

# 오픈소스프로그래밍

Dong-Geol Choi

Information and Communication Engineering

Hanbat National University



# 파일 다루기



# 목차

1. 파일 내용 보기 : `cat/more/tail`
2. 파일 복사 : `cp`
3. 파일 삭제 : `rm`
4. 파일 이동 : `mv`
5. 파일 링크 : `ln`
6. 파일 생성 및 수정 시간 변경 : `touch`



# 학습목표

- 파일 관련 기본 명령의 사용 방법을 익힌다.
- 파일의 내용을 보고, 복사, 삭제, 이동, 링크하는 방법을 익힌다.
- 빈 파일을 생성하고 수정 시간을 변경하는 방법을 익힌다.



## 01. 파일 내용 보기 : cat/more/tail

### cat [ 옵션 ] 파일명

❖ 간단한 파일보기 명령으로 짧은 파일에 적합

❖ 옵션

- -n : 행 번호 출력

❖ 사용법

```
user1@Solaris11:~$ cat /etc/.login
```



## 01. 파일 내용 보기 : cat/more/tail

### more [ 옵션 ] 파일명

#### ❖ 파일의 내용을 한 화면씩 출력옵션

#### ❖ 옵션

- + 행 번호 : 지정한 행부터 출력

#### ❖ 명령

- 다음 페이지 : space
- 이전페이지 : b
- 종료 : q
- 문자열 찾기 : /문자열

#### ❖ 사용법

```
user1@solaris11:~$ more /etc/services
```

```
#
```



## 01. 파일 내용 보기 : cat/more/tail

### tail [ 옵션 ] 파일명

#### ❖ 파일의 마지막 부분 출력

#### ❖ 옵션

- + 행 번호 : 지정한 행부터 끝까지 출력
- - 숫자 : 화면에 출력할 행 수
- -f : 파일 출력이 종료되지 않고 주기적으로 반복 출력

#### ❖ 사용법

```
user1@Solaris11:~$ tail /etc/services
mdns                5353/tcp
vnc-server          5900/tcp            # VNC Server
dtspc               6112/tcp            # CDE subprocess control
servicetag          6481/udp
servicetag          6481/tcp
```



## 02. 파일 복사 : cp

**cp [ 옵션 ] 원본 파일명 복사 파일명**

❖ 특정 파일이나 디렉토리를 다른 이름으로 복사

❖ 옵션

- -i : 파일을 덮어쓸 때 경고 메시지 출력
- -r : 디렉토리 복사

❖ 사용 형태

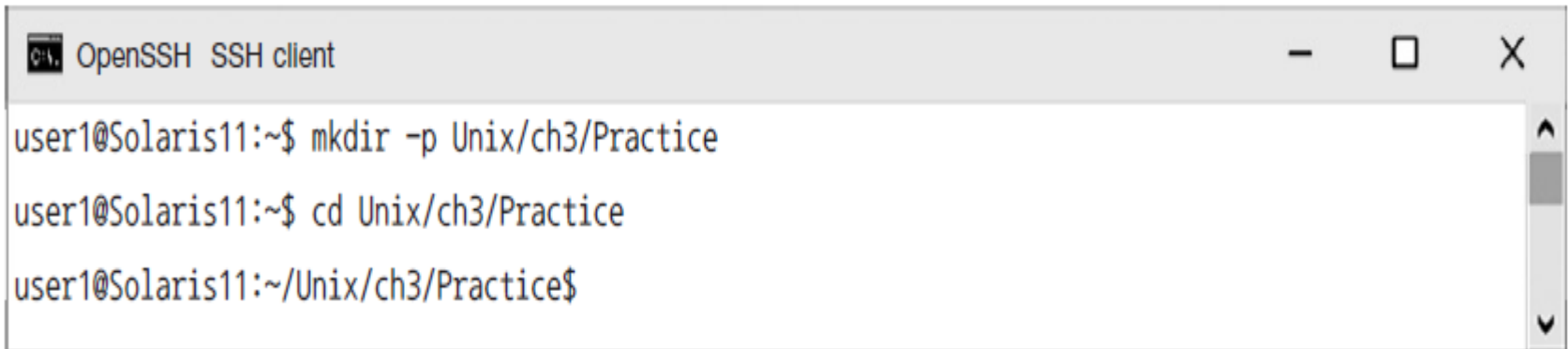
사용 형식	동 작
cp 파일1 파일2	파일1을 파일2로 복사.
cp 파일들 디렉토리	파일들을 디렉토리 밑에 같은 이름으로 복사.
cp -r 디렉토리1 디렉토리2	디렉토리1을 디렉토리2로 복사. 디렉토리1의 파일도 모두 복사됨.





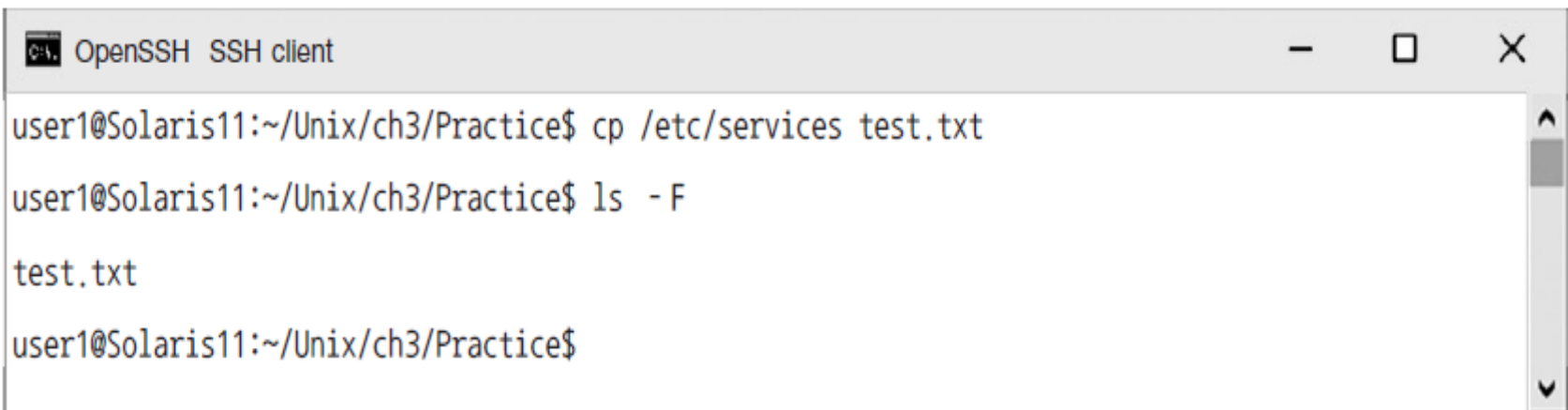
## [실습 2-1] 파일 내용 보기와 검색

### 1. 실습용 디렉토리 만들기

A terminal window titled "OpenSSH SSH client" with standard window controls. It shows a user at a Solaris11 prompt creating a directory and navigating into it.

```
user1@Solaris11:~$ mkdir -p Unix/ch3/Practice
user1@Solaris11:~$ cd Unix/ch3/Practice
user1@Solaris11:~/Unix/ch3/Practice$
```

### 2. 파일 복사하기

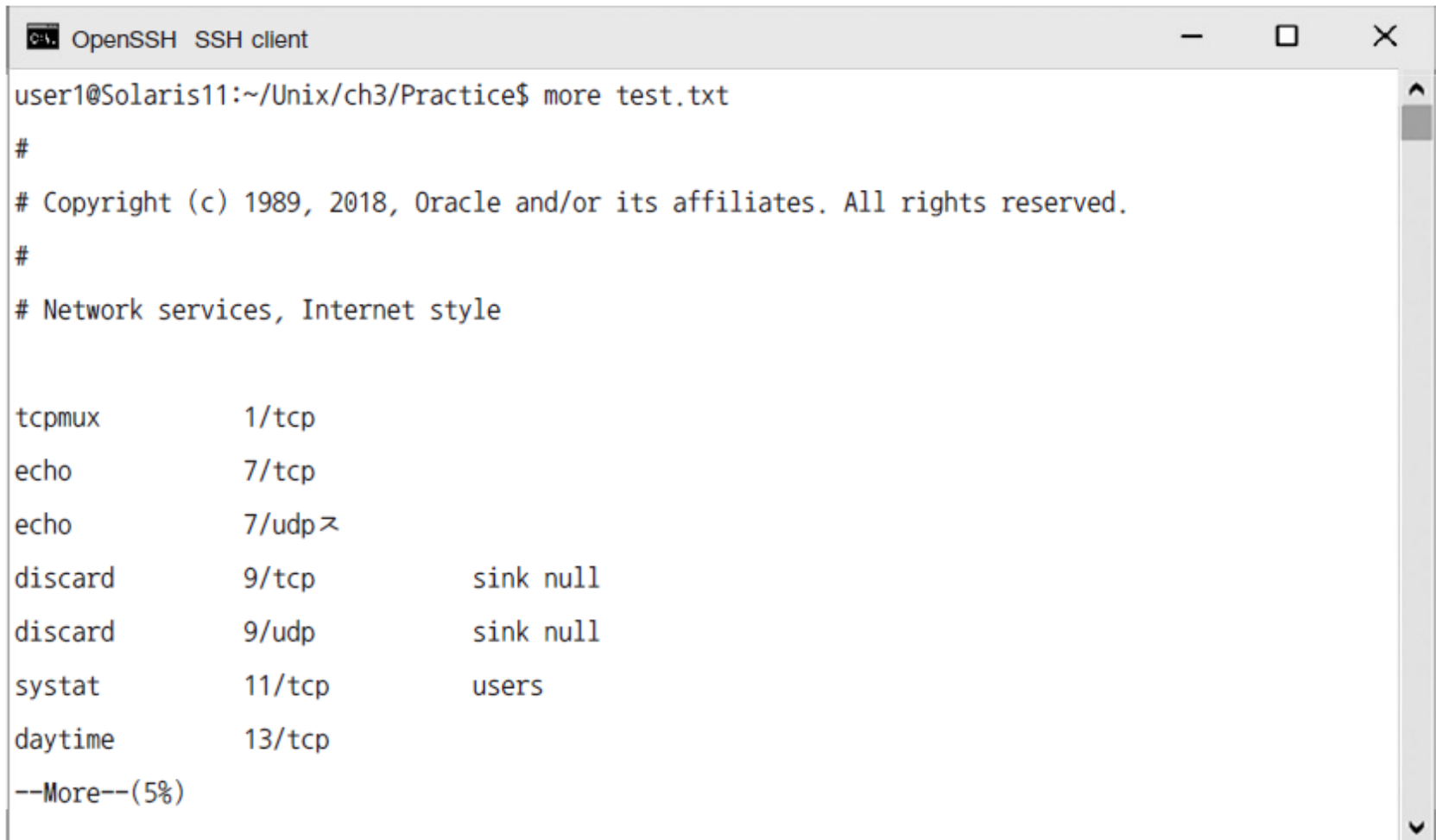
A terminal window titled "OpenSSH SSH client" with standard window controls. It shows a user at a Solaris11 prompt copying a file from /etc/services to the current directory and then listing the files.

```
user1@Solaris11:~/Unix/ch3/Practice$ cp /etc/services test.txt
user1@Solaris11:~/Unix/ch3/Practice$ ls -F
test.txt
user1@Solaris11:~/Unix/ch3/Practice$
```



