수 있다.
- 디렉터리 계층 구조를 보고 절대 경로명과 상대 경로명을 작성할 수 있다.
- 디렉터리를 이동하고, 디렉터리의 내용을 확인할 수 있다.
- 디렉터리를 만들고 삭제할 수 있다.
- 다양한 명령으로 파일 내용을 확인할 수 있다.
- 파일을 복사/이동/삭제할 수 있다.
- 파일 링크의 특징을 설명하고, 하드 링크와 심볼릭 링크를 만들 수 있다.
- 파일의 내용과 위치를 검색할 수 있다.

00 개요

00. 개요

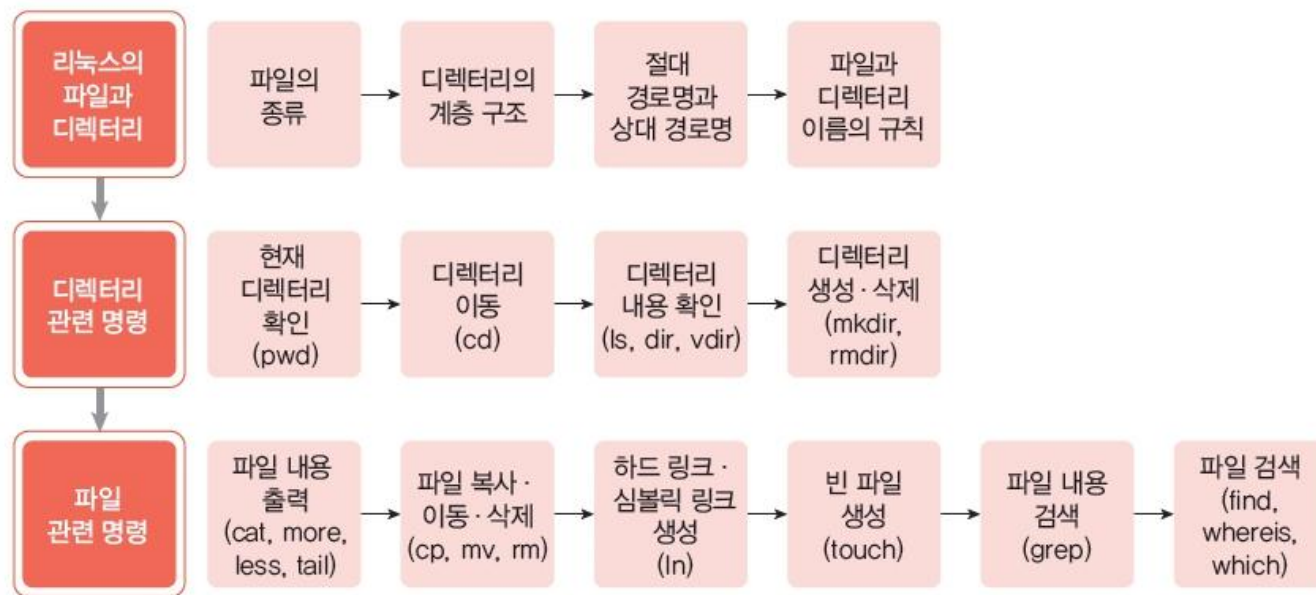
- 리눅스 학습 맵에서 2장의 위치
 - 2장은 제2단계 중 디렉터리와 파일을 다루는 명령을 학습한다.



00. 개요

■ 2장의 내용 구성

- 리눅스 파일의 종류와 특징, 디렉터리 사용 명령, 파일을 다루는 명령 순으로 구성
- 파일의 종류 이해
- 디렉터리를 사용하기 위해 디렉터리의 계층 구조 학습
- 디렉터리를 이동·생성·삭제하는 명령 학습
- 파일의 내용을 보는 명령과 파일을 생성·복사·이동·삭제하는 명령 학습



01 리눅스의 파일과 디렉터리

01. 리눅스의 파일과 디렉터리

■ 파일: 관련있는 정보들의 집합

■ 파일의 종류

- 일반 파일: 텍스트 파일, 실행파일 이미지 파일 등 주로 데이터를 저장
- 디렉터리
 - 리눅스에서는 디렉터리도 파일로 취급
 - 해당 디렉터리에 저장된 파일이나 하위 디렉터리에 대한 정보 저장
- 심볼릭 링크
 - 원본 파일을 다른 파일명으로 지정한것
 - 원도의 바로가기 파일과 유사
- 장치파일
 - 하드디스크나 키보드 같은 각종 장치도 파일로 취급

01. 리눅스의 파일과 디렉터리

■ 파일의 종류를 알아보는 명령: file

```
user1@myubuntu:~$ file .profile
.profile: ASCII text
user1@myubuntu:~$ file 다운로드
다운로드: directory
```

← 텍스트 파일, 디렉터리

```
user1@myubuntu:~$ file /bin/bash
/bin/bash: ELF 64-bit LSB pie executable, x86-64, version 1 (SYSV),
(생략)
```

