



**Sogang University: Dept of Computer Science and Engineering**

---

# **기초 공학 설계 (CSE2003)**

**Introduction to Engineering Design**

---

## **실습 1**

---

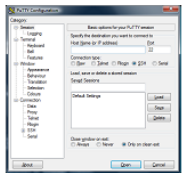
# Linux OS 환경에 대한 이해

## □ Linux 실습

- Linux 접속을 위한 프로그램 다운로드 및 설치하기
- 접속 프로그램을 이용하여, 리눅스 시스템에 접속하기

# Linux OS 환경에 대한 이해

- 접속 프로그램을 이용하여, 리눅스 시스템에 접속하기
  - PuTTY 실행(<https://www.putty.org/> 다운로드)



## Download PuTTY

PuTTY is an SSH and telnet client, developed originally by Simon Tatt and is developed and supported by a group of volunteers

You can download PuTTY [here](#).

Below suggestions are independent of the authors of PuTTY. They are *not*

## Package files

You probably want one of these. They include versions of all the PuTTY utilities.  
(Not sure whether you want the 32-bit or the 64-bit version? Read the [FAQ entry](#).)

### MSI ('Windows Installer')

32-bit:	<a href="#">putty-0.74-installer.msi</a>	<a href="#">(or by FTP)</a>	<a href="#">(signature)</a>
64-bit:	<a href="#">putty-64bit-0.74-installer.msi</a>	<a href="#">(or by FTP)</a>	<a href="#">(signature)</a>

### Unix source archive

.tar.gz:	<a href="#">putty-0.74.tar.gz</a>	<a href="#">(or by FTP)</a>	<a href="#">(signature)</a>
----------	-----------------------------------	-----------------------------	-----------------------------

## Bitvise SSH Client

Bitvise SSH Client is an SSH and SFTP client for Windows. It is developed and supports all features supported by PuTTY, as well as the following:

- graphical SFTP file transfer;
- single-click Remote Desktop tunneling;
- auto-reconnecting capability;
- dynamic port forwarding through an integrated proxy;
- an FTP-to-SFTP protocol bridge.

Bitvise SSH Client is **free to use**. You can [download it here](#).

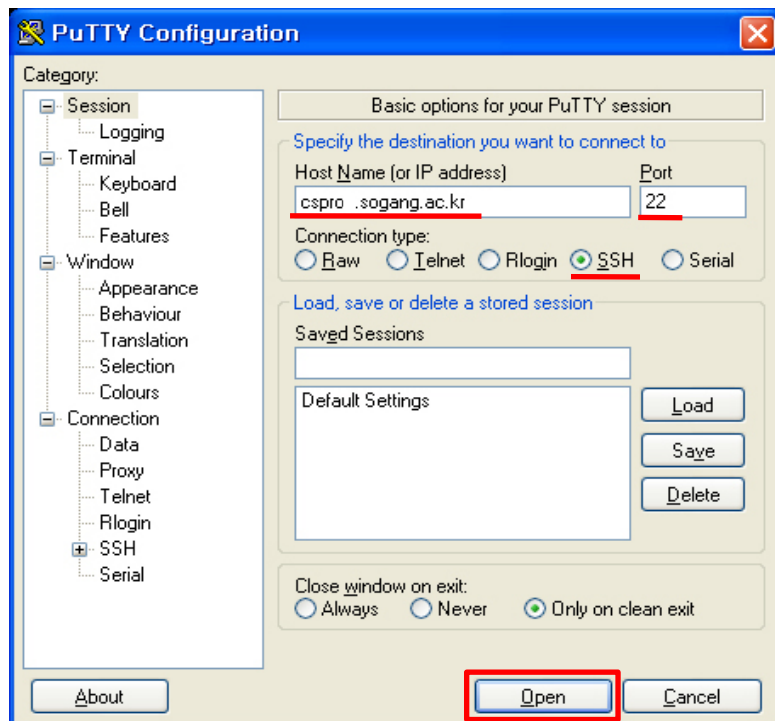
## Bitvise SSH Server

Bitvise SSH Server is an SSH, SFTP and SCP server for Windows. It is robust, easy to use, and supports all features supported by OpenSSH, and PuTTY. The SSH Server is developed and supported professionally by Bitvise.