## [실습 2-1] 파일 내용 보기와 검색

### 3. 파일 내용 보기



```
OpenSSH SSH client
user1@Solaris11:~/Unix/ch3/Practice$ more test.txt
#
# Copyright (c) 1989, 2018, Oracle and/or its affiliates. All rights reserved.
#
# Network services, Internet style

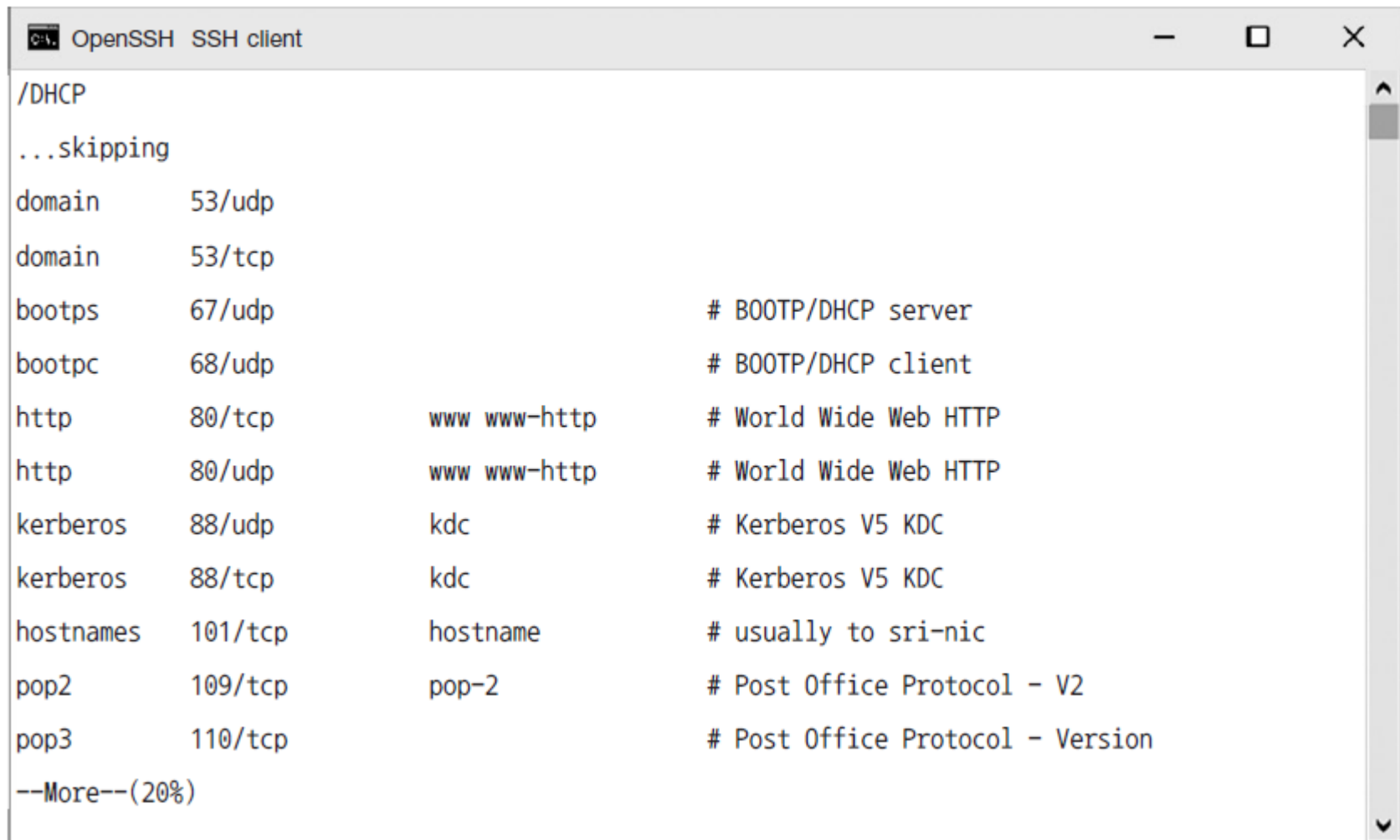
tcpmux      1/tcp
echo        7/tcp
echo        7/udp
discard     9/tcp      sink null
discard     9/udp      sink null
sysstat     11/tcp      users
daytime     13/tcp

--More--(5%)
```



## [실습 2-1] 파일 내용 보기와 검색

### 4. 파일 내용 검색하기

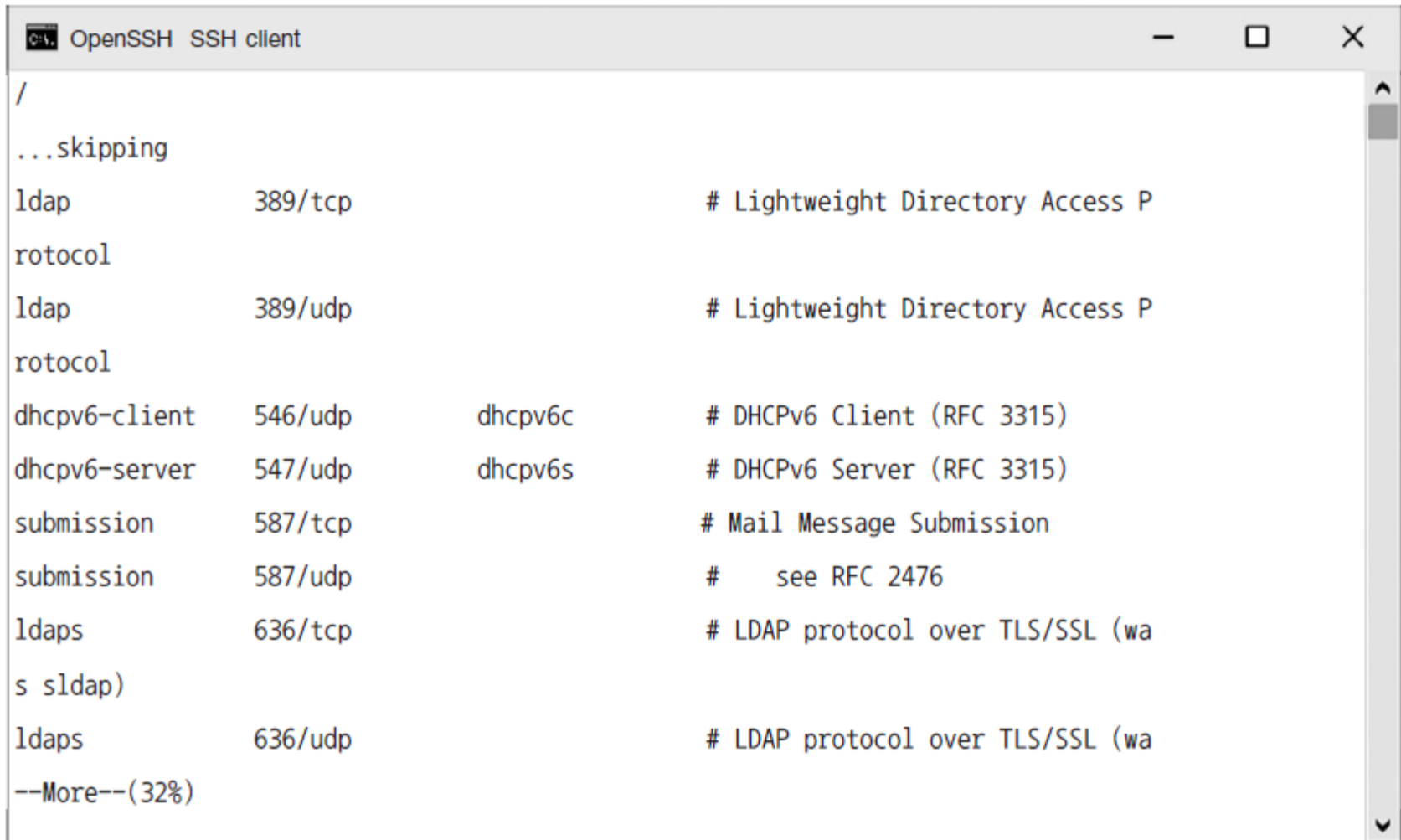


```
OpenSSH SSH client
/DHCP
...skipping
domain      53/udp
domain      53/tcp
bootps      67/udp      # BOOTP/DHCP server
bootpc      68/udp      # BOOTP/DHCP client
http        80/tcp      www www-http    # World Wide Web HTTP
http        80/udp      www www-http    # World Wide Web HTTP
kerberos    88/udp      kdc             # Kerberos V5 KDC
kerberos    88/tcp      kdc             # Kerberos V5 KDC
hostnames   101/tcp      hostname        # usually to sri-nic
pop2        109/tcp      pop-2           # Post Office Protocol - V2
pop3        110/tcp      # Post Office Protocol - Version
--More--(20%)
```



## [실습 2-1] 파일 내용 보기와 검색

### 5. 다음 검색하기



```
OpenSSH SSH client
/
...skipping
ldap          389/tcp          # Lightweight Directory Access P
rotocol
ldap          389/udp          # Lightweight Directory Access P
rotocol
dhcpv6-client 546/udp          dhcpv6c          # DHCPv6 Client (RFC 3315)
dhcpv6-server 547/udp          dhcpv6s          # DHCPv6 Server (RFC 3315)
submission    587/tcp          # Mail Message Submission
submission    587/udp          #   see RFC 2476
ldaps         636/tcp          # LDAP protocol over TLS/SSL (wa
s sldap)
ldaps         636/udp          # LDAP protocol over TLS/SSL (wa
--More--(32%)
```



## [실습 2-1] 파일 내용 보기와 검색

### 6. 내용 보기 종료하기

```
OpenSSH SSH client
/
...skipping
ldap      3      89/tcp      # Lightweight Directory Access P
rotocol
ldap      389/udp    # Lightweight Directory Access P
rotocol
dhcpv6-client  546/udp      dhcpv6c      # DHCPv6 Client (RFC 3315)
dhcpv6-server  547/udp      dhcpv6s      # DHCPv6 Server (RFC 3315)
submission  587/tcp      # Mail Message Submission
submission  587/udp      # see RFC 2476
ldaps      636/tcp      # LDAP protocol over TLS/SSL (wa
s sldap)
ldaps      636/udp      # LDAP protocol over TLS/SSL (wa
user1@Solaris11:~/Unix/ch3/Practice$
```

### 7. 디렉토리 위치 변경하기

```
OpenSSH SSH client
user1@Solaris11:~/Unix/ch3/Practice$ cd ..
user1@Solaris11:~/Unix/ch3$
```



## 03. 파일 삭제 : rm

**rm [ 옵션 ] 파일명/디렉토리명**

❖ 지정한 파일이나 디렉토리를 삭제

❖ 옵션

- -i: 확인 후 삭제
- -f: 무조건 삭제
- -r: 디렉토리 삭제

**주의 !!**  
**한번 삭제한 파일은**  
**되살릴 수 없음 !!**



## 04. 파일 이동 : mv

**mv [ 옵션 ] 원래이름 새이름**

❖ 파일 이름을 바꾸거나 이동

❖ 옵션

- -i : 확인 후 이동
- -f : 무조건 삭제

❖ 사용 형태

사용 형식	동 작
mv 파일1 파일2	파일1의 이름을 파일2로 바꿈.
mv 파일명... 디렉토리	지정한 파일들을 디렉토리로 옮김.
mv 디렉토리1 디렉토리2	디렉토리1의 이름을 디렉토리2로 바꿈.



## [실습 2-2] 이동/삭제하기

### 1. 현재 위치 이동하기

```
OpenSSH SSH client
user1@Solaris11:~/Unix/ch3$ cd Practice
user1@Solaris11:~/Unix/ch3/Practice$
```

### 2. 디렉토리 만들기

```
OpenSSH SSH client
user1@Solaris11:~/Unix/ch3/Practice$ mkdir Temp
user1@Solaris11:~/Unix/ch3/Practice$ ls -F
Temp/      test.txt
user1@Solaris11:~/Unix/ch3/Practice$
```

### 3. 파일 이동하기

```
OpenSSH SSH client
user1@Solaris11:~/Unix/ch3/Practice$ mv test.txt Temp
user1@Solaris11:~/Unix/ch3/Practice$ ls
Temp
user1@Solaris11:~/Unix/ch3/Practice$ ls -l Temp
total 11
-rw-r--r--  1 user1  staff    4739 Jun 17 20:53 test.txt
user1@Solaris11:~/Unix/ch3/Practice$
```





## [실습 2-2] 이동/삭제하기

### 4. 디렉토리 이름 바꾸기

```
OpenSSH SSH client
user1@Solaris11:~/Unix/ch3/Practice$ mv Temp mvTest
user1@Solaris11:~/Unix/ch3/Practice$ ls -ld mvTest
drwxr-xr-x  2 user1  staff          3 Jun 17 21:59 mvTest
user1@Solaris11:~/Unix/ch3/Practice$
```

### 5. 디렉토리 삭제하기

```
OpenSSH SSH client
user1@Solaris11:~/Unix/ch3/Practice$ rm -r mvTest
user1@Solaris11:~/Unix/ch3/Practice$ ls
user1@Solaris11:~/Unix/ch3/Practice$
```

### 6. 디렉토리 위치 변경하기

```
OpenSSH SSH client
user1@Solaris11:~/Unix/ch3/Practice$ cd ..
user1@Solaris11:~/Unix/ch3$
```



## 05. 파일 링크 : ln

### ln [ 옵션 ] 원래이름 새이름

- ❖ 기존의 파일과 연결된 링크파일 생성
- ❖ 경로를 다르게 지정할 수 있음
- ❖ 옵션
  - -s : 심볼릭 링크 파일 생성 (기본 설정은 하드링크)
- ❖ 사용법

```
user1@Solaris11:~/Unix/ch3$ cat > ln.txt
I love UNIX!
user1@Solaris11:~/Unix/ch3$ ls -l ln.txt
-rw-r--r--  1 user1  staff          13
user1@Solaris11:~/Unix/ch3$
```

```
user1@Solaris11:~/Unix/ch3$ ln ln.txt ln.hl
user1@Solaris11:~/Unix/ch3$ ls -l ln*
-rw-r--r--  2 user1  staff          13 Jun 17 22:02 ln.txt
-rw-r--r--  2 user1  staff          13 Jun 17 22:02 ln.hl
user1@Solaris11:~/Unix/ch3$
```