↑ 실행 파일

01. 리눅스의 파일과 디렉터리

■ 디렉터리 계층구조

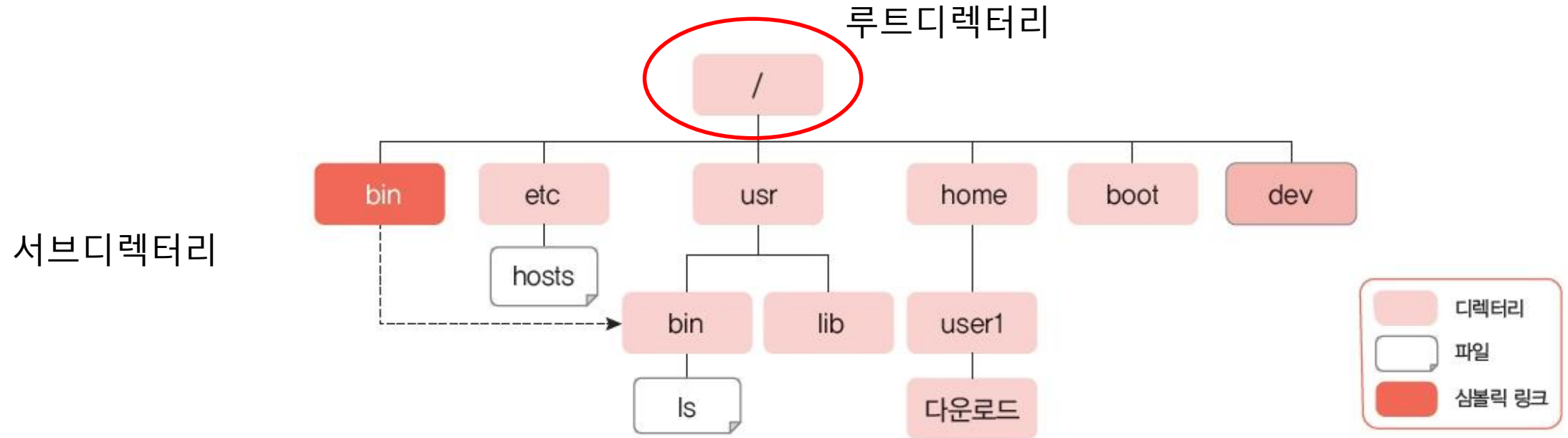


그림 2-1 디렉터리 계층 구조의 예

부모디렉터리(상위디렉터리)

자식디렉터리(하위디렉터리)

작업디렉터리, 홈 디렉터리

01. 리눅스의 파일과 디렉터리

■ 디렉터리의 주요기능

표 2-1 디렉터리의 주요 기능

디렉터리	기능
dev	장치 파일이 담긴 디렉터리다.
home	사용자 홈 디렉터리가 생성되는 디렉터리다.
media	DVD/CD나 USB 같은 외부 장치를 연결(마운트라고 함)하는 디렉터리다.
opt	추가 패키지가 설치되는 디렉터리다.
root	root 계정의 홈 디렉터리다. 루트(/) 디렉터리와 다른 것이므로 혼동하지 않도록 한다.
sys	리눅스 커널과 관련된 파일이 있는 디렉터리다.
usr	기본 실행 파일과 라이브러리 파일, 헤더 파일 등 많은 파일이 있다. 참고로 usr는 'Unix System Resource'의 약자다.
boot	부팅에 필요한 커널 파일을 가지고 있다.
etc	리눅스 설정을 위한 각종 파일을 가지고 있다.
lost+found	파일 시스템에 문제가 발생하여 복구할 경우, 문제가 되는 파일이 저장되는 디렉터리로 보통은 비어 있다.
mnt	파일 시스템을 임시로 마운트하는 디렉터리다.
proc	프로세스 정보 등 커널 관련 정보가 저장되는 디렉터리다.
run	실행 중인 서비스와 관련된 파일이 저장된다.
srv	FTP나 Web 등 시스템에서 제공하는 서비스의 데이터가 저장된다.
tmp	시스템 사용 중에 발생하는 임시 데이터가 저장된다. 이 디렉터리에 있는 파일은 재시작하면 모두 삭제된다.
var	시스템 운영 중에 발생하는 데이터나 로그 등 내용이 자주 바뀌는 파일이 주로 저장된다.

01. 리눅스의 파일과 디렉터리

■ 절대경로명과 상대경로명

표 2-2 절대 경로명과 상대 경로명의 특징

구분	특징
절대 경로명	<ul style="list-style-type: none">• 반드시 /로 시작한다.• 루트 디렉터리부터 시작하여 특정 파일이나 디렉터리의 위치에 이르기까지 중간에 있는 모든 디렉터리의 이름을 표시한다.• 특정 위치를 가리키는 절대 경로명은 항상 동일하다.
상대 경로명	<ul style="list-style-type: none">• / 이외의 문자로 시작한다.• 현재 디렉터를 기준으로 서브 디렉터리로 내려가면 그냥 서브 디렉터리명으로 시작한다.• 현재 디렉터를 기준으로 상위 디렉터리로 가려면 ..(마침표 두 개)로 시작한다.• 상대 경로명은 현재 디렉터리가 어디냐에 따라 달라진다.

