

목차

01. 리눅스 파일의 종류와 특징

02. 디렉터리 사용 명령

03. 파일 다루기

02 디렉터리 사용 명령

■ 실습

- ① 현재 위치를 확인한다. 홈 디렉터리가 아니면 홈 디렉터리로 이동한다.
- ② 실습을 위한 기본 디렉터를 만든다. 먼저 홈 디렉터리에 `linux_ex` 디렉터를 만들고 그 디렉터리로 이동한다. 앞으로 모든 실습은 이 디렉터리 아래에서 한다.
- ③ `ch2` 디렉터를 만들고 그 디렉터리로 이동하여 현재 위치를 확인한다.
- ④ `one`, `two`, `three` 디렉터를 동시에 만든다.
- ⑤ `one` 디렉터리 아래에 `tmp/test` 디렉터를 만든다. 중간 경로인 `tmp` 디렉터리가 자동 생성되도록 한다.
- ⑥ `two`, `three` 디렉터를 동시에 삭제한다.
- ⑦ 실습을 마치고 홈 디렉터리로 이동한다.

03 파일 다루기

- 파일의 내용을 보는 명령
- 파일을 복사하는 명령
- 파일을 삭제하고 이동하는 명령
- 하드 링크와 심벌릭 링크를 생성하는 명령
- 빈 파일을 만드는 명령

03 파일 다루기

■ 파일 내용 연속 출력하기

- 텍스트 파일 내용 확인

cat

기능 파일 내용을 출력한다.

형식 cat [옵션] 파일명...

옵션 -n : 행 번호를 붙여서 출력한다.

사용 예 cat file1 cat -n file1

- 예: /etc/hosts 파일 내용 확인

```
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
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
user1@myubuntu:~$
```

03 파일 다루기

■ 파일 내용 연속 출력하기

- 예: 행 번호 붙이기(-n 옵션)

```
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 IPv6 capable hosts
 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
user1@myubuntu:~$
```

03 파일 다루기

■ 화면 단위로 파일 내용 출력하기

more

기능 파일 내용을 화면 단위로 출력한다.

형식 more [옵션] 파일명...

옵션 + 행 번호 : 출력을 시작할 행 번호를 지정한다.

사용 예 more file1

- 아직 출력되지 않은 내용이 더 있으면 화면 하단에 '--More--(0%)'와 같이 표시
- 예: /etc/services 파일 내용 보기

```
user1@myubuntu:~$ more /etc/services
# Network services, Internet style
#
(생략)
# by IANA and used in the real-world or are needed by a debian package.
# If you need a huge list of used numbers please install the nmap package.
tcpmux          1/tcp    # TCP port service multiplexer
echo 7/tcp
(생략)
--More--(4%)
```

- 스페이스바: 다음 화면 출력, 엔터키: 한 줄씩 스크롤, /문자열: 해당 문자열 검색, q: 종료

03 파일 다루기

■ 개선된 화면 단위 파일 내용 출력하기

- 스크롤 되어 지나간 내용도 확인 가능

less

기능 파일 내용을 화면 단위로 출력한다.

형식 less 파일명...

사용 예 less file1

- 예: /etc/services

```
user1@myubuntu:~$ less /etc/services
# Network services, Internet style
#
(생략)
# by IANA and used in the real-world or are needed by a debian package.
# If you need a huge list of used numbers please install the nmap package.
tcpmux          1/tcp    # TCP port service multiplexer
echo 7/tcp
(생략)
/etc/services
```

03 파일 다루기

■ 파일 뒷부분 출력하기

tail

기능 파일의 뒷부분 몇 행을 출력한다.

형식 tail [옵션] 파일명...

옵션 +행 번호 : 지정한 행부터 끝까지 출력한다.
 -s숫자 : 화면에 출력할 행의 수를 지정한다(기본 값은 10).
 -f : 파일 출력이 종료되지 않고 주기적으로 계속 출력한다.