## 05. 파일 링크 : ln

### ❖ 복사(cp)와 링크(ln)의 차이는?

cp	ln
완전 별도 파일 생성	이름만 다르고 내용은 동일
둘 중 하나를 수정해도 다른 파일에 영향 없음	둘 중 하나를 수정하면 두 파일이 같이 수정
같은 파일을 별도로 수정하여 작업할 때 사용	파일을 공동으로 관리해야 할 때 사용



## 06. 파일 생성 및 수정시간 변경 : touch

### touch [ 옵션 ] 파일

❖ 지정한 파일이 존재하지 않으면 생성, 존재하면 현재 시간 또는 지정한 시간으로 변경

#### ❖ 옵션

- -a : 접근 시간만 변경
- -c : 지정한 파일이 없는 경우 새로 생성하지 않음
- -m : 수정 시간만 변경
- -r 파일1 파일2 : 파일2의 시간을 파일1에서 읽음
- -t [[CC]YY]MMDDhhmm[.ss] : 시간을 직접 입력

Ex) 2020년 02월 11일 10시 20분 35초

CCYY MM DD hh mm ss



## 06. 파일 생성 및 수정시간 변경 : touch

### ❖ 사용법

```
user1@Solaris11:~/Unix/ch3$ mkdir Touch
user1@Solaris11:~/Unix/ch3$ cd Touch
user1@Solaris11:~/Unix/ch3/Touch$ ls test           ➡ 파일이 있는지 검사
test: No such file or directory                     ➡ 파일이 없음
user1@Solaris11:~/Unix/ch3/Touch$
```

```
user1@Solaris11:~/Unix/ch3/Touch$ touch test
user1@Solaris11:~/Unix/ch3/Touch$ ls -l test
-rw-r--r--  1 user1  staff      0 Jun 17 22:47 test
user1@Solaris11:~/Unix/ch3/Touch$
```



# vim 사용법 익히기



# 목차

1. 유닉스 편집기
2. vim의 사용 방법
3. vim의 환경 설정



# 학습목표

- 유닉스에서 사용하는 편집기의 종류를 알아본다.
- 대표적인 화면 편집기인 vim의 사용 방법을 익힌다.
- vim의 환경 설정 방법을 익힌다.





# 01. 유닉스 편집기

## ❖ 유닉스 편집기의 종류

- 행 단위 편집기

ed

유닉스 초기에 사용 → 지금은 거의 사용하지 않음

ex

단독으로 사용하기 보다는 vi와 연결하여 사용

- 화면 단위 편집기

vim

vi와 유사한 화면 편집기이다.

솔라리스는 vi 명령을 사용하면 vim을 실행한다.

설치 명령어 : `sudo apt install vim`

Emacs

막강한 기능 제공 → 설치해서 사용해야 함



# 01. 유닉스 편집기

## ❖ 모드형과 비모드형

- 모드형

명령모드에서 입력한 키는 명령으로 간주

입력모드에서 입력한 키는 데이터로 간주

모드를 바꾸기 위한 특수 키가 있음 (예: Esc)

vim은 모드형

- 비모드형

입력한 모든 키는 데이터로 간주됨

명령은 특수키(Ctrl, Alt 등)와 일반키의 조합으로 구성

예 : 아래 한글, 워드



## 02. vim의 사용 방법

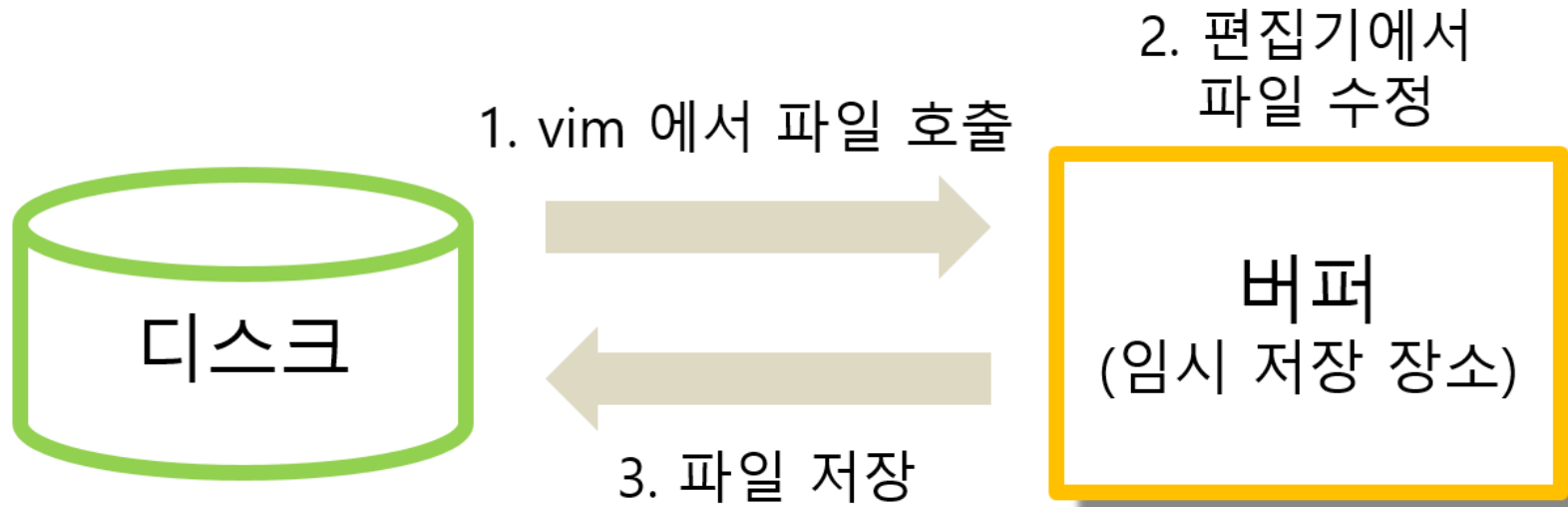


그림 4-1 vim 의 동작 구조

- 파일 내용을 잘못 수정한 경우 버퍼의 내용 무시
- 작업 취소기능 제공



## 02. vim의 사용 방법

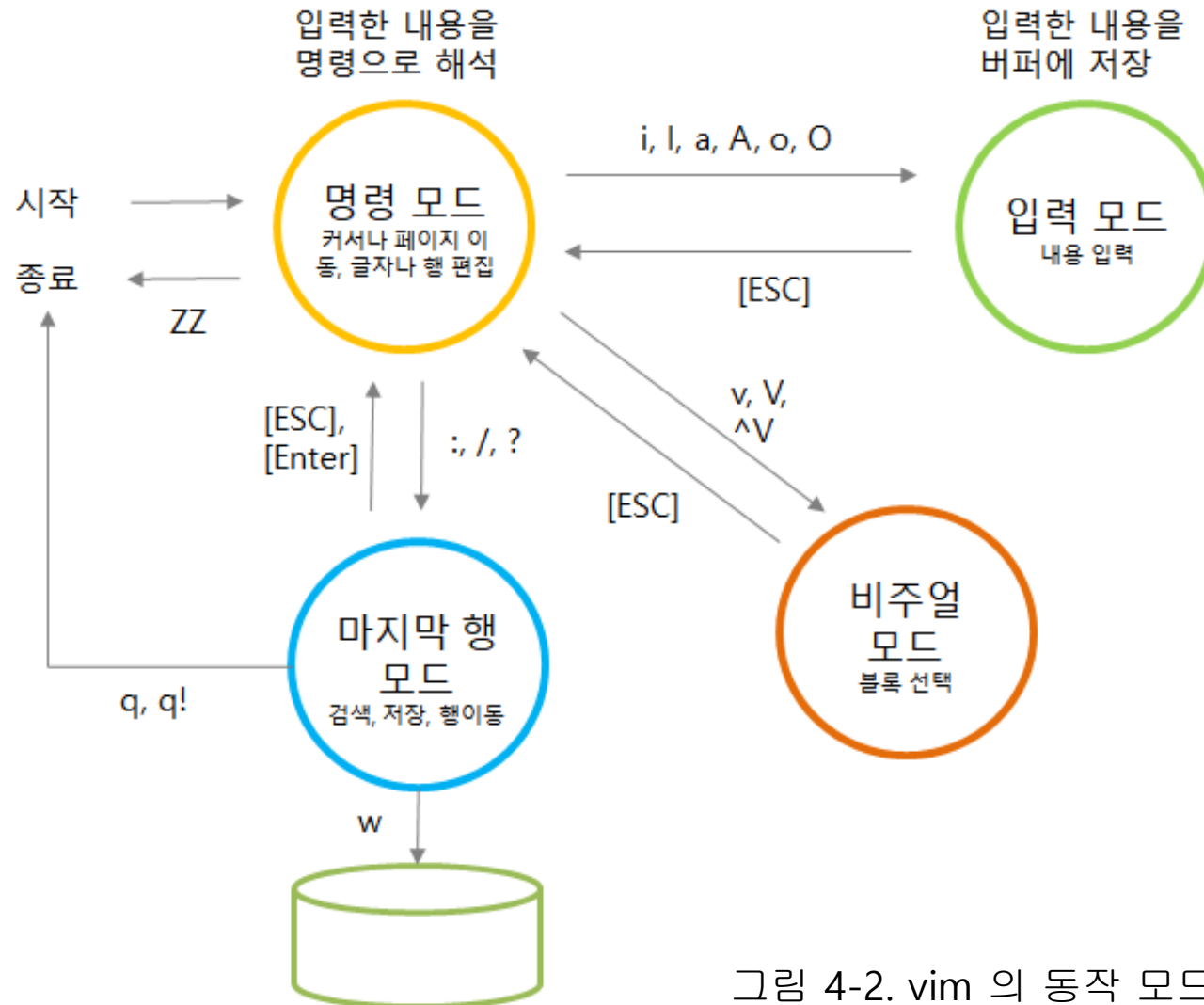


그림 4-2. vim 의 동작 모드



## 02. vim의 사용 방법

### ❖ vim

- 새로운 파일 시작
- 파일 저장할 때 이름 지정  
유닉스 초기에 사용 → 지금은 거의 사용하지 않음
- vim → 빈 파일이 열림

### ❖ vim 파일이름

- 지정한 이름이 없으면 새로운 파일 생성
- 지정한 이름이 있으면 기존 파일 열기
- vim first → first라는 파일이 열림





## 02. vim의 사용 방법

### ❖ vim 종료와 저장

표 4-2 vim 종료와 저장 명령키

명령키	기능
:q	vim에서 작업한 것이 없을 때 그냥 종료한다.
:q!	작업한 내용을 저장하지 않고 종료한다.
:w [ 파일명 ]	작업한 내용을 저장만 한다. 파일명을 지정하면 새 파일로 저장한다.
:wq, :wq!	작업한 내용을 저장하고 vim을 종료한다.
ZZ(Shift+zz)	작업한 내용을 저장하고 vim을 종료한다.



## 02. vim의 사용 방법

### ❖ 입력 모드로의 전환

표 4-3 vim 입력 모드 전환 명령키

명령키	기능
i	커서 앞에 입력한다(현재 커서 자리에 입력).
a	커서 뒤에 입력한다(현재 커서 다음 자리에 입력).
o	커서가 위치한 행의 다음 행에 입력한다.
I	커서가 위치한 행의 첫 컬럼으로 이동해 입력한다.
A	커서가 위치한 행의 마지막 컬럼으로 이동해 입력한다.
O	커서가 위치한 행의 이전 행에 입력한다.

- Esc 키를 누르면 다시 명령 모드가 된다.