You can [download Bitvise SSH Server here](#).

# Linux OS 환경에 대한 이해

- 접속 프로그램을 이용하여, 리눅스 시스템에 접속하기
  - PuTTY 실행
  - Host Name (or IP address)칸에 **cspro.sogang.ac.kr**입력 후, Open 버튼 클릭
  - 서강대학교 컴퓨터공학과는 Ubuntu 16, 리눅스를 사용함.



# Linux OS 환경에 대한 이해

소속	학번	password	이름
학부생	20182234	a123456	YUN XIANGXIN
학부생	20211494	a123456	강문석
학부생	20211497	a123456	강한이
학부생	20211501	a123456	권준상
학부생	20211504	a123456	김규현
학부생	20211507	a123456	김동건
학부생	20211511	a123456	김서아
학부생	20211514	a123456	김승준
학부생	20211517	a123456	김예원
학부생	20211521	a123456	김정원
학부생	20211524	a123456	김준우
학부생	20211527	a123456	김진
학부생	20211531	a123456	나호영
학부생	20211537	a123456	박준서
학부생	20211541	a123456	변정빈
학부생	20211543	a123456	성수빈
학부생	20211544	a123456	송종서
학부생	20211547	a123456	신지원
학부생	20211551	a123456	양현욱
학부생	20211557	a123456	윤제진
학부생	20211561	a123456	이규형
학부생	20211564	a123456	이상우
학부생	20211567	a123456	이승은
학부생	20211574	a123456	이재형
학부생	20211577	a123456	이준하
학부생	20211581	a123456	이희승
학부생	20211587	a123456	전준표
학부생	20211591	a123456	정지우
학부생	20211594	a123456	조경진
학부생	20211601	a123456	채민규
학부생	20211604	a123456	최지우
학부생	20211607	a123456	한유승

# Linux OS 환경에 대한 이해

- 접속 프로그램을 이용하여, 리눅스 시스템에 접속하기
  - 자신의 ID(cse학번)와 password를 입력한다.

```
login as: gr120080186
Using keyboard-interactive authentication.
Password:
Last login: Thu Nov 13 22:45:12 2008 from mclab21.sogang.ac.kr
gr120080186@csp01 ~ $
```

[ 로그인 성공 ]

```
login as: gr120080186
Using keyboard-interactive authentication.
Password:
Access denied
Using keyboard-interactive authentication.
Password:
```

[ 로그인 실패 ]

# Linux 명령어 실습

## □ passwd

- 자신의 비밀번호를 변경 시켜주는 명령어

passwd

```
gr120080186@cspiro ~ $ passwd
Changing password for gr120080186
(current) UNIX password:
New UNIX password:
Retype new UNIX password:
passwd: password updated successfully
gr120080186@cspiro ~ $
```

현재 패스워드를 입력(1번)

새로운 패스워드를 입력(2번)

## ■ who

- 현재 접속한 사용자의 정보 확인

who

```
gr120080186@cspiro1 ~ $ who
gr120080 pts/1 Jan 2 12:21 (mclab21.sogang.ac.kr)
gr120080186@cspiro1 ~ $
```

# Linux 명령어 실습

## □ cat

### □ 파일의 내용을 보는 명령어(catenate 의 약자)

`cat [filename]` 명시한 file의 내용을 출력한다.

```
gr120080186@cspro1 ~ $ cat
Assembly
Assembly
Operating System
Operating System
```

cat 명령어만 수행하면, 자신이 입력한 내용을 바로 연달아 화면에 출력해준다.

```
gr120080186@cspro1 ~ $ cat > catExample
Database
```

cat 다음에 ">"를 사용하면 입력한 내용을 ">" 다음에 명시한 catExample에 저장한다.

```
Data Structures
```

```
gr120080186@cspro1 ~ $ ls
```