- 예: /etc/services 파일의 마지막 10행 출력

```
user1@myubuntu:~$ tail /etc/services
vboxd 20012/udp
binkp 24554/tcp # binkp fidonet protocol
asp 27374/tcp # Address Search Protocol
asp 27374/udp
csync2 30865/tcp # cluster synchronization tool
dirproxy 57000/tcp # Detachable IRC Proxy
tfido 60177/tcp # fidonet EMSI over telnet
fido 60179/tcp # fidonet EMSI over TCP
# Local services
user1@myubuntu:~$
```


03 파일 다루기

■ 파일 뒷부분 출력하기

- 지정한 숫자만큼 출력하기 : - 숫자 옵션
- 예: /etc/services 파일의 마지막 7

```
user1@myubuntu:~$ tail -7 /etc/services
asp    27374/udp
csync2  30865/tcp    # cluster synchronization tool
dircproxy 57000/tcp    # Detachable IRC Proxy
tfido   60177/tcp    # fidonet EMSI over telnet
fido    60179/tcp    # fidonet EMSI over TCP
# Local services
user1@myubuntu:~$
```

- 파일 내용을 주기적으로 반복 출력하기: -f 옵션
 - -f 옵션을 사용하면 파일 출력이 종료되지 않고 대기 상태가 되며 파일 내용이 주기적으로 반복 출력

```
user1@myubuntu:~$ tail -f /etc/services
vboxd   20012/udp
binkp   24554/tcp    # binkp fidonet protocol
asp     27374/tcp    # Address Search Protocol
asp     27374/udp
csync2   30865/tcp    # cluster synchronization tool
(생략)
^C
user1@myubuntu:~$
```

03 파일 다루기

■ 파일(디렉터리) 복사하기

cp

기능 파일이나 디렉터리를 복사한다.

형식 cp [옵션] 파일명1/디렉터리명1 파일명2/디렉터리명2

옵션 -i : 대화식 복사 방법으로 파일명2가 이미 존재할 경우 덮어쓸 것인지 물어본다.
 -r : 디렉터를 복사할 때 지정한다.

사용 예 cp file1 file2
 cp f1 f2 f3 dir1
 cp -r dir1 dir2

■ 두 인자가 모두 파일인 경우 : 파일을 다른 파일로 복사

- 예: /etc/hosts 파일을 현재 디렉터리에 text1 파일로 복사

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

03 파일 다루기

■ 두 번째 인자가 디렉터리인 경우

- 파일을 해당 디렉터리 아래에 복사
- 예: temp 디렉터리에 text1 파일 복사

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

- 예: 원본 파일과 다른 이름으로 복사

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

- 예: 쓰기 권한이 없는 디렉터리에 파일을 복사하려고 하면 다음과 같은 오류가 발생

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

03 파일 다루기

■ 인자를 여러 개 지정할 경우

- cp 명령에서 첫 번째 인자의 자리에 파일명을 여러 개 지정할 수 있는데, 두 번째 인자는 반드시 디렉터리여야 한다.
- 이럴 경우 마지막에 지정한 디렉터리로 앞서 지정한 파일들이 모두 복사된다.
- 예: /etc/hosts와 /etc/services를 temp 디렉터리에 복사

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

■ -i 옵션 사용하기

- 두번째 인자로 지정한 파일이 이미 있는 경우 덮어서 복사할 것인지 확인

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

03 파일 다루기

■ 디렉터리 복사하기

- 디렉터리를 복사하려면 -r 옵션 사용
- 예: -r 옵션을 지정하지 않을 경우

```
user1@myubuntu:~/linux_ex/ch2$ cp temp temp2
cp: 'temp' 디렉터리 생략 중
user1@myubuntu:~/linux_ex/ch2$
```

- 두 번째 인자로 지정한 목적지 디렉터리가 존재하지 않는 경우 새로 생성
- 디렉터리가 복사되면 원본 디렉터리 아래에 있던 모든 내용도 함께 복사
- 예: temp 디렉터를 temp2 디렉터리로 복사

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

- 두 번째 인자로 지정한 디렉터리가 이미 있는 디렉터리일 경우, 원본 디렉터리가 목적지 디렉터리 아래에 원본 디렉터리와 같은 이름으로 복사
- 예: temp 디렉터를 다시 temp2 디렉터리로 복사(이미 앞에서 temp2 디렉터리가 생성되었으므로 이번에는 temp 디렉터리가 temp2 디렉터리 아래에 복사)

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

03 파일 다루기

■ 파일 이동하기

mv

기능 파일을 이동한다.

형식 mv [옵션] 파일명1/디렉터리명1 파일명2/디렉터리명2

옵션 -i : 파일명2/디렉터리명2가 존재하면 덮어쓸 것인지 물어본다.