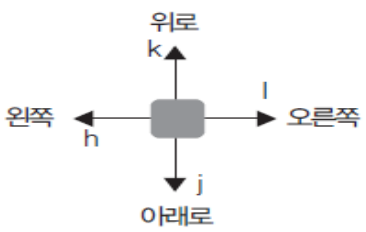





## 02. vim의 사용 방법

### ❖ 커서 이동

표 4-4 vim 커서 이동 명령키

명령키	기능	
k	커서를 한 행 위로 이동시킨다.	
j	커서를 한 행 아래로 이동시킨다.	
l	커서를 한 문자 오른쪽으로 이동시킨다.	
h	커서를 한 문자 왼쪽으로 이동시킨다.	
^ 또는 0	커서를 현재 행의 처음으로 이동시킨다.	
\$	커서를 현재 행의 마지막으로 이동시킨다.	
-	커서를 이전 행의 처음으로 이동시킨다.	
+ 또는 	커서를 다음 행의 처음으로 이동시킨다.	
H	커서를 화면의 맨 위 행으로 이동시킨다.	
M	커서를 화면의 중간 행으로 이동시킨다.	
L	커서를 화면의 맨 아래 행으로 이동시킨다.	
w	커서를 다음 단어의 첫 글자 위치로 이동시킨다.	
b	커서를 앞 단어의 첫 글자 위치로 이동시킨다.	
e	커서를 다음 단어의 마지막 글자 위치로 이동시킨다.	



## 02. vim의 사용 방법

### ❖ 화면 이동

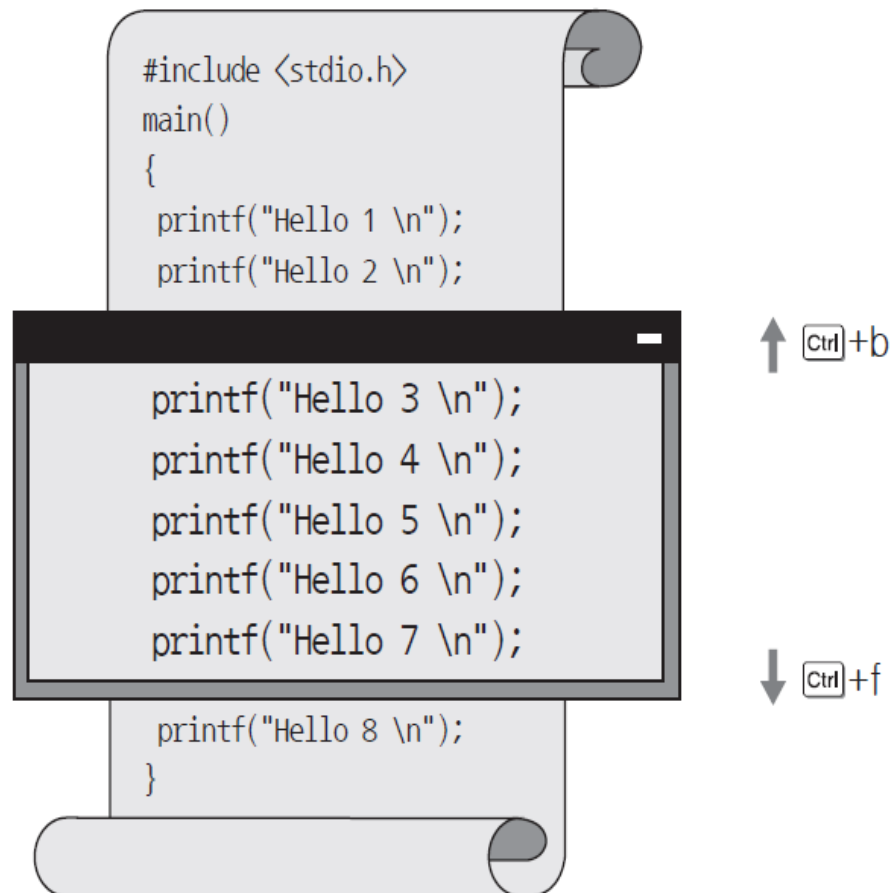


그림 4-5 화면 이동 명령키의 사용

표 4-5 vim 화면 이동 명령키

명령키	기능
<b>Ctrl</b> +u	반 화면 위로 이동시킨다.
<b>Ctrl</b> +d	반 화면 아래로 이동시킨다.
<b>Ctrl</b> +b	한 화면 위로 이동시킨다.
<b>Ctrl</b> +f	한 화면 아래로 이동시킨다.
<b>Ctrl</b> +y	화면을 한 행만 위로 이동시킨다.
<b>Ctrl</b> +e	화면을 한 행만 아래로 이동시킨다.



## 02. vim의 사용 방법

### ❖ 화면 이동

표 4-6 vim 지정한 행으로 이동 명령키



명령키	기능
G( <span>Shift</span> +g)	마지막 행으로 이동한다.
행번호G( <span>Shift</span> +g)	지정한 행으로 이동한다.
:행번호	지정한 행으로 이동한다.
:\$	마지막 행으로 이동한다.



## 02. vim의 사용 방법

### ❖ 수정 및 삭제

표 4-7 vim 내용 수정 명령키

명령키	기능
r	커서가 위치한 문자를 다른 문자로 수정한다.
cw, #cw	커서의 위치부터 현재 단어의 끝까지를 새로 입력한 내용으로 수정한다.  키를 입력하면 수정이 종료된다(#에는 수정할 단어의 수를 지정, 예를 들어 3cw는 커서 위치부터 3단어 수정).
s, #s	커서가 위치한 문자를  키를 입력할 때까지 입력한 내용으로 수정한다(#에는 수정할 문자의 수를 지정, 예를 들어 5s는 커서 위치부터 5글자를 수정).
cc	커서가 위치한 행의 내용을 모두 수정한다.
C	커서의 위치부터 행의 끝까지 수정한다.



## 02. vim의 사용 방법

### ❖ 수정 및 삭제

표 4-8 vim 내용 삭제 및 취소 명령키

명령키	기능
x, #x	커서가 위치한 문자를 삭제한다(#은 삭제할 문자 수, 예를 들어 3x는 3글자 삭제).
dw, #dw	커서 위치의 단어를 삭제한다(#은 삭제할 단어 수).
dd, #dd	커서가 위치한 행을 삭제한다(#은 삭제할 행의 수, 예를 들어 5dd는 커서 위치부터 5행을 삭제).
u	방금 수행한 명령을 취소한다.
U	해당 행에서 한 모든 명령을 취소한다.
:e!	마지막으로 저장한 내용 이후의 것을 버리고 새로 작업한다.



# [실습 3-1] vim 입력과 수정 방법 익히기

## 1. 실습용 디렉토리 만들기

```
OpenSSH SSH client
user1@Solaris11:~/Unix/ch4$ mkdir Practice
user1@Solaris11:~/Unix/ch4$ cd Practice
user1@Solaris11:~/Unix/ch4/Practice$
```

## 2. 파일 생성하기

```
OpenSSH SSH client
user1@Solaris11:~/Unix/ch4$ vi test.txt
```

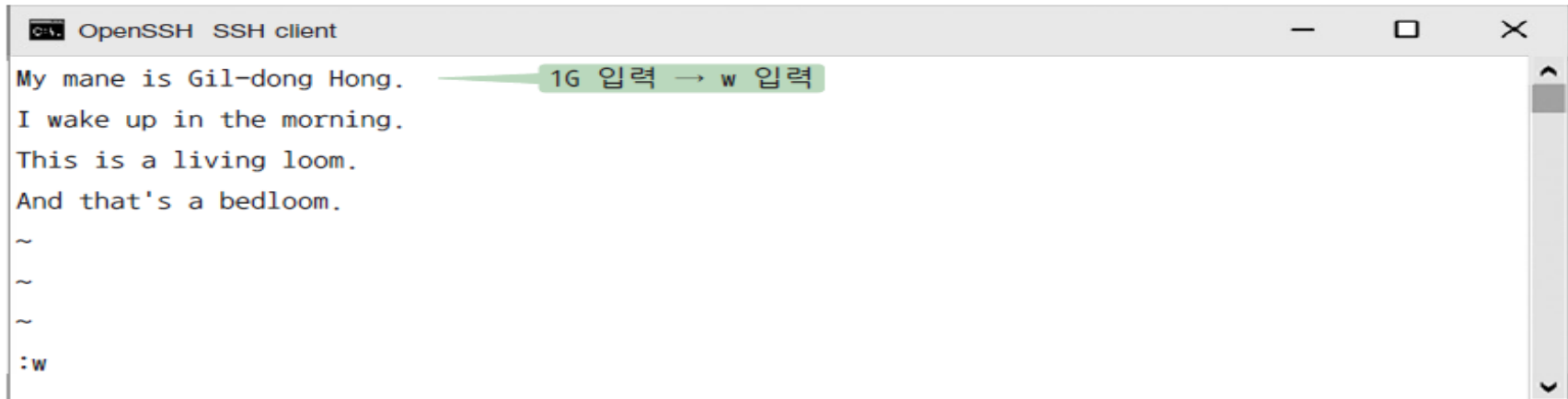
```
OpenSSH SSH client
My mane is Gil-dong Hong.
I wake up in the morning.
This is a living loom.
And that's a bedroom.
~
~
~
:w
```

i 입력 → 내용 입력 → Esc 키 입력 → :w 입력



## [실습 3-1] vim 입력과 수정 방법 익히기

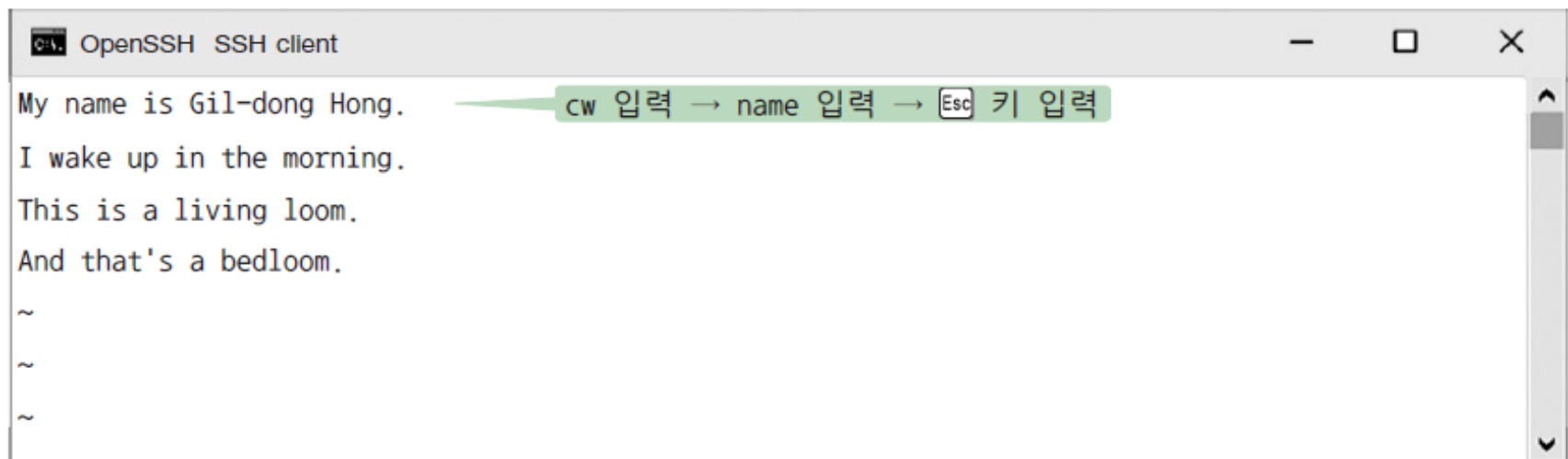
### 3. 커서 이동하기



```
OpenSSH SSH client
My mane is Gil-dong Hong.
I wake up in the morning.
This is a living loom.
And that's a bedroom.
~
~
~
:w
```

1G 입력 → w 입력

### 4. 단어 수정하기



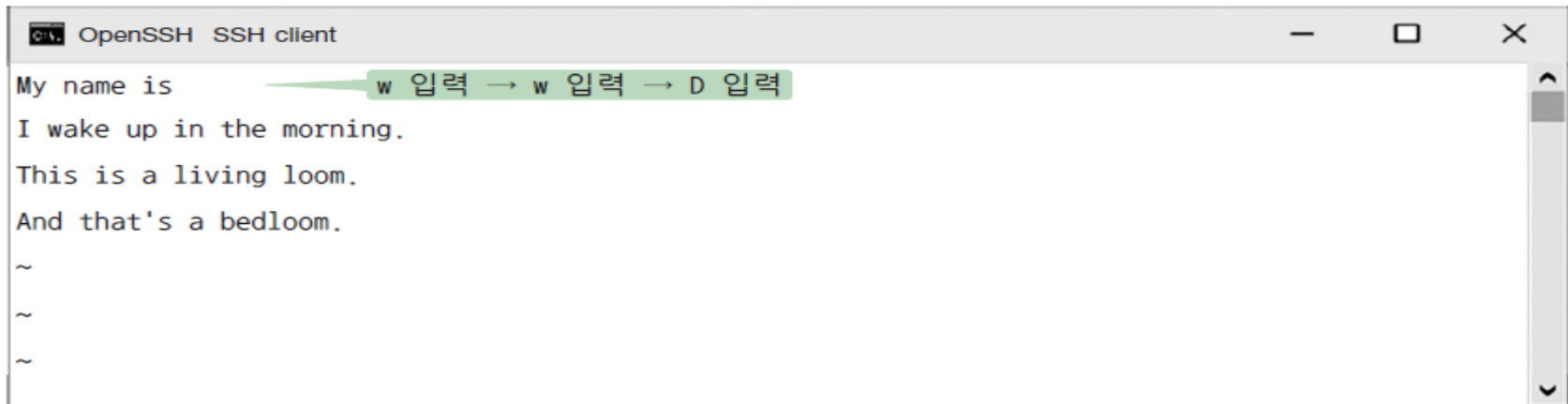
```
OpenSSH SSH client
My name is Gil-dong Hong.
I wake up in the morning.
This is a living loom.
And that's a bedroom.
~
~
~
```

cw 입력 → name 입력 → Esc 키 입력



## [실습 3-1] vim 입력과 수정 방법 익히기

### 5. 이름 삭제하기



```
OpenSSH SSH client
My name is
I wake up in the morning.
This is a living loom.
And that's a bedroom.
~
~
~
```