입력이 끝나면 Ctrl + D를 눌러 내용을 저장한다.

```
catExample test1
```

```
gr120080186@cspro1 ~ $ cat catExample
```

cat 다음에 열어볼 파일명을 쓰면, 파일의 저장된 내용을 확인할 수 있다.

```
Database
```

```
Data Structures
```

```
gr120080186@cspro1 ~ $ cat < catExample > Redirection
```

```
gr120080186@cspro1 ~ $ ls
```

```
Redirection catExample test1
```

```
gr120080186@cspro1 ~ $ cat Redirection
```

기호 "<" 다음의 catExample의 내용을 받아서, 기호 ">" 다음의 Redirection 파일에 저장한다.

```
Database
```

```
Data Structures
```

### 리다이렉션(Redirection) 기호

>, < 는 리다이렉션 기호라고 합니다. 이 문자를 사용해서 입력(<)과 출력(>)을 할 수 있습니다.



# Linux 명령어 실습

## ls

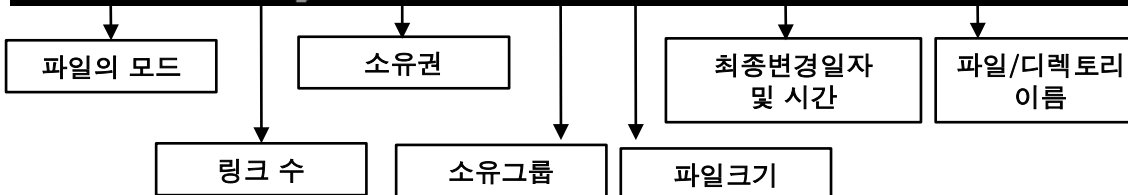
### 파일보기 명령어 ls (list의 약자)

**ls** 파일 이름 및 디렉토리 이름만 출력

```
gr120080186@cspro1 ~ $ ls
test1 test2
```

**ls -l** 파일의 모드, 소유권, 파일크기, 최종변경 일자 및 시간을 출력

```
gr120080186@cspro1 ~ $ ls -l
total 0
drwxr-xr-x  2 gr120080186 501 48 Nov 14 13:05 test1
drwxr-xr-x  2 gr120080186 501 48 Nov 14 13:05 test2
```



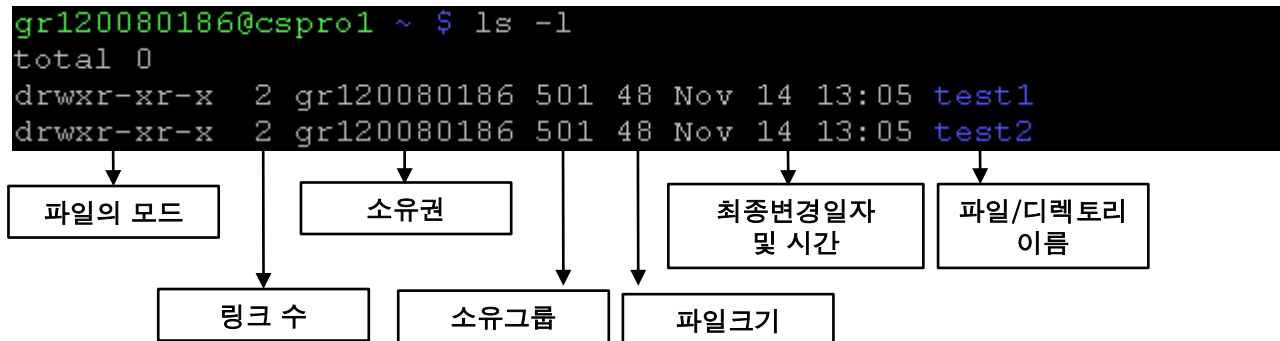
**ls -a** 숨겨진 파일(.(dot)으로 시작하는 파일)들까지 출력

```
gr120080186@cspro1 ~ $ ls -a
.  .bash_history  .bash_profile  .gdbinit  .viminfo  test1
.. .bash_logout  .bashrc        .ssh       .xsession  test2
```