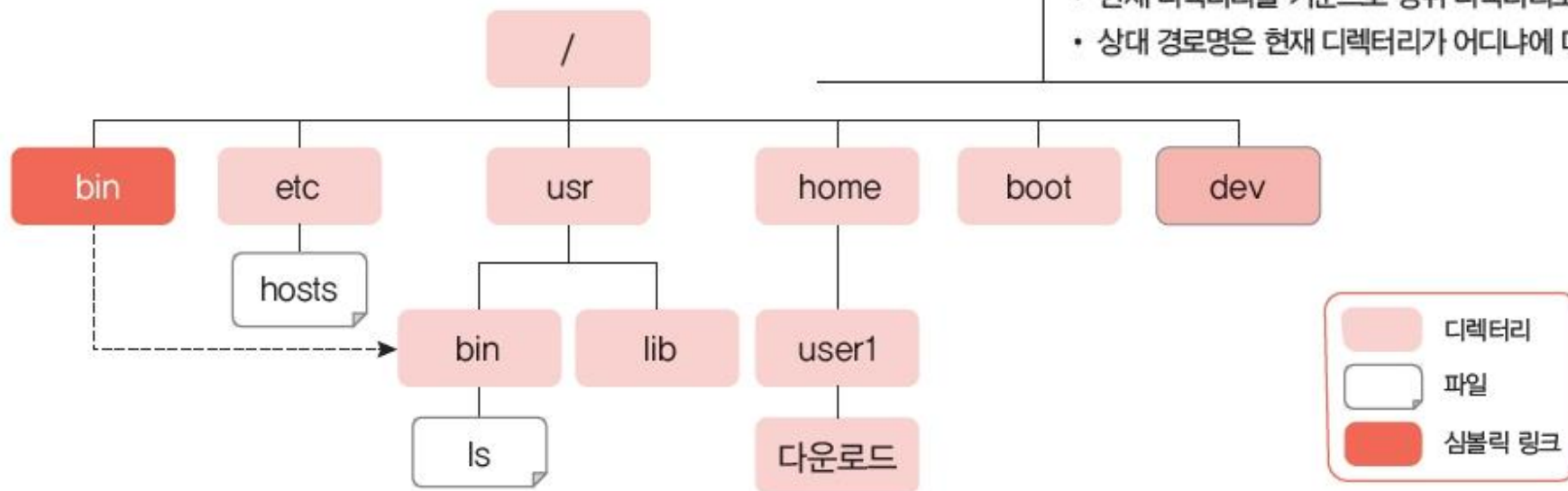


그림 2-1 디렉터리 계층 구조의 예

01. 리눅스의 파일과 디렉터리

■ 파일과 디렉터리 이름의 규칙

표 2-3 파일과 디렉터리 이름을 정하는 규칙과 예

규칙	<ul style="list-style-type: none">• 파일과 디렉터리의 이름에는 /를 사용할 수 없다. /는 경로명에서 구분자로 사용하기 때문이다.• 파일과 디렉터리의 이름에는 알파벳, 숫자, 하이픈(-), 언더스코어(_), 마침표(.)를 사용할 수 있다.• 파일과 디렉터리의 이름에는 /를 제외한 특수문자를 사용할 수 있으나 >, , :, &와 공백문자의 사용은 피하는 것이 좋다.• 파일과 디렉터리 이름의 영문은 대문자와 소문자를 구별하여 다른 글자로 취급한다.• 파일과 디렉터리의 이름이 .(마침표)로 시작하면 숨김 파일로 간주한다.• 파일 이름에는 널 문자("\0")를 사용하지 않는다.
예	<ul style="list-style-type: none">• 좋은 이름 예: game.txt, hello.c, test, sample11• 나쁜 이름 예: &game, *dir, my home, game\• 사용할 수 없는 이름 예: myhome/, /test, bad/name

02 디렉터리 관련 명령

02 디렉터리 관련 명령

■ 현재 디렉터리 확인

pwd

- **기능** 현재 디렉터리의 위치를 확인한다. 즉 현재 디렉터리의 절대 경로를 출력한다.
- **형식** pwd

```
user1@myubuntu:~$ pwd  
/home/user1
```

02 디렉터리 관련 명령

■ 디렉터리 이동

cd

- **기능** 지정한 디렉터리로 이동한다.
- **형식** cd [디렉터리]
- **사용 예** cd, cd /tmp, cd 다운로드

홈 디렉터리로 바로 이동
cd ~ 또는 cd

```
user1@myubuntu:~$ cd /tmp
user1@myubuntu:/tmp$ pwd
/tmp
```



절대경로명 사용

```
user1@myubuntu:/tmp$ cd ../usr/lib
user1@myubuntu:/usr/lib$ pwd
/usr/lib
```



상대경로명 사용

02 디렉터리 관련 명령

■ 디렉터리 내용 확인

ls

- **기능** 디렉터리의 내용을 출력한다.
- **형식** ls [옵션] [디렉터리(파일)]
- **옵션**
 - a: 숨김 파일을 포함하여 모든 파일의 목록을 출력한다.
 - d: 디렉터리 자체의 정보를 출력한다.
 - i: 첫 번째 행에 inode 번호를 출력한다.
 - l: 파일의 상세 정보를 출력한다.
 - A: .(마침표)와 ..(마침표 두 개)를 제외한 모든 파일 목록을 출력한다.
 - F: 파일의 종류를 표시한다(*: 실행 파일, /: 디렉터리, @: 심볼릭 링크).
 - L: 심볼릭 링크 파일의 경우 원본 파일의 정보를 출력한다.
 - R: 하위 디렉터리의 목록까지 출력한다.
- **사용 예** ls ls -F ls -al /tmp

02 디렉터리 관련 명령

- 현재 디렉터리 내용 확인: ls

```
user1@myubuntu:~$ ls
공개  다운로드  문서  바탕화면  비디오  사진  음악  템플릿
```

- 숨김 파일 확인하기: ls -a
 - 숨김파일: . 으로 시작하는 파일

```
user1@myubuntu:~$ ls -a
.          .bashrc  .profile  문서      음악
..         .cache   .sudo_as_admin_successful  바탕화면  템플릿
.bash_history .config  공개      비디오
.bash_logout .local   다운로드  사진
```

- 파일의 종류 표시: ls -F

```
user1@myubuntu:~$ ls -F
공개/  다운로드/  문서/  바탕화면/  비디오/  사진/  음악/  템플릿/
```

02 디렉터리 관련 명령

- 지정한 디렉터리의 내용 출력하기
 - ls 디렉터리명
- 상세한 정보 출력하기: ls -l

```
user1@myubuntu:~$ ls -l
```

합계 32

```
drwxr-xr-x 2 user1 user1 4096 11월 21 12:05 공개
drwxr-xr-x 2 user1 user1 4096 11월 21 12:05 다운로드
drwxr-xr-x 2 user1 user1 4096 11월 21 12:05 문서
drwxr-xr-x 2 user1 user1 4096 11월 21 12:05 바탕화면
drwxr-xr-x 2 user1 user1 4096 11월 21 12:05 비디오
drwxr-xr-x 2 user1 user1 4096 11월 21 12:05 사진
drwxr-xr-x 2 user1 user1 4096 11월 21 12:05 음악
drwxr-xr-x 2 user1 user1 4096 11월 21 12:05 템플릿
```

```
user1@myubuntu:~$ ls /tmp
VMwareDnD
dbus-cuuB0gWHL8
snap.firefox
snap.snap-store
```

표 2-4 파일의 상세 정보

필드 번호	필드 값	의미
1	d	다음과 같은 파일 종류를 나타낸다. -: 일반(정규) 파일 d: 디렉터리 파일 l: 심볼릭 링크 파일 b: 블록 단위로 읽고 쓰는 블록 장치 파일 c: 섹터 단위로 읽고 쓰는 문자 장치 파일 p: 파이프 파일(프로세스 간 통신에 사용되는 특수 파일) s: 소켓(네트워크 통신에 사용되는 특수 파일)
2	rw-r--r--	파일 접근 권한, 파일의 소유자, 그룹, 기타 사용자가 읽고 수정하고 실행할 수 있는 권한이 어떻게 부여되어 있는지를 보여준다.
3	2	하드 링크의 개수
4	user1	파일 소유자
5	user1	파일이 속한 그룹
6	6	파일 크기(바이트 단위)
7	11월 21 12:05	파일이 마지막으로 수정된 시간
8	공개	파일명