사용 예 mv file1 file2

- 파일을 다른 디렉터리로 이동하거나 파일명을 바꿀 때는 mv(move) 명령을 사용
- 디렉터리를 이동하거나 디렉터리명을 바꿀 때도 mv 명령을 사용
- mv 명령의 첫 번째 인자는 원본 파일명이나 디렉터리명을 지정하며, 두 번째 인자는 목적지 파일명이나 디렉터리명을 지정

03 파일 다루기

■ 파일을 파일로 이동하기

- 파일을 다른 파일로 이동하는 것은 결국 원본 파일의 파일명을 다른 파일명으로 바꾸는 것
- 만약 두 번째 인자로 지정한 파일명이 이미 존재하는 파일이면 원본 파일의 내용으로 덮어쓰고 기존의 내용이 삭제
- 두 번째 인자로 지정한 파일명이 존재하지 않는 파일이라면 새 파일이 생성
- 예: text1 파일을 data1 파일로 이동(파일명 변경)

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

03 파일 다루기

■ 파일을 다른 디렉터리로 이동하기

- 두 번째 인자로 디렉터리를 지정할 경우 원본 파일을 지정한 디렉터리로 이동
- 예: data1 파일을 temp 디렉터리로 이동

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

- 두 번째 인자에 디렉터리와 파일명을 함께 지정할 경우

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

- 쓰기 권한이 없는 디렉터리로 파일을 이동하려고 할 경우 오류 발생

```
user1@myubuntu:~/linux_ex/ch2$ mv temp/data2 /etc
mv: 'temp/data2'를 '/etc/data2'로 옮길 수 없음: 허가 거부
user1@myubuntu:~/linux_ex/ch2$
```


03 파일 다루기

■ 파일 여러 개를 디렉터리로 이동하기

- mv 명령으로 파일 여러 개를 지정한 디렉터리로 한 번에 이동 가능
- 두 번째 인자는 반드시 디렉터리여야 함

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

■ -i 옵션 사용하기

- 두 번째 인자에 지정한 파일명이 기존에 있는 파일일 경우 덮어서 이동할 것인지를 물어봄

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

03 파일 다루기

■ 디렉터리를 디렉터리로 이동하기

- 인자를 모두 디렉터리로 지정하면 디렉터리가 이동
- 두 번째 인자가 기존에 있던 디렉터리가 아닐 경우에는 디렉터리명이 변경
- 예: temp2 디렉터리가 temp3 디렉터리로 이름 변경

```
user1@myubuntu:~/linux_ex/ch2$ mv temp2 temp3
user1@myubuntu:~/linux_ex/ch2$ ls
data1 data2 one temp temp3
user1@myubuntu:~/linux_ex/ch2$
```

- 두 번째 인자가 기존에 있던 디렉터리일 경우, 원본 디렉터리가 두 번째 인자로 지정된 디렉터리 아래로 이동
- 예: temp3 디렉터리가 temp 디렉터리 아래로 이동

```
user1@myubuntu:~/linux_ex/ch2$ ls
data1 data2 one temp temp3
user1@myubuntu:~/linux_ex/ch2$ mv temp3 temp
user1@myubuntu:~/linux_ex/ch2$ ls
data1 data2 one temp
user1@myubuntu:~/linux_ex/ch2$ ls temp
hosts services temp3 text1 text2
user1@myubuntu:~/linux_ex/ch2$
```

03 파일 다루기

■ 파일 삭제하기

rm

기능 파일을 삭제한다.

형식 rm [옵션] 파일명/디렉터리명 ...

옵션 -i : 대화식으로 지정한 파일을 정말 삭제할 것인지 확인한다.

-r : 디렉터리를 삭제할 때 지정한다.

사용 예 rm file rm -r dir

- 삭제할 파일을 인자로 지정하면 해당 파일이 삭제
- 바로 삭제되어 복구할 수도 없으므로 파일을 삭제할 때는 신중해야 함
- 예: data2 파일 삭제

```
user1@myubuntu:~/linux_ex/ch2$ ls
data1 data2 one temp
user1@myubuntu:~/linux_ex/ch2$ rm data2
user1@myubuntu:~/linux_ex/ch2$ ls
data1 one temp
user1@myubuntu:~/linux_ex/ch2$
```