# Linux 명령어 실습

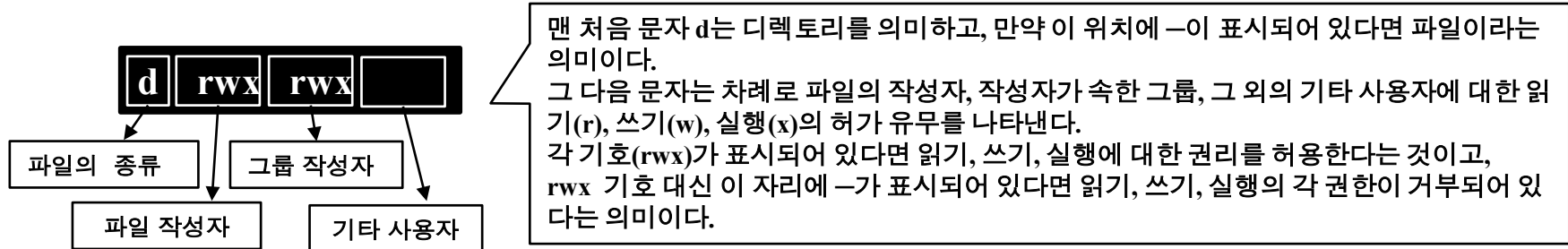
## □ chmod

- 파일의 권한을 변경하는 명령어
- chmod를 학습하기 전에 파일의 소유권과 허가권에 대해서 알아보자.
  - ls -l 명령을 사용하여 리눅스의 파일들을 열람하면 파일 이름 정도가 아닌 많은 정보들이 보인다.
  - 이 정보를 통해서, 이 파일이 누구의 소유이며 어떠한 사용자는 사용할 수 없는지를 알 수 있다.



# Linux 명령어 실습

## □ 파일의 모드를 살펴보면, 다음과 같다.



예1 `-rw-r--r-- 1 gr120080186 501 9 Dec 5 16:09 TestFile.txt`

- **TestFile.txt**의 파일의 종류 부분(맨 첫번째 기호)이 **-**이므로 **TestFile.txt**는 파일이다. 파일작성자, 같은 그룹에 속해 있는 사용자, 혹은 그밖에 다른 사용자들도 읽어볼 수 있는 권한이 설정되어 있다. 그리고 파일작성자만이 쓰기 권한을 가지고 있다.

예2 `dr-xr-xr-x 2 gr120080186 501 48 Dec 5 16:13 test`

- 파일인가? 디렉토리인가?
- 파일 작성자가 가지고 있는 권한들은 무엇인가?

# Linux 명령어 실습

## □ 디렉토리의 권한 변경 실습

- 본인 디렉토리를 다른 사람이 접근하지 못하도록 한다.

`ls -a -l` 모든 디렉토리의 정보 확인

```
drwxr-xr-x 2 cs20111589 cs20111589 4096 2015-03-02 21:41 .
```

현재 디렉토리 정보

`chmod [permission] [file_name/directory_name]`

```
cs20111589@CNUbuntuSrv:~/cs20111589$ chmod 700 ./
```

700 은 파일 모드.  
파일 작성자 이외에는  
아무도 접근 못하게 함.  
./ 은 현재 디렉토리

`ls -a -l` 모든 디렉토리의 정보 재확인

```
drwx----- 2 cs20111589 cs20111589 4096 2015-03-02 21:41 .
```

현재 디렉토리의 그룹과  
기타 사용자에게는  
아무 권한이 없음

# Linux 명령어 실습

## ❑ mkdir

- ❑ 디렉토리를 생성하는 명령어 (make directory의 약자)

`mkdir [directory_name]` directory\_name으로 디렉토리를 만든다.

```
gr120080186@cspro1 ~ $ mkdir test1
gr120080186@cspro1 ~ $ ls
test1
```

```
gr120080186@cspro1 ~ $ mkdir test2
gr120080186@cspro1 ~ $ ls
test1 test2
```

## ❑ rmdir

- ❑ 디렉토리를 삭제하는 명령어 (remove directory의 약자)

`rmdir [directory_name]` 디렉토리를 삭제한다.

```
gr120080186@cspro1 ~ $ rmdir test1
gr120080186@cspro1 ~ $ ls
test2
```

```
gr120080186@cspro1 ~ $ rmdir test2
rmdir: `test2': Directory not empty
```

디렉토리 test2안에 파일이나 다른 디렉토리가 존재한다면, rmdir로 삭제할 수 없다.