02 디렉터리 관련 명령

- 디렉터리의 자체 정보 확인하기

- ls -ld

```
user1@myubuntu:~$ ls -l /
합계 945456
lrwxrwxrwx   1 root root          7 11월 21 11:10 bin → usr/bin
drwxr-xr-x   4 root root    4096 12월  4 11:07 boot
drwxrwxr-x   2 root root    4096 11월 21 11:11 cdrom
drwxr-xr-x  20 root root   4300 12월  4 13:03 dev
drwxr-xr-x 129 root root  12288 12월  5 12:01 etc
(생략)
user1@myubuntu:~$ ls -ld /
drwxr-xr-x 20 root root 4096 11월 21 11:11 /
```

- 파일이 있는지 확인하기

- 비슷한 명령: dir, vdir

```
user1@myubuntu:~$ ls .bashrc
.bashrc
user1@myubuntu:~$ ls game
ls: 'game'에 접근할 수 없습니다: 그런 파일이나 디렉터리가 없습니다
```

02 디렉터리 관련 명령

■ 디렉터리 생성

mkdir

- **기능** 디렉터를 생성한다.
- **형식** mkdir [옵션] [디렉터리]
- **옵션** -p: 하위 디렉터를 계층적으로 생성할 때 중간 단계의 디렉터리가 없으면 자동으로 중간 단계 디렉터를 생성하고 지정한 디렉터를 생성한다.
- **사용 예** mkdir temp

02 디렉터리 관련 명령

- 디렉터리 한 개 만들기

```
user1@myubuntu:~$ mkdir temp
user1@myubuntu:~$ ls temp
user1@myubuntu:~$ ls
temp  공개  다운로드  문서  바탕화면  비디오  사진  음악  템플릿
```

- 디렉터리 여러 개를 동시에 만들기

```
user1@myubuntu:~$ mkdir tmp1 tmp2 tmp3
user1@myubuntu:~$ ls
temp  tmp2  공개      문서      비디오  음악
tmp1  tmp3  다운로드  바탕화면  사진    템플릿
```

02 디렉터리 관련 명령

- 중간 디렉터리를 자동으로 만들기: `mkdir -p`

```
user1@myubuntu:~$ mkdir temp/mid/han
```

```
mkdir: `temp/mid/han' 디렉터를 만들 수 없습니다: 그런 파일이나 디렉터리가 없습니다
```



```
user1@myubuntu:~$ mkdir -p temp/mid/han
```

```
user1@myubuntu:~$ ls -R temp
```

```
temp:
```

```
mid
```

```
temp/mid:
```

```
han
```

```
temp/mid/han:
```

02 디렉터리 관련 명령

■ 디렉터리 삭제

rmmdir

- **기능** 디렉터를 삭제한다.
- **형식** `rmmdir [옵션] [디렉터리]`
- **옵션** `-p`: 지정한 디렉터를 삭제하고, 그 디렉터리의 부모 디렉터리가 빈 디렉터리일 경우 부모 디렉터리도 자동으로 삭제한다.
- **사용 예** `rmmdir temp`

02 디렉터리 관련 명령

- 디렉터리 삭제

```
user1@myubuntu:~$ rmdir tmp3
user1@myubuntu:~$ ls
temp  tmp1  tmp2  공개  다운로드  문서  바탕화면  비디오  사진  음악  템플릿
```

- 빈 디렉터리가 아니면 삭제 불가

```
user1@myubuntu:~$ rmdir temp
rmdir: failed to remove 'temp': 디렉터리가 비어있지 않음
```

89p. 따라해보기 : 디렉터리 만들고 삭제하기

① 홈 디렉터리로 이동

```
user1@myubuntu:~$ cd  
user1@myubuntu:~$ pwd  
/home/user1
```

② 실습을 위한 기본 디렉터리 생성

```
user1@myubuntu:~$ mkdir linux_ex  
user1@myubuntu:~$ cd linux_ex  
user1@myubuntu:~/linux_ex$
```

③ ch2 디렉터리를 만들고 그 디렉터리로 이동

```
user1@myubuntu:~/linux_ex$ mkdir ch2  
user1@myubuntu:~/linux_ex$ cd ch2  
user1@myubuntu:~/linux_ex/ch2$ pwd  
/home/user1/linux_ex/ch2
```

89p. 따라해보기 : 디렉터리 만들고 삭제하기

④ one, two, three 디렉터를 동시에 만들기

```
user1@myubuntu:~/linux_ex/ch2$ mkdir one two three
user1@myubuntu:~/linux_ex/ch2$ ls
one  three  two
```

⑤ one 디렉터리 아래에 tmp/test 디렉터리 생성

- 중간 경로인 tmp 디렉터리는 자동 생성

```
user1@myubuntu:~/linux_ex/ch2$ mkdir -p one/tmp/test
user1@myubuntu:~/linux_ex/ch2$ ls -R one
```

one:

tmp

one/tmp:

test

⑥ one 디렉터를 rmdir 명령으로 삭제

- 삭제되는가?

one/tmp/test:

⑦ two, three 디렉터를 동시에 삭제

03 파일 관련 명령

03 파일 관련 명령

■ 파일 내용 연속 출력: cat

cat

- **기능** 파일 내용을 출력한다.
- **형식** cat [옵션] [파일]
- **옵션** -n: 행 번호를 붙여서 출력한다.
- **사용 예** cat file1 cat -n file1

```
user1@myubuntu:~$ cat /etc/hosts
127.0.0.1 localhost
127.0.1.1 myubuntu

# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
```

```
user1@myubuntu:~$ cat -n /etc/hosts
1 127.0.0.1 localhost
2 127.0.1.1 myubuntu
3
4 # The following lines are desirable for
5 ::1 ip6-localhost ip6-loopback
6 fe00::0 ip6-localnet
7 ff00::0 ip6-mcastprefix
8 ff02::1 ip6-allnodes
9 ff02::2 ip6-allrouters
```