03 파일 다루기

■ -i 옵션 사용하기

- -i 옵션을 지정하고 rm 명령을 사용하면 정말 삭제할 것인지 물어봄

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

■ 디렉터리 삭제하기

- rm 명령으로 디렉터를 지울 때는 -r 옵션을 지정(삭제된 디렉터리는 복구 불가능)
- 예: -r 옵션을 지정하지 않을 경우 오류 메시지 출력

```
user1@myubuntu:~/linux_ex/ch2$ cd temp
user1@myubuntu:~/linux_ex/ch2/temp$ ls
hosts services temp3 text1 text2
user1@myubuntu:~/linux_ex/ch2/temp$ rm temp3
rm: 'temp3'를 지울 수 없음: 디렉터리입니다
user1@myubuntu:~/linux_ex/ch2/temp$
```

03 파일 다루기

■ 디렉터리 삭제하기

- 예: rmdir 명령으로 temp3을 삭제하려고 하면 temp3 디렉터리가 비어 있지 않다고 오류 메시지 출력

```
user1@myubuntu:~/linux_ex/ch2/temp$ rmdir temp3
rmdir: failed to remove 'temp3': 디렉터리가 비어있지 않음
user1@myubuntu:~/linux_ex/ch2/temp$ ls
hosts  services  temp3  text1  text2
user1@myubuntu:~/linux_ex/ch2/temp$
```

- 예: -r 옵션 지정

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

03 파일 다루기

■ 디렉터리 삭제하기

- -i 옵션을 사용: 삭제하려는 디렉터리 아래에 있는 파일이나 서브 디렉터리를 삭제할 것인지 계속 물어봄

```
user1@myubuntu:~/linux_ex/ch2$ rm -ri temp
rm: 'temp' 디렉터리로 내려가겠습니까? y
rm: 일반 파일 'temp/text1'를 제거할까요? y
rm: 일반 파일 'temp/hosts'를 제거할까요? n
rm: 일반 파일 'temp/services'를 제거할까요? n
rm: 일반 파일 'temp/text2'를 제거할까요? n
rm: 디렉터리 'temp'를 제거할까요? n
user1@myubuntu:~/linux_ex/ch2$ ls
data1  one  temp
user1@myubuntu:~/linux_ex/ch2$ ls temp
hosts  services  text2
user1@myubuntu:~/linux_ex/ch2$
```

실습1

1. /etc/hosts 파일을test.org로 복사한다.
2. test 디렉토리를 만든다.
3. test.org 파일을test 디렉토리로 복사한다.
4. test 디렉토리에 있는test.org를 test.bak로 파일명을 바꾼다.
5. test.org 파일을 삭제한다.
6. test 디렉토리에 있는test.bak를 현재 디렉토리에test.org로 복사한다.
7. test 디렉토리를 삭제한다.

실습2

1. test.org파일을test.txt로 복사한다
2. backup 디렉터리를 생성한다
3. text.txt 파일을backup 디렉터리로 이동한다
4. backup 디렉터리의 이름을work로 바꾼다.
5. rmdir로 work 디렉터를 삭제해본다.어떤 현상이 왜 일어나는가?
6. rm 명령으로 디렉터를 삭제한다. 정말로 삭제할 것인지 물어보게 하려면 어떻게 해야 하는가? .

03 파일 다루기

■ 하드 링크 만들기 : ln

ln

기능 파일의 링크를 생성한다.

형식	ln [옵션]	원본 파일명	링크 파일명
----	-----------	--------	--------

옵션 -s : 심벌릭 링크 파일을 생성한다.

사용 예 ln test lntest ln -s test lntest

- 파일에 여러 개의 이름을 붙일 수 있는데, 이때 붙이는 파일명을 하드 링크라고 함
- 예: data1에 대한 하드 링크로 data1.ln 생성

```

user1@myubuntu:~/linux_ex/ch2$ ls -l
-rw-r--r-- 1 user1 user1 223 2월 22 12:36 data1
drwxrwxr-x 3 user1 user1 4096 2월 22 12:11 one
drwxrwxr-x 2 user1 user1 4096 2월 22 13:22 temp
-rw-r--r-- 1 user1 user1 223 2월 22 14:31 test.org
user1@myubuntu:~/linux_ex/ch2$ ln data1 data1.ln
user1@myubuntu:~/linux_ex/ch2$ ls -l
-rw-r--r-- 2 user1 user1 223 2월 22 12:36 data1
-rw-r--r-- 2 user1 user1 223 2월 22 12:36 data1.ln
drwxrwxr-x 3 user1 user1 4096 2월 22 12:11 one
drwxrwxr-x 2 user1 user1 4096 2월 22 13:22 temp
-rw-r--r-- 1 user1 user1 223 2월 22 14:31 test.org
user1@myubuntu:~/linux_ex/ch2$

```

달라진 것은 무엇인가?