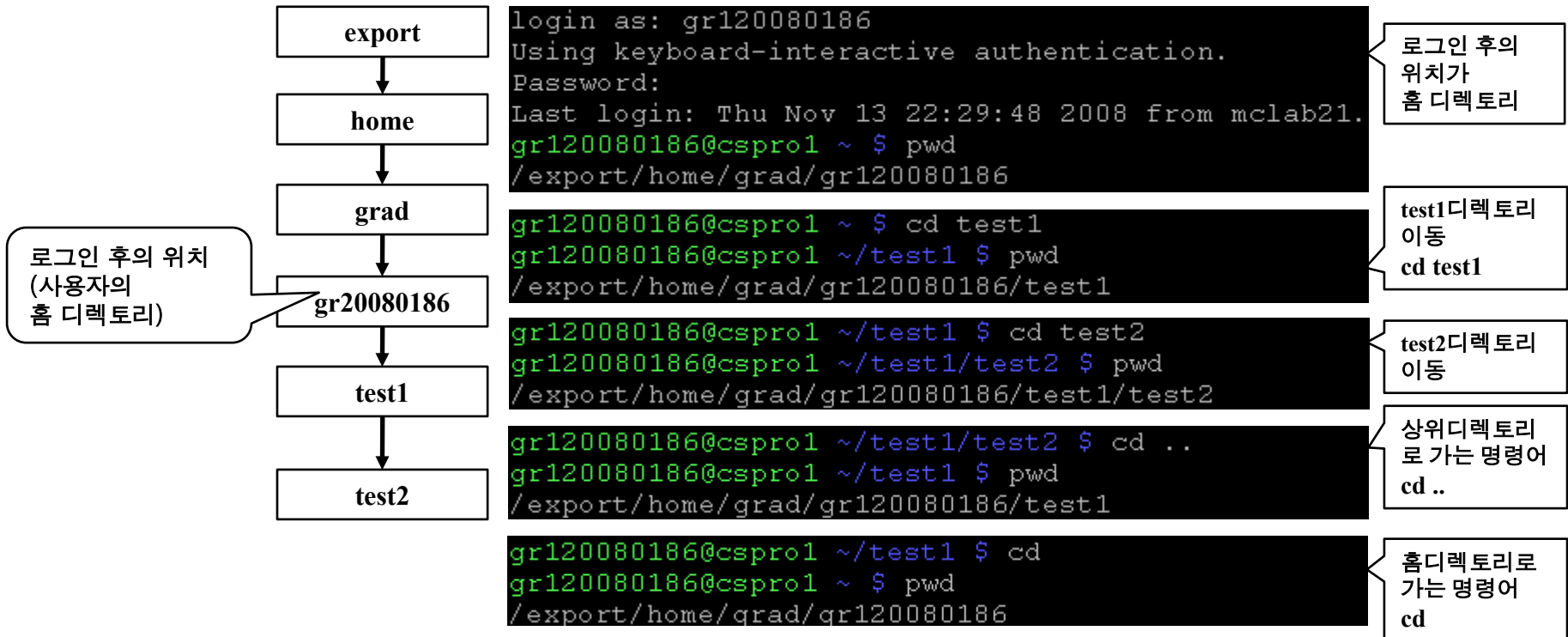
# Linux 명령어 실습

## □ cd

- 다른 디렉토리로 이동하기 위해 사용하는 명령어(change directory의 약자)

`cd [test1] / cd [..] / cd`

“cd test1”, “cd ..”, “cd” 모두 다른 기능을 수행한다.