03 파일 관련 명령

■ 파일 내용을 화면 단위로 출력하기: more

more

- **기능** 파일 내용을 화면 단위로 출력한다.
- **형식** more [옵션] [파일]
- **옵션** + 행 번호: 출력을 시작할 행 번호를 지정한다.
- **사용 예** more file1

- 스페이스바: 다음 화면 출력
- 엔터키: 한 행씩 스크롤

```
user1@myubuntu:~$ more /etc/services
# Network services, Internet style
#
# Note that it is presently the policy of IANA to assign a single well-known
# port number for both TCP and UDP; hence, officially ports have two entries
# even if the protocol doesn't support UDP operations.
#
(생략)
qotd          17/tcp          quote
chargen       19/tcp          ttytst source
chargen       19/udp          ttytst source
ftp-data      20/tcp
ftp           21/tcp
--More--(5%)
```

03 파일 관련 명령

■ 파일 내용을 화면 단위로 출력하기: less

less

- **기능** 파일 내용을 화면 단위로 출력한다.
- **형식** less [파일]
- **사용 예** less file1

표 2-5 less 명령에서 사용하는 키와 동작

키	동작
j, 방향키(아래)	한 행씩 다음 행으로 스크롤한다.
k, 방향키(위)	한 행씩 이전 행으로 스크롤한다.
[Space Bar], [Ctrl]+f	다음 화면으로 이동한다.
[Ctrl]+b	이전 화면으로 이동한다.

```
user1@myubuntu:~$ less /etc/services
# Network services, Internet style
#
# Note that it is presently the policy of IANA to assign a single well-known
# port number for both TCP and UDP; hence, officially ports have two entries
# even if the protocol doesn't support UDP operations.
#
(생략)
qotd          17/tcp          quote
chargen       19/tcp          ttytst source
chargen       19/udp          ttytst source
ftp-data      20/tcp
ftp           21/tcp
/etc/services
```

03 파일 관련 명령

■ 파일 내용의 뒷부분 출력하기: tail

tail

- **기능** 파일 뒷부분의 몇 행을 출력한다.
- **형식** tail [옵션] [파일]
- **옵션** +행 번호: 지정한 행부터 끝까지 출력한다.
 - 숫자: 화면에 출력할 행의 수를 지정한다(기본 값은 10).
 - f: 파일 출력을 종료하지 않고 주기적으로 계속 출력한다.

```
user1@myubuntu:~$ tail -7 /etc/services
asp                27374/udp
csync2             30865/tcp
dircproxy          57000/tcp
tfido              60177/tcp
fido               60179/tcp

# Local services
```

숫자 옵션
사용

```
user1@myubuntu:~$ tail /etc/services
sgi-cad            17004/tcp          # Cluster Admin daemon
binkp              24554/tcp          # binkp fidonet proto
asp                27374/tcp          # Address Search Prot
asp                27374/udp
csync2             30865/tcp          # cluster synchroniza
dircproxy          57000/tcp          # Detachable IRC Prox
tfido              60177/tcp          # fidonet EMSI over t
fido               60179/tcp          # fidonet EMSI over T

# Local services
```

* tail -f : 파일 내용을 주기적으로 반복 출력

97p. 따라해보기 : 파일 내용 출력하기

- ① less 명령으로 /etc/services 파일의 내용을 출력
- ② <스페이스바> 를 눌러 다음 페이지를 출력
- ③ k 키를 네 번 눌러 위로 다시 이동
- ④ '/TCP'를 입력하여 파일 내용 중에서 문자열 'TCP'가 있는 곳을 검색
- ⑤ n 키를 눌러 문자열 'TCP'를 계속 검색
- ⑥ q 키를 눌러 less 명령을 종료

03 파일 관련 명령

■ 파일 복사

cp

- **기능** 파일이나 디렉터리를 복사한다.
- **형식** cp [옵션] [파일1(디렉터리1)] [파일2(디렉터리2)]
- **옵션** -i: 파일2가 이미 존재하면 덮어쓸 것인지 물어본다.
-r: 디렉터리를 복사할 때 지정한다.
- **사용 예** cp file1 file2
cp f1 f2 f3 dir1
cp -r dir1 dir2

03 파일 관련 명령

- 두 인자가 모두 파일인 경우

```
user1@myubuntu:~/linux_ex/ch2$ cp /etc/hosts text1
user1@myubuntu:~/linux_ex/ch2$ ls
one  text1
```

- 두 번째 인자가 디렉터리인 경우

```
user1@myubuntu:~/linux_ex/ch2$ mkdir temp
user1@myubuntu:~/linux_ex/ch2$ cp text1 temp
user1@myubuntu:~/linux_ex/ch2$ ls temp
text1
```

원본과 다른
파일명 지정

```
user1@myubuntu:~/linux_ex/ch2$ cp text1 temp/text2
user1@myubuntu:~/linux_ex/ch2$ ls temp
text1  text2
```

쓰기 권한이 없는 디렉
터리로 복사할 경우

```
user1@myubuntu:~/linux_ex/ch2$ cp text1 /etc
cp: 일반 파일 '/etc/text1'을(를) 생성할 수 없음: 허가 거부
```

03 파일 관련 명령

- 인자를 여러 개 지정할 경우

마지막에 지정한 디렉터리로 파일들 복사

```
user1@myubuntu:~/linux_ex/ch2$ cp /etc/hosts /etc/services temp
user1@myubuntu:~/linux_ex/ch2$ ls temp
hosts  services  text1  text2
```

- i 옵션 사용: 이미 있는 파일인 경우 덮어쓸 것인지 질문

```
user1@myubuntu:~/linux_ex/ch2$ cp -i /etc/hosts text1
cp: 'text1'를 덮어쓸까요? n
```

- 디렉터리 복사하기: cp -r

```
user1@myubuntu:~/linux_ex/ch2$ cp -r temp temp2
user1@myubuntu:~/linux_ex/ch2$ ls temp2
hosts  services  text1  text2
```