### 6. 이전 동작 취소하고 복구하기



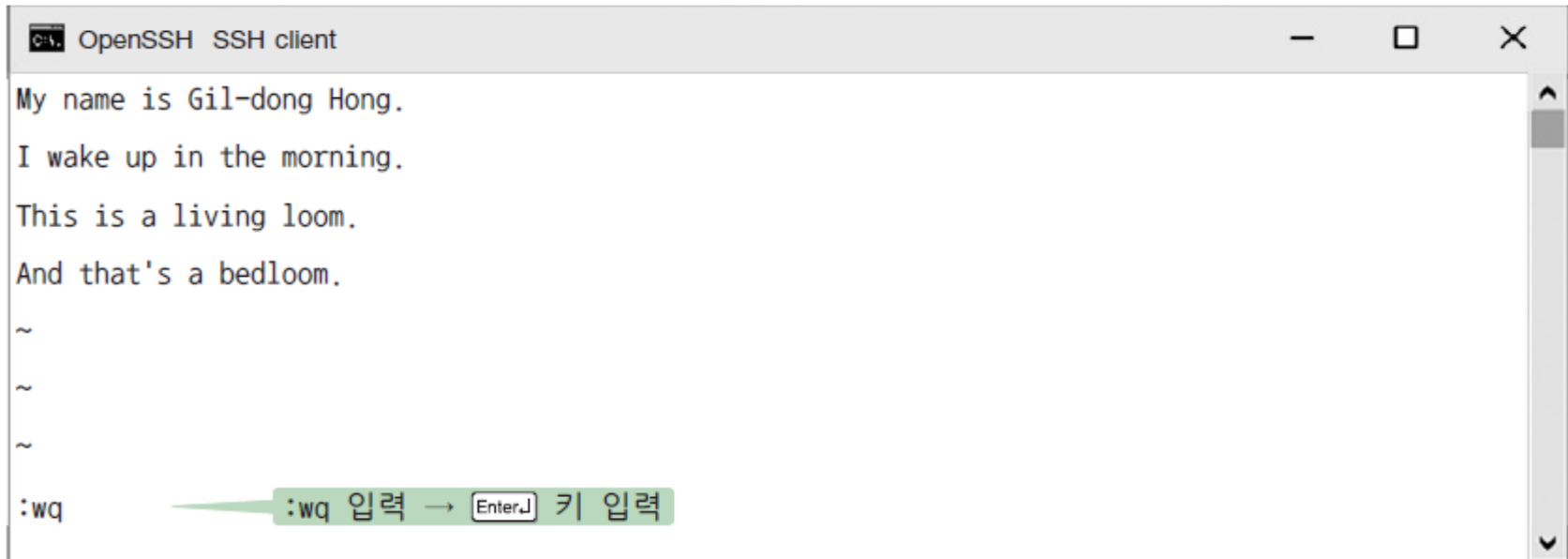
```
OpenSSH SSH client
My name is Gil-dong Hong.
I wake up in the morning.
This is a living loom.
And that's a bedroom.
~
~
~
```





## [실습 3-1] vim 입력과 수정 방법 익히기

### 7. 저장하고 종료하기



```
OpenSSH SSH client
My name is Gil-dong Hong.
I wake up in the morning.
This is a living loom.
And that's a bedroom.
~
~
~
:wq
```

:wq 입력 → [Enter] 키 입력

### 8. 디렉토리 위치 변경하기



```
OpenSSH SSH client
user1@Solaris11:~/Unix/ch4/Practice$ cd ..
user1@Solaris11:~/Unix/ch4$
```



## 02. vim의 사용 방법

### ❖ 편집 기능

- 복사하기 / 붙이기 / 잘라내기 명령

표 4-9 vim 복사하기와 붙이기 명령키

명령키	기능
yy, #yy	커서가 위치한 행을 복사한다(#에는 복사할 행의 수 지정, 예를 들어 3yy하면 3행 복사)
p(소문자)	커서가 위치한 행의 아래쪽에 붙인다.
P(대문자)	커서의 위치한 행의 위쪽에 붙인다.
dd, #dd	커서가 위치한 행을 잘라둔다(#에는 잘라줄 행의 수 지정, 예를 들어 3dd 명령을 입력하면 3행 잘라 두기(삭제)).



## 02. vim의 사용 방법

### ❖ 편집 기능

- 버퍼의 사용

버퍼

vim은 작업 내용을 버퍼에 저장 → 실행 취소 가능  
복사하기, 잘라내기 사용

버퍼 종류

Unnamed buffer : 이름 없는 버퍼

Named buffers : 이름이 있는 버퍼

Numbered buffers : 번호가 있는 버퍼

사용 예

“a3yy → 현재 행부터 아래로 3줄을 a버퍼에 저장

“ap → a버퍼의 내용을 붙이기



## 02. vim의 사용 방법

### ❖ 편집 기능

- 마지막 행 모드에서 복사하기와 잘라내기 범위 지정하기

표 4-10 vim 범위 지정 명령키

명령키	기능
1, \$ 또는 %	1행부터 마지막 행까지 지정한다.
1, .	1행부터 커서가 있는 행까지 지정한다.
., \$	커서가 있는 행부터 마지막 행까지 지정한다.
., -3	현재 행부터 앞으로 세 번째 행까지 지정한다.
10, 20	10행부터 20행까지 지정한다.



## 02. vim의 사용 방법

### ❖ 편집 기능

- 마지막 행 모드에서 복사하기와 잘라내기  
복사하기 / 잘라내기

표 4-11 vim 마지막 행 모드에서의 이동/복사/삭제 명령키

명령키	기능
:#y	#으로 지정한 행을 복사한다(예 :10y → 10행을 복사).
:<범위>y	범위로 지정한 행을 복사한다(예 :10,20y → 10행~20행까지 복사).
:#d	#으로 지정한 행을 삭제한다(예 :10d → 10행 삭제).
:<범위>d	범위로 지정한 행을 삭제한다(예 :10,20d → 10행~20행까지 삭제).
:pu	현재 행 다음에 버퍼의 내용을 붙인다.
:#pu	#으로 지정한 행 다음에 버퍼의 내용을 붙인다(예 :5pu).



## 02. vim의 사용 방법

### ❖ 편집 기능

- 검색하기 / 바꾸기

검색하기

검색을 위해 마지막 행 모드 이동 → /나? 입력

표 4-12 vim 검색 명령키

명령키	기능
/문자열	문자열을 아래 방향으로 검색한다.
?문자열	문자열을 위 방향으로 검색한다.
n	원래 찾는 방향으로 다음 문자열을 찾는다.
N	역방향으로 다음 문자열을 찾는다.



## 02. vim의 사용 방법

### ❖ 편집 기능

#### ■ 검색하기 / 바꾸기

##### 바꾸기

기존의 문자열을 다른 문자열로 바꾸려면 먼저 :를 입력  
→ 마지막 행 모드로 이동

파일 전체 대상 또는 특정 범위만 지정 실행 가능

표 4-13 vim 바꾸기 명령키

명령키	기능
:s/문자열1/문자열2/	커서가 위치한 행의 첫 번째 문자열1을 문자열2로 바꾼다.
:%s/문자열1/문자열2/g	파일 전체에서 모든 문자열1을 문자열2로 바꾼다.
:<범위>s/문자열1/문자열2/	<범위> 안의 모든 행에 대해서 각 행의 첫 번째 문자열을 찾아 문자열2로 바꾼다.
:<범위>s/문자열1/문자열2/g	<범위> 안의 모든 행에서 모든 문자열1을 문자열2로 바꾼다.
:<범위>s/문자열1/문자열2/gc	<범위> 안의 모든 행에 대해서 각 문자열1을 문자열2로 바꿀 때 수정할지 여부를 묻는다.



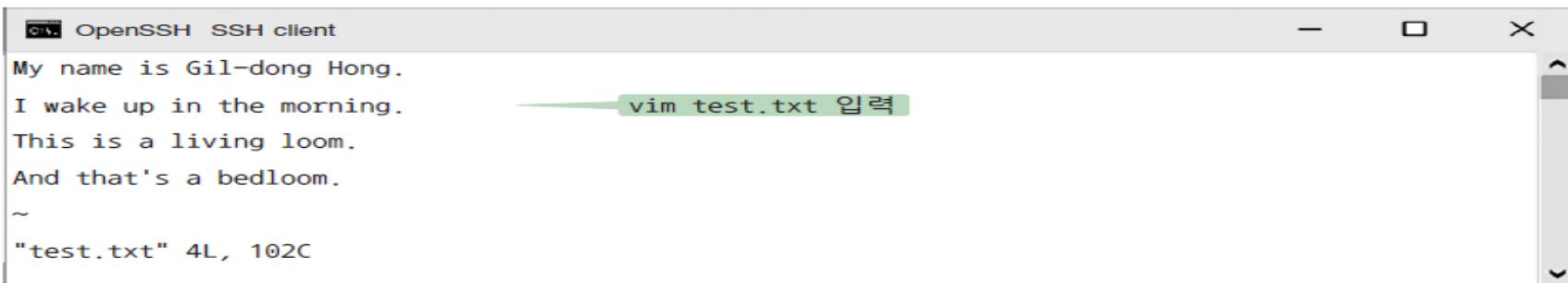
## [실습 3-2] vim 편집 방법 익히기

### 1. 현재 위치 이동하기



```
OpenSSH SSH client
user1@Solaris11:~/Unix/ch4$ cd Practice
user1@Solaris11:~/Unix/ch4/Practice$
```

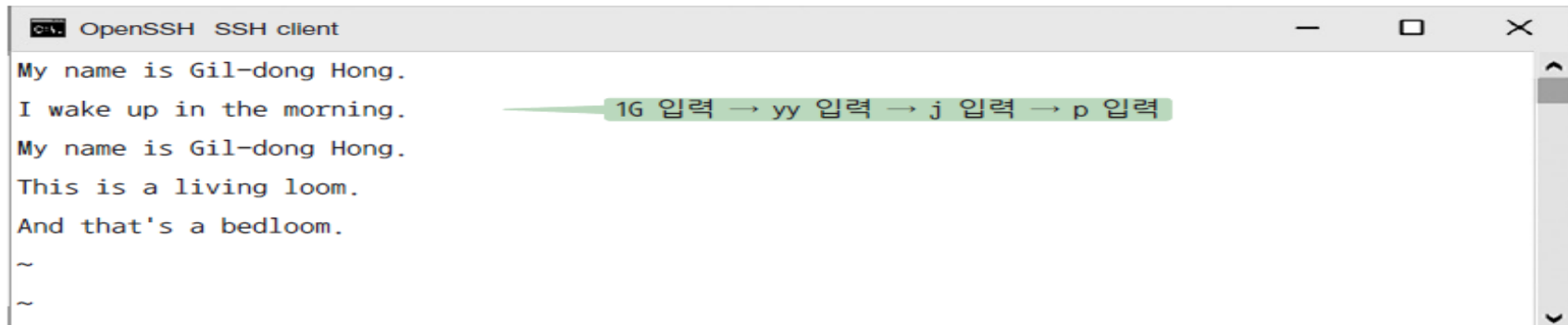
### 2. 실습용 예제 파일 열기



```
OpenSSH SSH client
My name is Gil-dong Hong.
I wake up in the morning.
This is a living loom.
And that's a bedroom.
~
"test.txt" 4L, 102C
```

vim test.txt 입력

### 3. 복사해 붙이기



```
OpenSSH SSH client
My name is Gil-dong Hong.
I wake up in the morning.
My name is Gil-dong Hong.
This is a living loom.
And that's a bedroom.
~
~
```