# Linux 명령어 실습

## □ pwd

- 현재 자신이 어느 디렉토리에 위치해 있는지를 알려주는 명령어 (present working directory의 약자)

pwd

```
gr120080186@cspro1 ~ $ pwd
/export/home/grad/gr120080186
gr120080186@cspro1 ~ $ ls
test2
gr120080186@cspro1 ~ $ cd test2
gr120080186@cspro1 ~/test2 $ pwd
/export/home/grad/gr120080186/test2
gr120080186@cspro1 ~/test2 $ mkdir test3
gr120080186@cspro1 ~/test2 $ ls
test3
gr120080186@cspro1 ~/test2 $ cd test3
gr120080186@cspro1 ~/test2/test3 $ pwd
/export/home/grad/gr120080186/test2/test3
gr120080186@cspro1 ~/test2/test3 $
```

pwd를 이용하여 현재 내 디렉토리를 확인해보자.

내 디렉토리 밑에 있는 test2 디렉토리로 이동하자.  
“cd”에 대해서는 다음장에서 배워보자

test2 디렉토리 밑에 test3 디렉토리를 만들어보자.

“cd test2” 실행 후에, 내 현재 위치가 “test2”인지 확인해보자.

# Linux 명령어 실습

## □ rm

### □ 파일 삭제 명령어 (remove의 약자)

- 테스트 파일을 우선 만들어보자.

```
gr120080186@cspro1 ~ $ ls
test2
gr120080186@cspro1 ~ $ cat > testfile
Sogang Univ.
Computer Science
C language
gr120080186@cspro1 ~ $ ls
test2  testfile
gr120080186@cspro1 ~ $ cat testfile
Sogang Univ.
Computer Science
C language
```

입력이 끝나면 Ctrl + D를 눌러 내용을 저장한다.

testfile의 내용을 확인한다.  
Cat명령어에 대해서는 뒤에서 자세히 다루도록 하겠다.

**rm [filename]** filename을 삭제

```
gr120080186@cspro1 ~ $ rm testfile
gr120080186@cspro1 ~ $ ls
test2
gr120080186@cspro1 ~ $
```

Linux에는 윈도우 OS에 있는 휴지통이 없다. 즉, 한번 삭제된 파일은 복원할 수 없다.  
그러므로, rm 명령어를 사용할 때는 주의를 기울여야 한다.



# Linux 명령어 실습

## □ cp

- 파일을 복사하는 명령어 (copy의 약자)
  - 테스트 파일을 우선 만들어보자!!

```
gr120080186@cspro1 ~ $ cat > testfile1
Sogang Univ.
Computer Science
gr120080186@cspro1 ~ $
test1 testfile1
```

입력이 끝나면 Ctrl + D를 눌러 내용을 저장한다.

**cp [filename1] [filename2]** filename1을 filename2로 복사

```
gr120080186@cspro1 ~ $ cp testfile1 testfile2
gr120080186@cspro1 ~ $ ls
test1 testfile1 testfile2
gr120080186@cspro1 ~ $ cat testfile2
Sogang Univ.
Computer Science
```

```
gr120080186@cspro1 ~ $ cp testfile1 test1/testfile2
gr120080186@cspro1 ~ $ cd test1
gr120080186@cspro1 ~/test1 $ ls
test2 testfile2
```

특정 디렉토리로 복사를 할 수 있다.  
위의 예제는 test1 디렉토리로 testfile1을 복사하는 것을 보여준다.

# Linux 명령어 실습

## □ mv

- 파일을 이동 명령어 (move의 약자)

`mv [filename1] [filename2]` filename1을 filename2라는 파일 이름으로 이동시킨다.

```
gr120080186@csp1 ~/test1 $ ls
test2  testfile2
gr120080186@csp1 ~/test1 $ mv testfile2 sample
gr120080186@csp1 ~/test1 $ ls
sample test2
```

```
gr120080186@csp1 ~/test1 $ ls
sample test2
gr120080186@csp1 ~/test1 $ mv sample ..
gr120080186@csp1 ~/test1 $ ls
test2
gr120080186@csp1 ~/test1 $ cd ..
gr120080186@csp1 ~ $ ls
sample test1 testfile1 testfile2
```

현재 디렉토리에 있는 sample 파일을  
상위 디렉터리( .. )로 이동 시킨다.