03 파일 관련 명령

■ 파일 이동과 파일명 변경

mv

- **기능** 파일 또는 디렉터리를 이동하거나 이름을 바꾼다.
- **형식** mv [옵션] [파일1(디렉터리1)] [파일2(디렉터리2)]
- **옵션** -i: 파일2(디렉터리2)가 존재하면 덮어쓸 것인지 물어본다.
- **사용 예** mv file1 file2

03 파일 관련 명령

- 파일을 파일로 이동하기(파일명 바꾸기)

```
user1@myubuntu:~/linux_ex/ch2$ mv text1 data1
user1@myubuntu:~/linux_ex/ch2$ ls
data1  one  temp  temp2
```

- 파일을 다른 디렉터리로 이동하기

```
user1@myubuntu:~/linux_ex/ch2$ mv data1 temp
user1@myubuntu:~/linux_ex/ch2$ ls
one  temp  temp2
user1@myubuntu:~/linux_ex/ch2$ ls temp
data1  hosts  services  text1  text2
```

파일을 temp 디렉터리로
이동

```
user1@myubuntu:~/linux_ex/ch2$ cp temp/data1 text1
user1@myubuntu:~/linux_ex/ch2$ ls
one  temp  temp2  text1
user1@myubuntu:~/linux_ex/ch2$ mv text1 temp/data2
user1@myubuntu:~/linux_ex/ch2$ ls temp
data1  data2  hosts  services  text1  text2
```

파일을 temp 디렉터리로
이동하고 이름도 바꾸기

03 파일 관련 명령

- 여러 파일을 디렉터리로 이동하기

```
user1@myubuntu:~/linux_ex/ch2$ ls temp
data1 data2 hosts services text1 text2
user1@myubuntu:~/linux_ex/ch2$ mv temp/data1 temp/data2 .
user1@myubuntu:~/linux_ex/ch2$ ls
data1 data2 one temp temp2
user1@myubuntu:~/linux_ex/ch2$ ls temp
hosts services text1 text2
```

- -i 옵션 사용하기

```
user1@myubuntu:~/linux_ex/ch2$ mv -i data1 data2
mv: 'data2'를 덮어쓸까요? n
user1@myubuntu:~/linux_ex/ch2$ ls
data1 data2 one temp temp2
```

- 디렉터리를 디렉터리로 이동하기(디렉터리명 바꾸기)
 - mv 명령에서 인자를 모두 디렉터리로 지정하면 디렉터리가 이동

03 파일 관련 명령

■ 파일 삭제

rm

- **기능** 파일을 삭제한다.
- **형식** rm [옵션] [파일 또는 디렉터리]
- **옵션** -i: 파일을 정말 삭제할 것인지 확인한다.
-r: 디렉터리를 삭제할 때 지정한다.
- **사용 예** rm file rm -r dir

```
user1@myubuntu:~/linux_ex/ch2$ rm data2
```

```
user1@myubuntu:~/linux_ex/ch2$ ls
```

```
data1  one  temp
```


03 파일 관련 명령

- -i 옵션 사용하기

```
user1@myubuntu:~/linux_ex/ch2$ rm -i data1
rm: 일반 파일 'data1'를 제거할까요? n
user1@myubuntu:~/linux_ex/ch2$ ls
data1 one temp
```

- 디렉터리 삭제하기: rm -r

```
user1@myubuntu:~/linux_ex/ch2/temp$ ls
hosts services temp3 text1 text2
user1@myubuntu:~/linux_ex/ch2/temp$ rm -r temp3
user1@myubuntu:~/linux_ex/ch2/temp$ ls
hosts  services  text1  text2
```

109p. 따라해보기 : 파일 복사/이동/삭제하기

- ① /etc/hosts 파일을 test.org로 복사
- ② test 디렉터리 생성
- ③ test.org 파일을 test 디렉터리로 복사
- ④ test 디렉터리에 있는 test.org의 파일명을 test.bak로 변경
- ⑤ test.org 파일을 삭제
- ⑥ test 디렉터리에 있는 test.bak를 현재 디렉터리에 test.org로 복사
- ⑦ test, one 디렉터리를 삭제

03 파일 관련 명령

■ 리눅스 파일의 구성

- 파일명 + inode + 데이터 블록

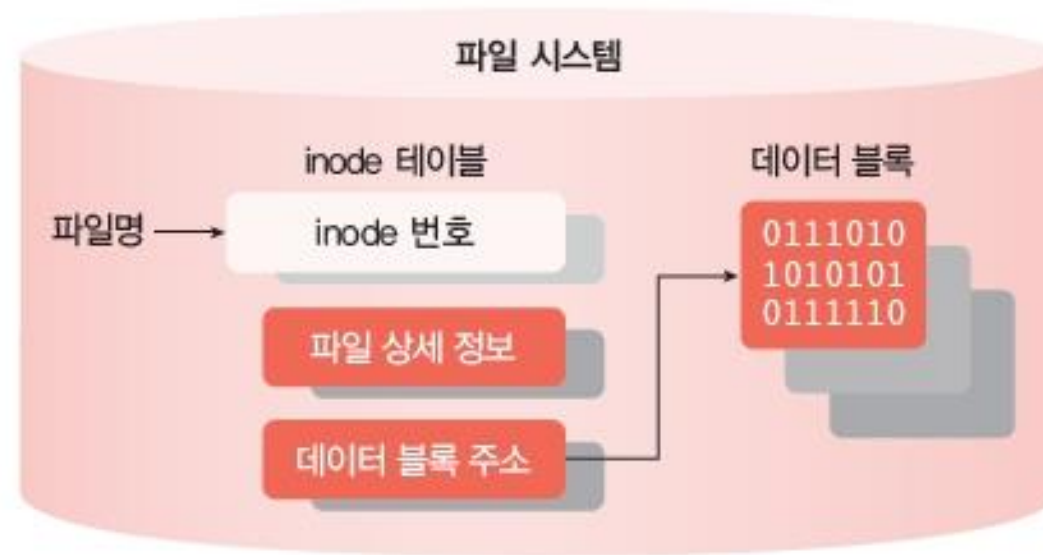


그림 2-2 리눅스 파일의 구성 요소

- 파일의 inode 번호 검색: `ls -li`

```
user1@myubuntu:~/linux_ex/ch2$ ls -li
1058724 data1  1058725 temp  1058745 test.org
```