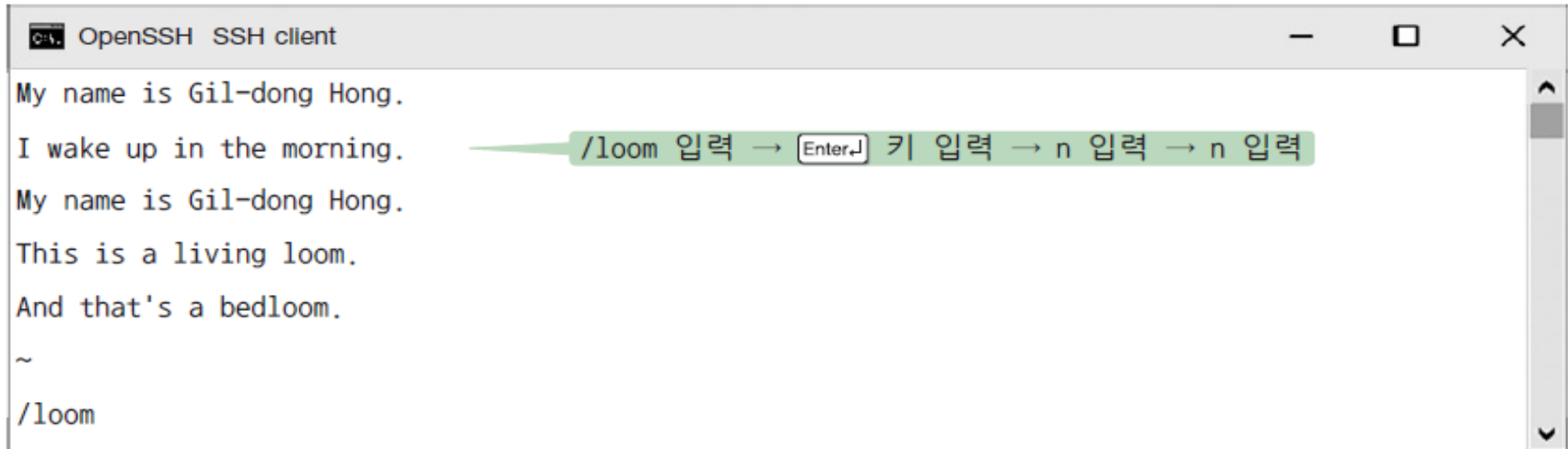
1G 입력 → yy 입력 → j 입력 → p 입력





## [실습 3-2] vim 편집 방법 익히기

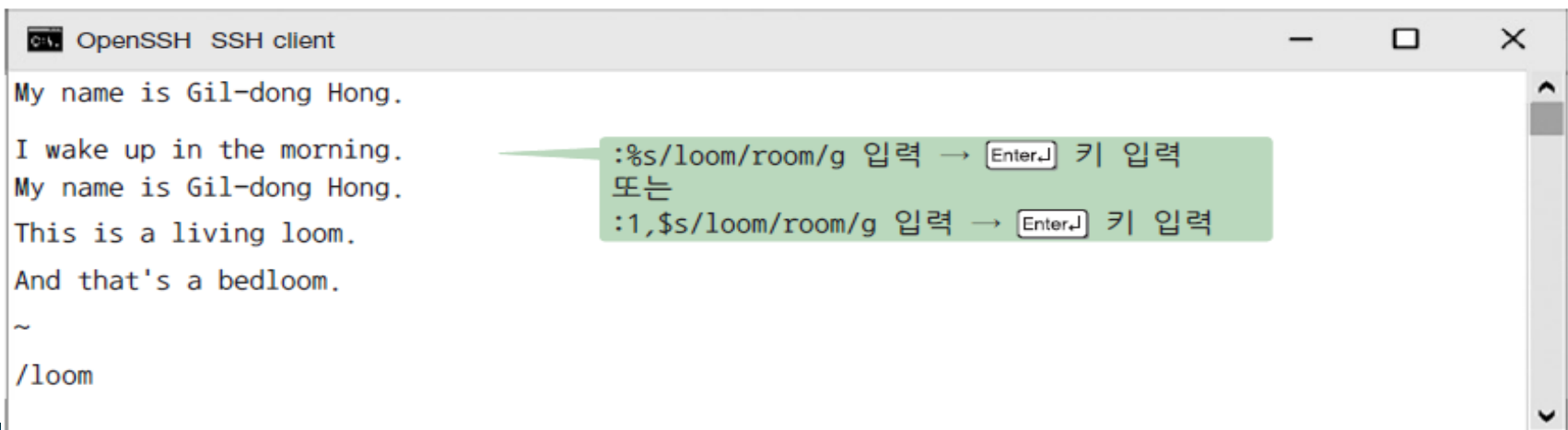
### 4. 단어 검색하기



```
OpenSSH SSH client
My name is Gil-dong Hong.
I wake up in the morning.
My name is Gil-dong Hong.
This is a living loom.
And that's a bedroom.
~
/loom
```

/loom 입력 → Enter 키 입력 → n 입력 → n 입력

### 5. 단어 바꾸기



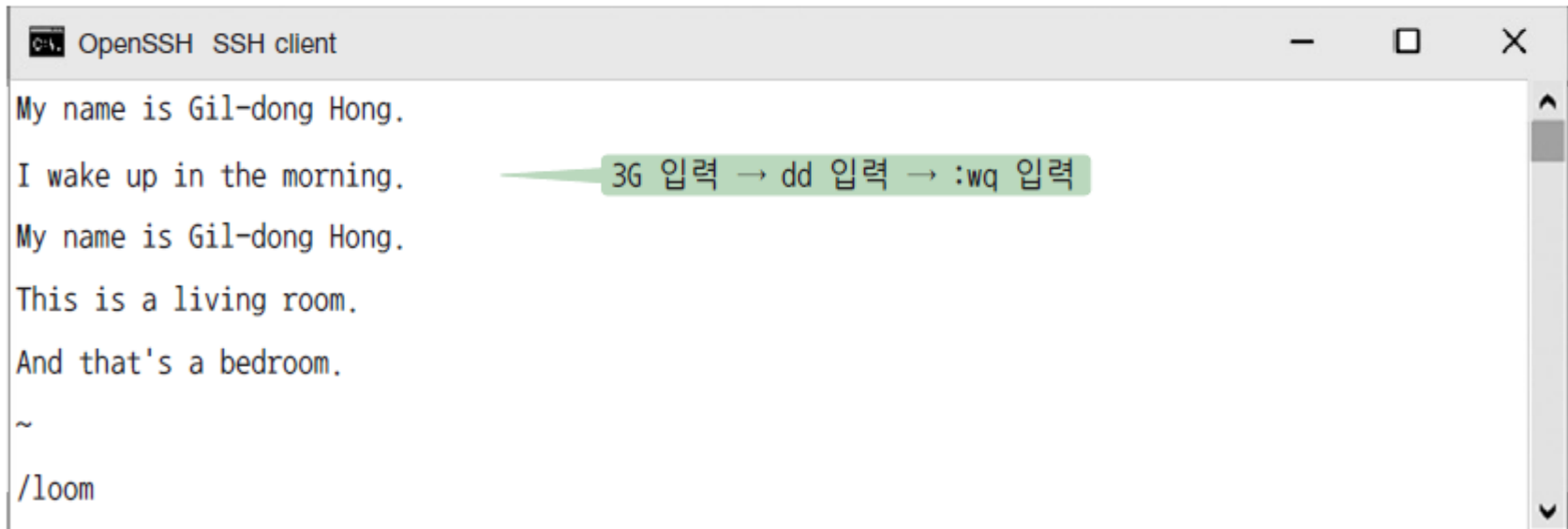
```
OpenSSH SSH client
My name is Gil-dong Hong.
I wake up in the morning.
My name is Gil-dong Hong.
This is a living loom.
And that's a bedroom.
~
/loom
```

:%s/loom/room/g 입력 → Enter 키 입력  
또는  
:1,\$s/loom/room/g 입력 → Enter 키 입력



## [실습 3-2] vim 편집 방법 익히기

### 6. 행을 삭제하고 저장한 후 종료하기



```
OpenSSH SSH client

My name is Gil-dong Hong.
I wake up in the morning.
My name is Gil-dong Hong.
This is a living room.
And that's a bedroom.
~
/loom
```

### 7. 디렉토리 위치 변경하기



```
OpenSSH SSH client

user10@Solaris11:~/Unix/ch4/Practice$ cd ..
user10@Solaris11:~/Unix/ch4$
```



## 02. vim의 사용 방법

### ❖ 비주얼 모드로의 전환

- 편집할 블록을 선택 → 비주얼 모드
- v, V, Ctrl+v 명령키 입력 → 서로 다른 비주얼 모드

표 4-14 비주얼 모드 전환키

구분	명령키	기능
비주얼	v	현재 커서 위치부터 블록 지정
비주얼 라인	V(Shift+v)	현재 커서가 위치한 행 단위로 블록 지정
비주얼 블록	Ctrl+v	현재 커서가 위치한 열부터 열 단위로 블록 지정

표 4-15 블록 적용 명령키

명령키	기능
y	블록 복사
d	블록 삭제



## 02. vim의 사용 방법

### ❖ 기타 유용한 명령키

- 파일 읽어오기/여러 파일 편집하기 명령키

표 4-16 vim 파일 관련 명령키

명령키	기능
:r 파일명	지정한 파일을 읽어 현재 커서 위치로 삽입한다.
:e 파일명	지정한 파일로 전환한다(기존 파일을 :w로 저장한 뒤에 실행해야 함).
:n	vim을 시작할 때 여러 파일명을 지정하였을 경우 다음 파일로 작업을 이동한다.

표 4-17 vim에서 셸 명령 사용 방법

명령키	기능
!: 셸 명령	vim 작업을 잠시 중단하고 셸 명령을 실행한다(vim으로 돌아오려면 <code>[Enter]</code> 키 입력).
:sh	vim을 잠시 빠져나가서 셸 명령을 실행한다(vim으로 돌아오려면 <code>exit</code> 명령 입력).



## 02. vim의 사용 방법

### ❖ 기타 유용한 명령키

- 기타 명령 알아보기

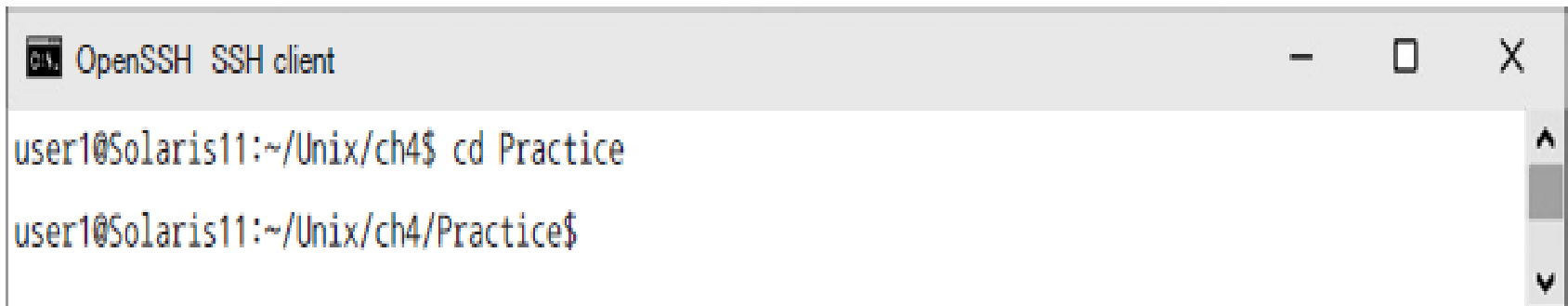
표 4-18 vim 기타 명령키

명령키	기능
<span>Shift</span> + j(대문자 J)	현재 행과 아래 행을 연결하여 하나의 행으로 만든다.
.	바로 직전에 했던 명령을 반복한다.