\* cp 명령어와 mv 명령어의 차이점은 무엇일까?

# Linux 명령어 실습

## □ man

### □ 명령어의 도움말을 보여주는 명령어(manual 의 약자)

**man [command\_name]** 명시한 명령어의 내용을 출력한다.

```
gr120080186@cspro1 ~ $ man ls
LS(1)
NAME
    ls, dir, vdir - list directory contents
SYNOPSIS
    ls [options] [file...]
    dir [file...]
    vdir [file...]
POSIX options: [-CFRacdilqrutl] [--]
GNU options (shortest form): [-labcdfghiklmnopqrst
[-w cols] [-T cols] [-I pattern] [--full-time] [--
[--block-size=size] [--format={long,verbose,commas,s
gle-column}] [--sort={none,time}]
[--time={atime,access,use,ctime,status}] [--color={
[--help] [--version] [--]
```

명령어 ls에 대한 도움말을 찾아보자.

man 화면에서의 주요키(key)

Q: 종료

[space] : 한 페이지 넘기기

[enter]: 다음 한 줄 이동

[B]: 한 페이지 위로 올라가기

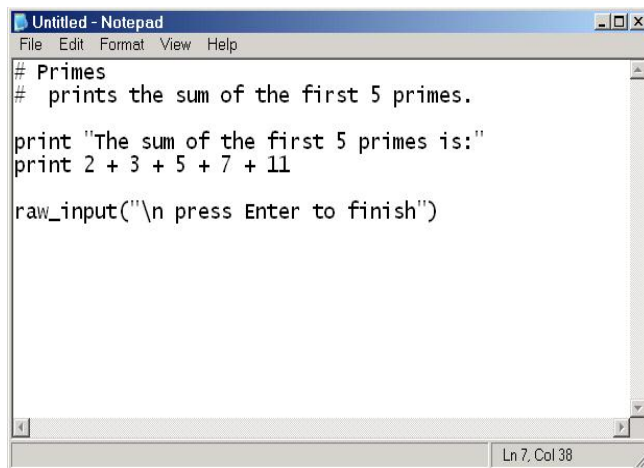
\*찾아봅시다\*

man을 활용하여, rm 명령어의 -f (rm -f)와 -r (rm -r)이 의미하는 것이 무엇인지 알아보자.

# vi 편집기 실습

## vi 편집기

- 원도우 메모장처럼 리눅스에서 널리 사용되는 편집기
- 막강한 기능을 가진 강력한 편집기



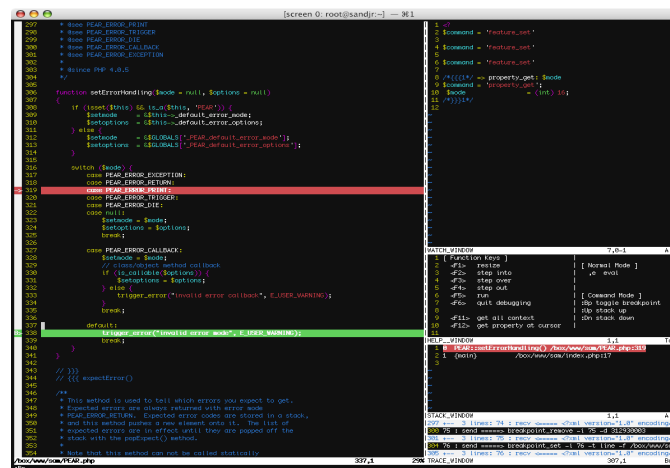
```

# Primes
# prints the sum of the first 5 primes.

print "The sum of the first 5 primes is:"
print 2 + 3 + 5 + 7 + 11

raw_input("\n press Enter to finish")
  