03 파일 관련 명령

■ 파일 링크

- 기존 파일에 새로운 이름을 붙이는 것
- 하드 링크 만들기: `ln`
 - 한 파일에 여러 개의 이름 붙이기

`ln`

- **기능** 파일의 링크를 생성한다.
- **형식** `ln [옵션] [원본 파일] [링크 파일]`
- **옵션** `-s`: 심볼릭 링크 파일을 생성한다.
- **사용 예** `ln test hdtest ln -s test sbtest`

03 파일 관련 명령

```
user1@myubuntu:~/linux_ex/ch2$ ls -l
```

합계 12

```
-rw-r--r--  1 user1 user1  221 12월  7 20:36 data1
drwxrwxr-x  2 user1 user1 4096 12월  7 20:50 temp
-rw-r--r--  1 user1 user1  221 12월  7 20:56 test.org
```

```
user1@myubuntu:~/linux_ex/ch2$ ln data1 data1.ln
```

```
user1@myubuntu:~/linux_ex/ch2$ ls -l
```

합계 16

```
-rw-r--r--  2 user1 user1  221 12월  7 20:36 data1
-rw-r--r--  2 user1 user1  221 12월  7 20:36 data1.ln
drwxrwxr-x  2 user1 user1 4096 12월  7 20:50 temp
-rw-r--r--  1 user1 user1  221 12월  7 20:56 test.org
```

```
user1@myubuntu:~/linux_ex/ch2$ ls -li
```

```
1058724 data1 1058724 data1.ln 1058725 temp 1058745 test.org
```

03 파일 관련 명령

- 링크와 파일을 복사한 것의 차이

```
user1@myubuntu:~/linux_ex/ch2$ cp data1 data1.cp
user1@myubuntu:~/linux_ex/ch2$ ls -li
1058724 data1      1058724 data1.ln  1058745 test.org
1058746 data1.cp  1058725 temp
```

- 심볼릭 링크 만들기: -s 옵션

```
user1@myubuntu:~/linux_ex/ch2$ ln -s data1 data1.sl
user1@myubuntu:~/linux_ex/ch2$ ls -li
1058747 data1.sl  1058746 data1.cp  1058725 temp
1058724 data1      1058724 data1.ln  1058745 test.org
```

```
user1@myubuntu:~/linux_ex/ch2$ ls -l data1.sl
lrwxrwxrwx. 1 user1 user1 5  12월 7 21:38 data1.sl -> data1
```

03 파일 관련 명령

- 심볼릭 링크와 하드 링크의 차이

- 심볼릭 링크는

- 파일의 종류가 l(소문자 L)로 표시된다.
- 하드 링크의 개수가 하나다. 즉, 원본 파일에 이름을 추가하는 것이 아니다.
- 파일명 뒤에 원본 파일의 이름이 표시된다(-> data1).
- inode 번호가 원본 파일과 다르다. 즉, 원본 파일과 심볼릭 링크 파일은 별개의 파일이다.

03 파일 관련 명령

- 심볼릭 링크의 원본 파일이 삭제되면

```
user1@myubuntu:~/linux_ex/ch2$ rm data1  
user1@myubuntu:~/linux_ex/ch2$ cat data1.sl  
cat: data1.sl: 그런 파일이나 디렉터리가 없습니다
```


109p. 따라해보기 : 파일 복사/이동/삭제하기

- ① test.org 파일의 하드 링크로 test.ln을 생성
- ② test.org 파일의 하드 링크로 test.ln2를 생성
- ③ test.org와 test.ln, test.ln2의 inode 번호가 같은지 확인
- ④ test.org와 test.ln, test.ln2의 하드 링크 개수를 확인
- ⑤ temp 디렉터리에 대한 심볼릭 링크로 tmp 생성
- ⑥ temp와 tmp 디렉터리의 내용이 같다는 것을 확인
- ⑦ tmp 디렉터리와 test.ln, test.ln2, test.org 파일을 모두 삭제

03 파일 관련 명령

■ 빈 파일 만들기, 접근/수정 시간 변경하기: touch

touch

- **기능** 빈 파일을 생성한다.
- **형식** touch [-acm] [-r ref_file | -t time] [파일]
- **옵션**
 - a: 접근 시간만 변경한다.
 - m: 수정 시간만 변경한다.
 - t [[CC]YY]MMDDhhmm[.ss]: 시간을 직접 입력한다.
- **사용 예** touch test

```
user1@myubuntu:~/linux_ex/ch2$ ls -l data1.cp
-rw-r--r-- 1 user1 user1 223 12월  7 21:37 data1.cp
user1@myubuntu:~/linux_ex/ch2$ date
2021. 12. 07. (화) 21:48:22 KST
user1@myubuntu:~/linux_ex/ch2$ touch data1.cp
user1@myubuntu:~/linux_ex/ch2$ ls -l data1.cp
-rw-r--r-- 1 user1 user1 223 12월  7 21:48 data1.cp
```

```
user1@myubuntu:~/linux_ex/ch2$ touch test
user1@myubuntu:~/linux_ex/ch2$ ls -l test
-rw-rw-r-- 1 user1 user1 0 12월  7 21:47 test
```

빈 파일 생성

파일 수정 시간
수정

03 파일 관련 명령

- -t 옵션을 사용하면 변경할 시간을 지정

시간 표시

- **형식** [[CC]YY]MMDDhhmm[.ss]
- **설명** CC: 연도의 첫 두 자리
YY: 연도의 마지막 두 자리
MM: 달(01~12 범위 내 지정)
DD: 날짜(01~31 범위 내 지정)
hh: 시간(00~23 범위 내 지정)
mm: 분(00~59 범위 내 지정)
ss: 초(00~59 범위 내 지정)