## [실습 3-3] 기타 유용한 명령 익히기

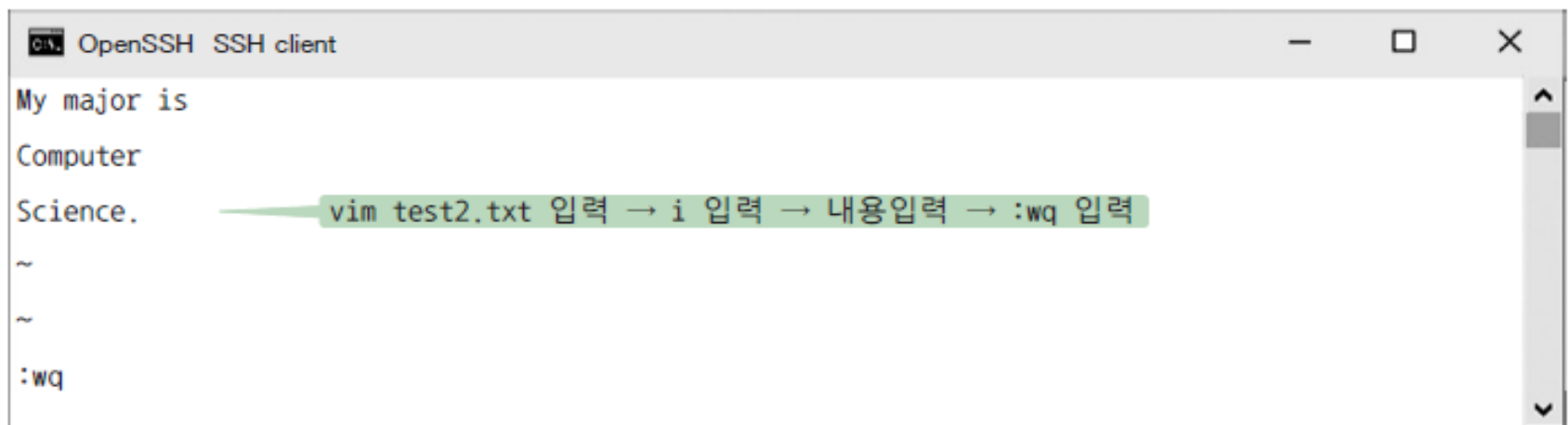
### 1. 현재 위치 이동하기



A terminal window titled "OpenSSH SSH client" showing a user at a Solaris11 machine. The user enters the command `cd Practice` to change the current directory from `~/Unix/ch4` to `~/Unix/ch4/Practice`. The prompt changes accordingly.

```
user1@Solaris11:~/Unix/ch4$ cd Practice
user1@Solaris11:~/Unix/ch4/Practice$
```

### 2. 실습용 예제 파일 하나 더 만들기



A terminal window titled "OpenSSH SSH client" showing a user creating a file named `test2.txt` using the `vim` editor. The user enters `vim test2.txt`, then `i` to enter insert mode, types the text "My major is Computer Science.", and finally enters `:wq` to save and quit. A green callout box points to the `vim test2.txt` command with the text "vim test2.txt 입력 → i 입력 → 내용입력 → :wq 입력".

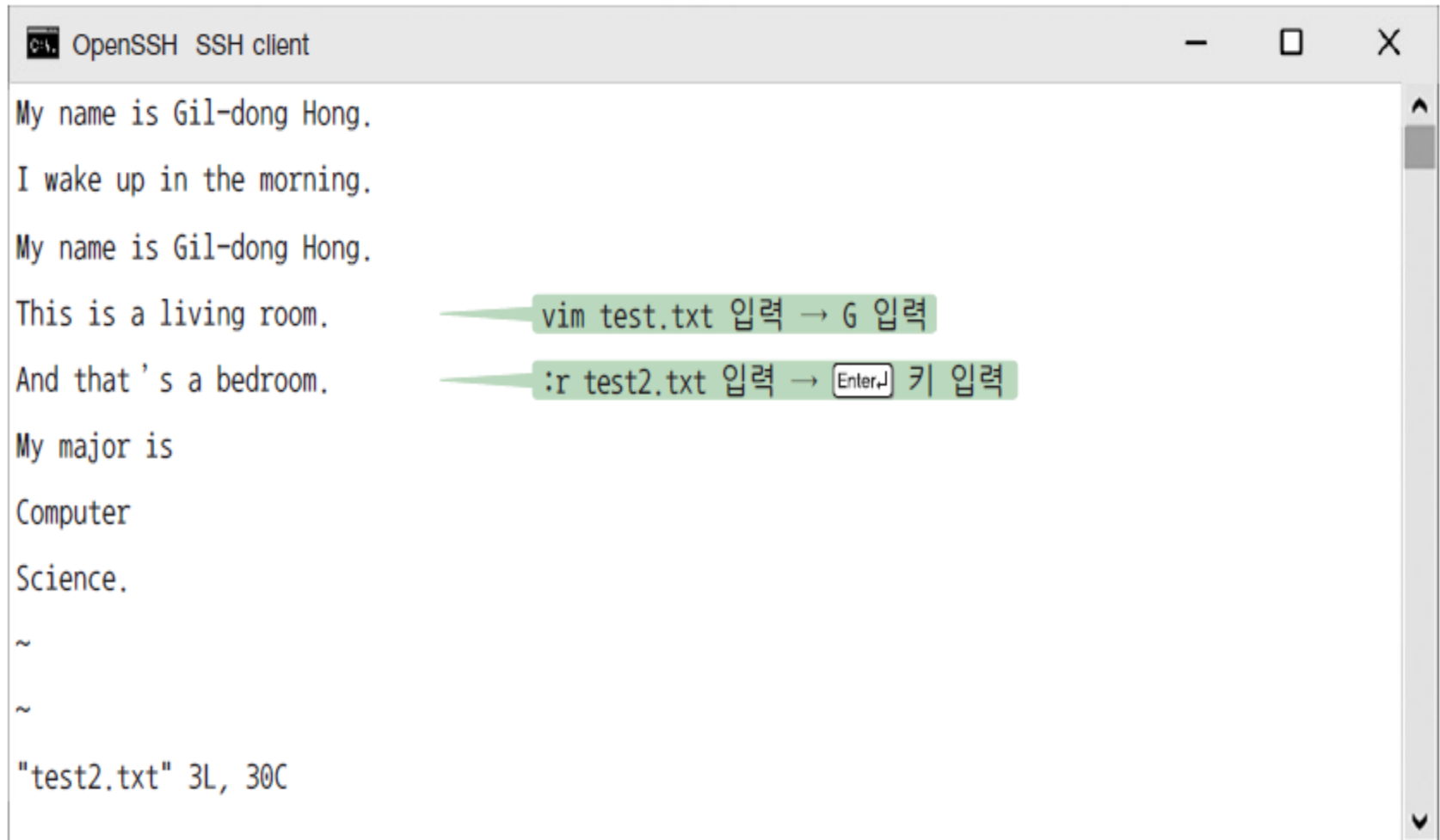
```
My major is
Computer
Science.
~
~
:wq
```

vim test2.txt 입력 → i 입력 → 내용입력 → :wq 입력



## [실습 3-3] 기타 유용한 명령 익히기

### 3. 다른 파일 읽어오기



```
OpenSSH SSH client
```

```
My name is Gil-dong Hong.  
I wake up in the morning.  
My name is Gil-dong Hong.  
This is a living room.  
And that 's a bedroom.  
My major is  
Computer  
Science.  
~  
~  
"test2.txt" 3L, 30C
```

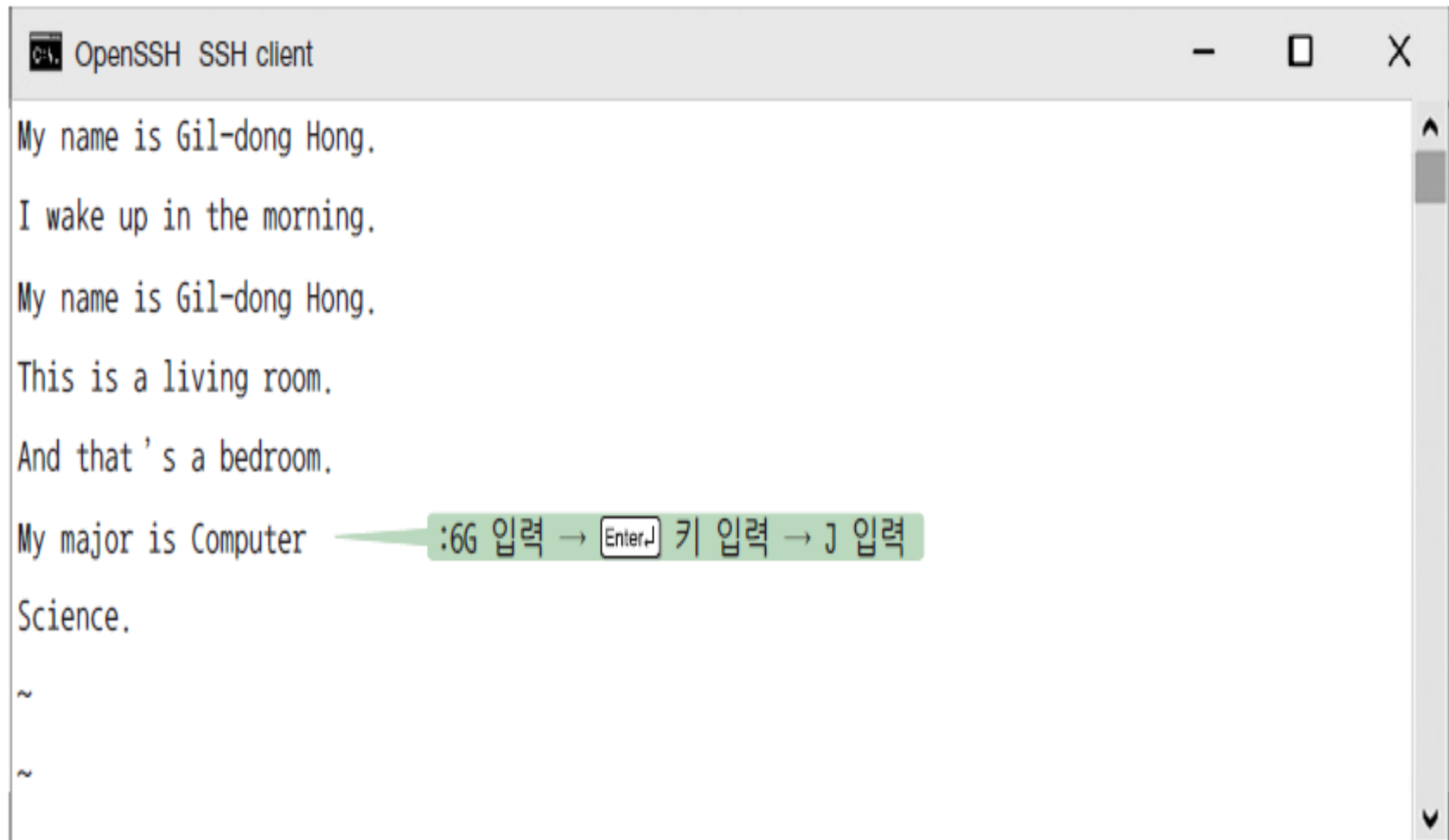
vim test.txt 입력 → G 입력

:r test2.txt 입력 → Enter 키 입력



## [실습 3-3] 기타 유용한 명령 익히기

### 4. 행 연결하기



The screenshot shows a terminal window titled "OpenSSH SSH client". The text inside the window is as follows:

```
My name is Gil-dong Hong.  
I wake up in the morning.  
My name is Gil-dong Hong.  
This is a living room.  
And that 's a bedroom.  
My major is Computer  
Science.  
~  
~
```