03 파일 다루기

■ 하드 링크 만들기 : ln

- inode 비교: 다음 예에서 inode가 같은 파일은?

```
user1@myubuntu:~/linux_ex/ch2$ ls -i
1055628 data1 1055628 data1.ln 1055862 one 1055865 temp 1055864 test.org
user1@myubuntu:~/linux_ex/ch2$
```

- 파일의 내용도 동일

```
user1@myubuntu:~/linux_ex/ch2$ cat data1
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
(생략)
user1@myubuntu:~/linux_ex/ch2$ cat data1.ln
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
(생략)
user1@myubuntu:~/linux_ex/ch2$
```

03 파일 다루기

■ 하드 링크 만들기 : ln

- 링크와 복사의 차이: 무엇이 다른가?

```
user1@myubuntu:~/linux_ex/ch2$ cp data1 data1.cp
user1@myubuntu:~/linux_ex/ch2$ ls -i
1055628 data1      1055628 data1.ln   1055865 temp
1055940 data1.cp  1055862 one        1055864 test.org
user1@myubuntu:~/linux_ex/ch2$
```

- data1.cp 파일의 내용을 수정해도 data1 파일에는 반영안됨

03 파일 다루기

■ 심벌릭 링크 만들기 : -s 옵션

- 예: data1 파일의 심벌릭 링크로 data1.sl을 생성

```
user1@myubuntu:~/linux_ex/ch2$ ln -s data1 data1.sl
user1@myubuntu:~/linux_ex/ch2$ ls -l
1055628 data1      1055628 data1.ln  1055862 one      1055864 test.org
1055940 data1.cp  1055941 data1.sl  1055865 temp
user1@myubuntu:~/linux_ex/ch2$
```

- 심벌릭 링크의 inode 번호를 보면 원본 파일과 다른 번호
- ls -l 명령으로 확인해보면 파일의 종류가 'l(소문자 L)'로 표시되고 파일명도 '->'를 사용하여 원본 파일이 무엇인지를 알려줌

```
user1@myubuntu:~/linux_ex/ch2$ ls -l data1.sl
lrwxrwxrwx 1 user1 user1 5 2월 22 14:40 data1.sl -> data1
user1@myubuntu:~/linux_ex/ch2$
```

03 파일 다루기

■ 심벌릭 링크와 하드 링크의 차이

- 심벌릭 링크는 하드 링크와 비교하여 다음과 같은 몇 가지 특징이 있다.
 - 파일의 종류가 l(소문자 L)로 표시된다.
 - 하드 링크의 개수가 하나이다. 즉, 원본 파일에 이름을 추가하는 것이 아니다.
 - 파일 이름 뒤에 원본 파일의 이름이 표시된다(->data1).
 - inode 번호가 원본 파일과 다르다. 즉, 원본 파일과 심벌릭 링크 파일은 별개의 파일이다.
 - 디렉터리에 심벌릭 링크 생성 가능
 - 파일시스템이 달라도 심벌릭 링크 생성 가능
- 심벌릭 링크 파일의 내용은 원본 파일의 경로
- 심벌릭 링크에서는 원본 파일이 삭제되면 심벌릭 링크로 연결할 수 없다는 점을 주의해야 한다

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

실습

1. test.org파일의 하드 링크로test.ln을 만든다.
2. test.org 파일의 하드 링크로test.ln2를 만든다.
3. test.org와 test.ln, test.ln2의 inode 번호가 같다는 것을 확인한다.
4. test.org와 test.ln, test.ln2의 하드 링크 개수를 확인한다.
5. 디렉터리temp에 대한 심벌릭 링크로tmp를 만든다.
6. 디렉터리temp와 tmp의 내용이 같다는 것을 확인한다.
7. tmp 디렉터리와test.ln, test.ln2, test.org파일을 모두 삭제한다.

03 파일 다루기

■ 빈 파일 생성하기, 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

- 인자를 지정하지 않으면 빈 파일 생성

```
user1@myubuntu:~/linux_ex/ch2$ touch test
user1@myubuntu:~/linux_ex/ch2$ ls -l test
-rw-rw-r-- 1 user1 user1 0 2월 22 14:53 test
user1@myubuntu:~/linux_ex/ch2$
```