표 2-6 연도 지정 방법

YY	69~99	00~68
CC	19	20

```
user1@myubuntu:~/linux_ex/ch2$ ls -l test
-rw-rw-r-- 1 user1 user1 0 12월  7 21:47 test
user1@myubuntu:~/linux_ex/ch2$ touch -t 12311200 test
user1@myubuntu:~/linux_ex/ch2$ ls -l test
-rw-rw-r--. 1 user1 user1 0 12월 31 2021 test
```

03 파일 관련 명령

■ 파일 내용 검색하기: grep

grep

- **기능** 지정한 패턴이 포함된 행을 찾는다.
- **형식** `grep [옵션] [패턴] [파일]`
- **옵션**
 - i: 대문자 · 소문자를 모두 검색한다.
 - l: 지정한 패턴이 포함된 파일명을 출력한다.
 - n: 행 번호를 출력한다.
- **사용 예**
 - `grep root /etc/passwd`
 - `grep -n unix ~/.txt`
 - `grep -l hello *.c`

```
user1@myubuntu:~/linux_ex/ch2$ cp /etc/services data
user1@myubuntu:~/linux_ex/ch2$ grep NNTP data
nntp          563/tcp          sntp          # NNTP over SSL
```

```
user1@myubuntu:~/linux_ex/ch2$ grep -n NNTP data
121:nntp 563/tcp sntp # NNTP over SSL
```

03 파일 관련 명령

■ 파일 찾기: find

find

- **기능** 지정한 위치에서 검색 조건에 맞는 파일을 찾는다.
- **형식** `find [경로] [검색 조건] [동작]`
- **검색 조건**
 - name filename: 파일명으로 검색한다.
 - type 파일 종류: 파일 종류로 검색한다.
 - user loginID: 지정한 사용자가 소유한 모든 파일을 검색한다.
 - perm 접근 권한: 지정한 사용 권한과 일치하는 파일을 검색한다.
- **동작**
 - exec 명령 {} \:: 검색된 파일에 명령을 실행한다.
 - ok 명령 {} \:: 사용자의 확인을 받아서 명령을 실행한다.
 - print: 검색된 파일의 절대 경로명을 화면에 출력한다(기본 동작).
 - ls: 검색 결과를 긴 목록 형식으로 출력한다.
- **사용 예**
 - `find ~ -name hello.c`
 - `find /tmp -user user10 -exec rm {} \;`

03 파일 관련 명령

- 예: /usr/bin 디렉터리에서 ls 파일의 위치를 찾기

```
user1@myubuntu:~/linux_ex/ch2$ find /usr/bin -name ls
/usr/bin/ls
```

- 예: -user 옵션을 사용 하여 특정 사용자 계정이 소유자인 파일을 찾기

```
user1@myubuntu:~/linux_ex/ch2$ find /home -user user1
/home/user1
/home/user1/바탕화면
/home/user1/.bashrc
/home/user1/.cache
/home/user1/.cache/tracker3
/home/user1/.cache/tracker3/files
/home/user1/.cache/tracker3/files/http%3A%2F%2Ftracker.api.gnome.org%2Fontology%2Fv3%2F
tracker%23FileSystem.db-wal
/home/user1/.cache/tracker3/files/http%3A%2F%2Ftracker.api.gnome.org%2Fontology%2Fv3%2F
tracker%23Audio.db-shm
(생략)
```

03 파일 관련 명령

- find 명령으로 검색한 모든 파일을 대상으로 동일한 작업을 수행하려면 -exec나 -ok 옵션 사용

```
user1@myubuntu:~/linux_ex/ch2$ find /tmp -user user1 -exec rm {} \;  
find: '/tmp/snap.firefox': 허가 거부  
find: '/tmp/systemd-private-2de14edc877647a39fc3b92a0d9fa6c8-power-profiles-daemon.service-B9S9zF': 허가 거부  
find: '/tmp/systemd-private-2de14edc877647a39fc3b92a0d9fa6c8-switcheroo-control.service-LYIMmg': 허가 거부  
find: '/tmp/systemd-private-2de14edc877647a39fc3b92a0d9fa6c8-systemd-timesyncd.service-t56bhl':  
허가 거부  
rm: 쓰기 보호된 일반 파일 '/tmp/.X1-lock'를(을) 지울까요? n  
rm: '/tmp/tracker-extract-3-files.1000'를 지울 수 없음: 디렉터리입니다  
find: '/tmp/systemd-private-2de14edc877647a39fc3b92a0d9fa6c8-colord.service-0ZtAdt': 허가 거부  
(생략)
```

03 파일 관련 명령

■ 명령의 위치 찾기: whereis

whereis

- **기능** 지정된 경로에서 명령의 바이너리 파일이나 매뉴얼 파일의 위치를 찾는다.
- **형식** whereis [옵션] [파일]
- **옵션** -b: 바이너리 파일만 검색한다.
-m: 매뉴얼 파일만 검색한다.
-s: 소스 파일만 검색한다.
- **사용 예** whereis ls

```
user1@myubuntu:~/linux_ex/ch2$ whereis mv
mv: /usr/bin/mv /usr/share/man/man1/mv.1.gz
```


03 파일 관련 명령

■ 명령의 위치 찾기: which

which

- **기능** 명령 파일의 위치를 찾아서 그 경로나 앨리어스를 출력한다.
- **형식** which [명령]
- **사용 예** which ls

```
user1@myubuntu:~/linux_ex/ch2$ which mv
/usr/bin/mv
```

123p. 따라해보기 : grep, find 명령 사용하기

- ① grep 명령으로 /etc/services 파일에서 문자열 'NETBIOS'가 있는 행을 찾아 행 번호와 함께 출력하기
- ② find 명령으로 홈 디렉터리에서 파일명이 data1.cp인 파일이 있는지 검색하기
- ③ find 명령으로 홈 디렉터리에서 파일명이 data1.cp인 파일을 찾아 temp 디렉터리로 이동하기

```
user1@myubuntu:~/linux_ex/ch2$ find ~ -name data1.cp -exec mv {} temp \;  
user1@myubuntu:~/linux_ex/ch2$ ls temp  
data1.cp  hosts  text1  text2
```

Thank You !