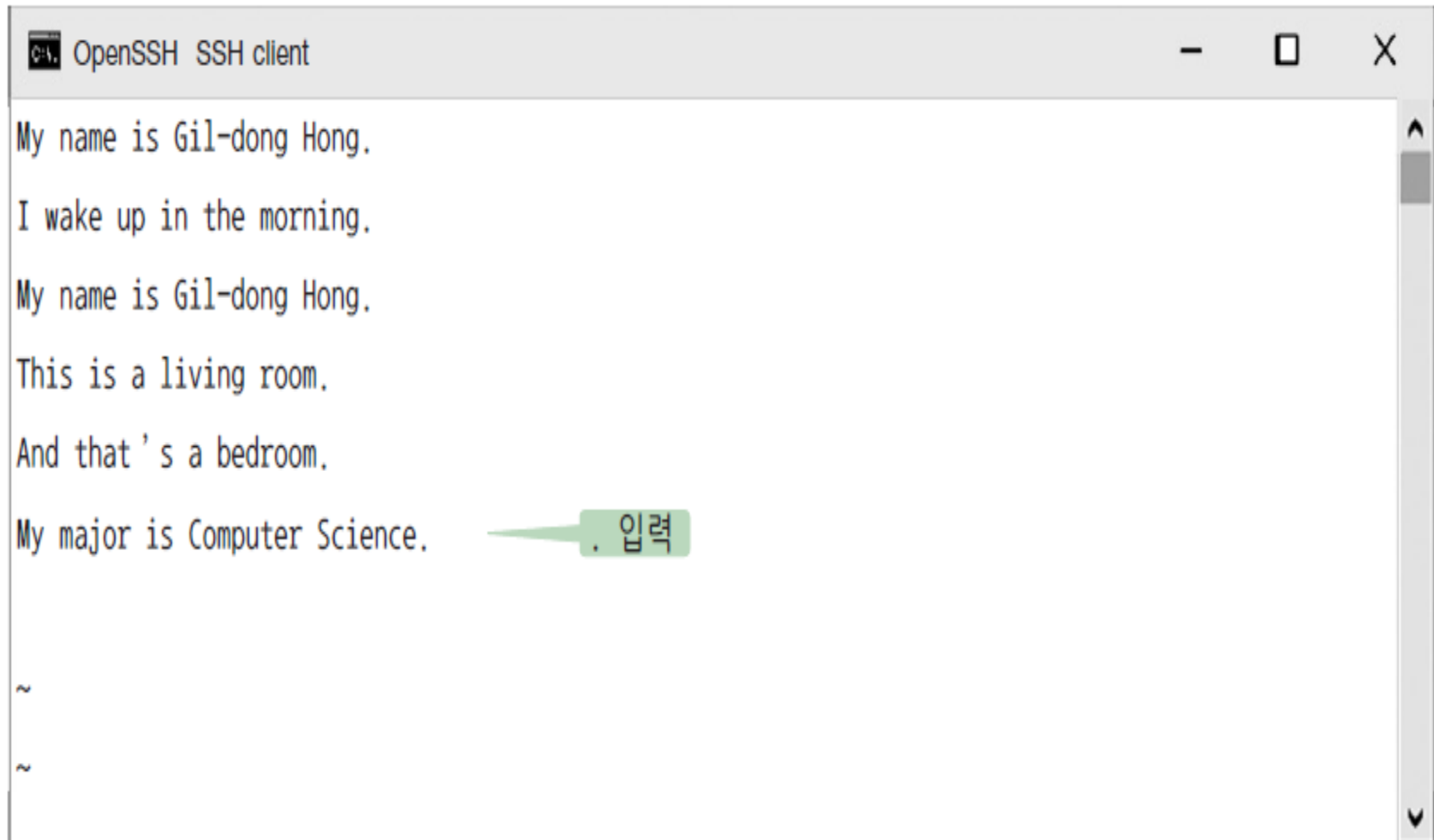
A green callout box points to the text "My major is Computer" with the text ":6G 입력 → Enter 키 입력 → J 입력".





## [실습 3-3] 기타 유용한 명령 익히기

### 5. 이전 동작 반복하기



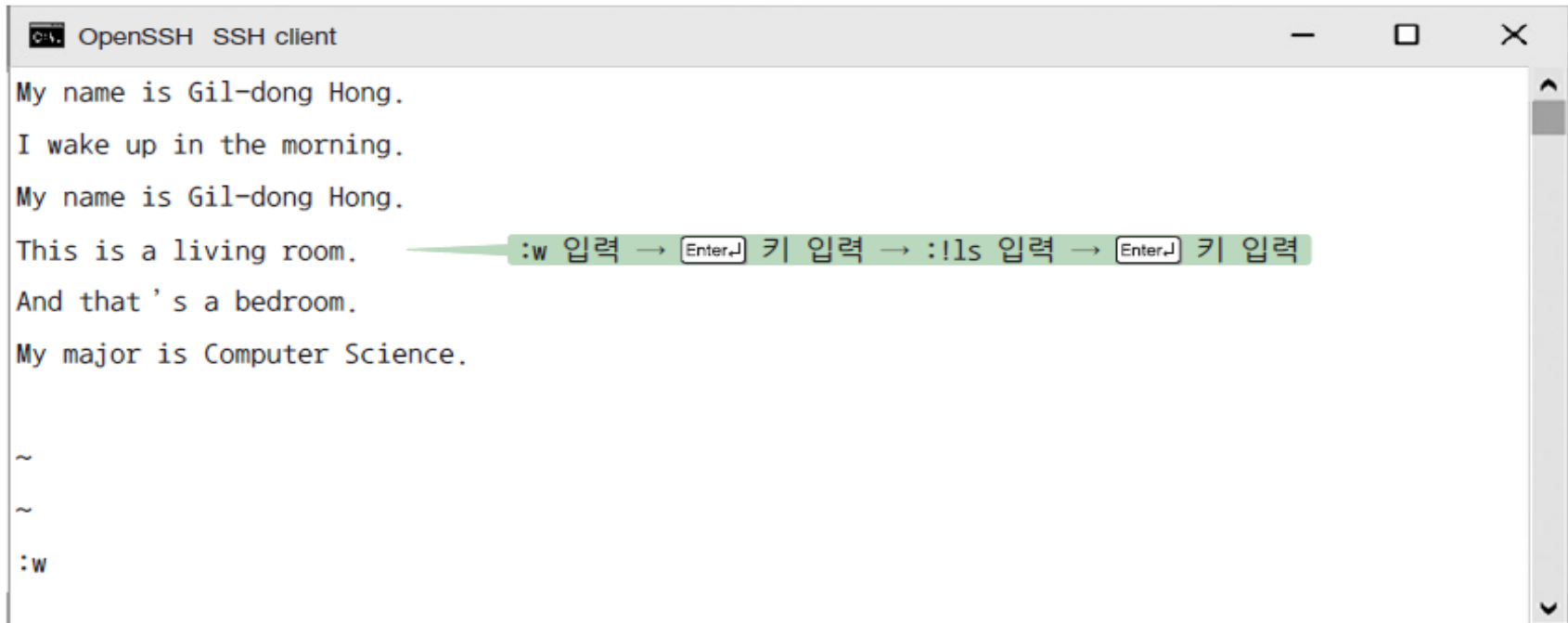
```
OpenSSH SSH client

My name is Gil-dong Hong.
I wake up in the morning.
My name is Gil-dong Hong.
This is a living room.
And that 's a bedroom.
My major is Computer Science. . 입력
~
~
```



## [실습 3-3] 기타 유용한 명령 익히기

### 6. 저장하고 셸 기능 사용하기

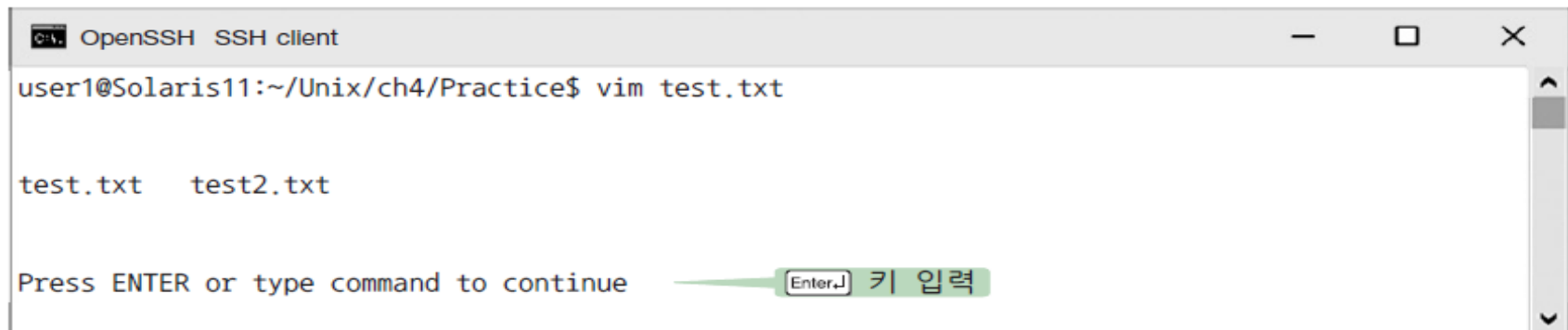


```
OpenSSH SSH client

My name is Gil-dong Hong.
I wake up in the morning.
My name is Gil-dong Hong.
This is a living room.
And that's a bedroom.
My major is Computer Science.

~
~
:w
```

:w 입력 → Enter 키 입력 → :!ls 입력 → Enter 키 입력



```
OpenSSH SSH client

user1@Solaris11:~/Unix/ch4/Practice$ vim test.txt

test.txt  test2.txt

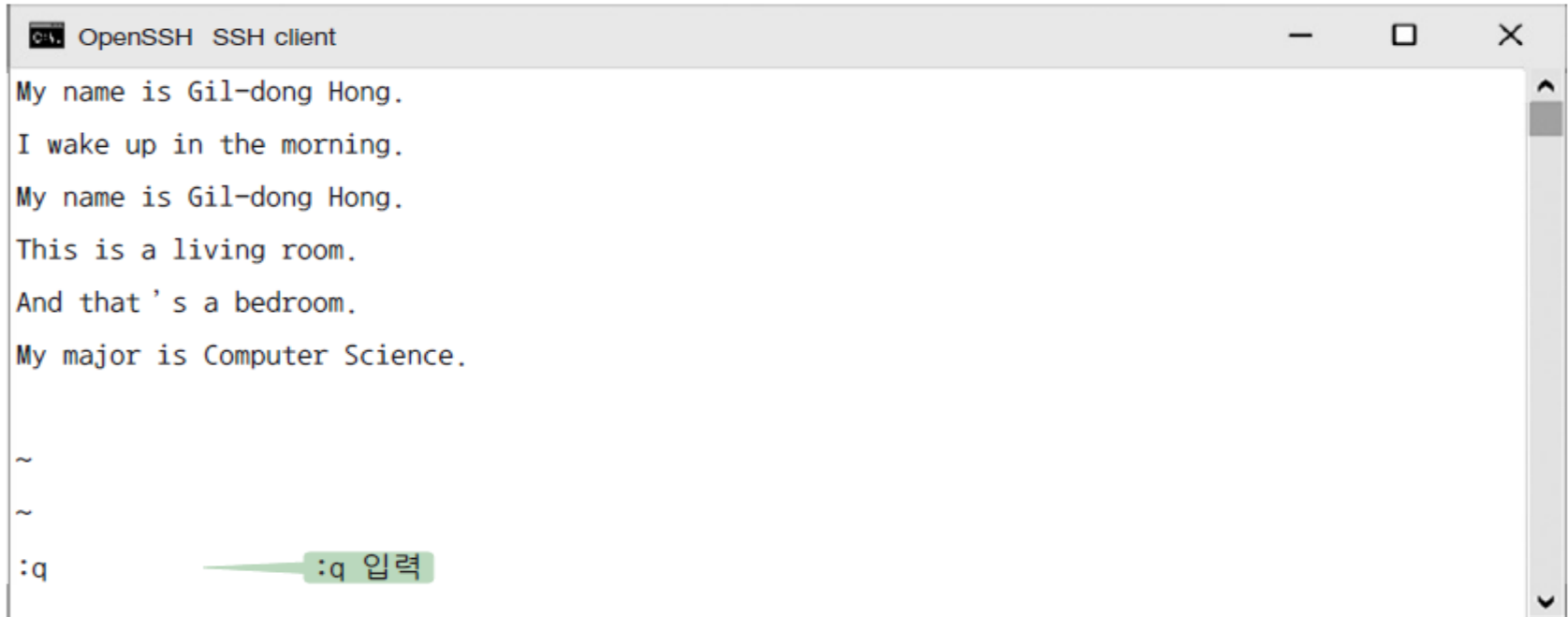
Press ENTER or type command to continue
```

:w 입력 → Enter 키 입력



## [실습 3-3] 기타 유용한 명령 익히기

### 7. vim 종료하기



```
OpenSSH SSH client
My name is Gil-dong Hong.
I wake up in the morning.
My name is Gil-dong Hong.
This is a living room.
And that 's a bedroom.
My major is Computer Science.
~
~
:q
```

A green callout bubble points to the `:q` command with the text `:q 입력`.

### 8. 디렉토리 위치 변경하기



```
OpenSSH SSH client
user1@Solaris11:~/Unix/ch4/Practice$ cd ..
user1@Solaris11:~/Unix/ch4@
```



## 03. vim의 환경 설정

### ❖ vi 환경을 설정하는 특수 명령과 변수들

표 4-19 vim 환경 설정 명령

명령	기능
:set nu	파일 내용의 각 행에 행 번호를 표시한다(보이기만 할 뿐 저장은 되지 않는다).
:set nonu	행 번호를 감춘다.
:set list	눈에 보이지 않는 특수 문자를 표시한다(tab:^\, eol:\$ 등).
:set nolist	특수 문자를 감춘다.
:set showmode	현재 모드를 표시한다.
:set noshowmode	모드 표시를 감춘다.
:set	set으로 설정한 모든 vim 변수를 출력한다.
:set all	모든 vim 변수와 현재 값을 출력한다.