```

[ 윈도우 메모장 ]



```

207 * base PEAR_ERROR_RETURN
208 * base PEAR_ERROR_TRIGGER
209 * base PEAR_ERROR_INFO
210 * base PEAR_ERROR_CALLBACK
211 * base PEAR_ERROR_EXCEPTION
212 *
213 *
214 *
215 *
216 *
217 *
218 *
219 *
220 *
221 *
222 *
223 *
224 *
225 *
226 *
227 *
228 *
229 *
230 *
231 *
232 *
233 *
234 *
235 *
236 *
237 *
238 *
239 *
240 *
241 *
242 *
243 *
244 *
245 *
246 *
247 *
248 *
249 *
250 *
251 *
252 *
253 *
254 *
255 *
256 *
257 *
258 *
259 *
260 *
261 *
262 *
263 *
264 *
265 *
266 *
267 *
268 *
269 *
270 *
271 *
272 *
273 *
274 *
275 *
276 *
277 *
278 *
279 *
280 *
281 *
282 *
283 *
284 *
285 *
286 *
287 *
288 *
289 *
290 *
291 *
292 *
293 *
294 *
295 *
296 *
297 *
298 *
299 *
300 *
301 *
302 *
303 *
304 *
305 *
306 *
307 *
308 *
309 *
310 *
311 *
312 *
313 *
314 *
315 *
316 *
317 *
318 *
319 *
320 *
321 *
322 *
323 *
324 *
325 *
326 *
327 *
328 *
329 *
330 *
331 *
332 *
333 *
334 *
335 *
336 *
337 *
338 *
339 *
340 *
341 *
342 *
343 *
344 *
345 *
346 *
347 *
348 *
349 *
350 *
351 *
352 *
353 *
354 *
355 *
356 *
357 *
358 *
359 *
360 *
361 *
362 *
363 *
364 *
365 *
366 *
367 *
368 *
369 *
370 *
371 *
372 *
373 *
374 *
375 *
376 *
377 *
378 *
379 *
380 *
381 *
382 *
383 *
384 *
385 *
386 *
387 *
388 *
389 *
390 *
391 *
392 *
393 *
394 *
395 *
396 *
397 *
398 *
399 *
400 *
401 *
402 *
403 *
404 *
405 *
406 *
407 *
408 *
409 *
410 *
411 *
412 *
413 *
414 *
415 *
416 *
417 *
418 *
419 *
420 *
421 *
422 *
423 *
424 *
425 *
426 *
427 *
428 *
429 *
430 *
431 *
432 *
433 *
434 *
435 *
436 *
437 *
438 *
439 *
440 *
441 *
442 *
443 *
444 *
445 *
446 *
447 *
448 *
449 *
450 *
451 *
452 *
453 *
454 *
455 *
456 *
457 *
458 *
459 *
460 *
461 *
462 *
463 *
464 *
465 *
466 *
467 *
468 *
469 *
470 *
471 *
472 *
473 *
474 *
475 *
476 *
477 *
478 *
479 *
480 *
481 *
482 *
483 *
484 *
485 *
486 *
487 *
488 *
489 *
490 *
491 *
492 *
493 *
494 *
495 *
496 *
497 *
498 *
499 *
500 *
501 *
502 *
503 *
504 *
505 *
506 *
507 *
508 *
509 *
510 *
511 *
512 *
513 *
514 *
515 *
516 *
517 *
518 *
519 *
520 *
521 *
522 *
523 *
524 *
525 *
526 *
527 *
528 *
529 *
530 *
531 *
532 *
533 *
534 *
535 *
536 *
537 *
538 *
539 *
540 *
541 *
542 *
543 *
544 *
545 *
546 *
547 *
548 *
549 *
550 *
551 *
552 *
553 *
554 *
555 *
556 *
557 *
558 *
559 *
560 *
561 *
562 *
563 *
564 *
565 *
566 *
567 *
568 *
569 *
570 *
571 *
572 *
573 *
574 *
575 *
576 *
577 *
578 *
579 *
580 *
581 *
582 *
583 *
584 *
585 *
586 *
587 *
588 *
589 *
590 *
591 *
592 *
593 *
594 *
595 *
596 *
597 *
598 *
599 *
600 *
601 *
602 *
603 *
604 *
605 *
606 *
607 *
608 *
609 *
610 *
611 *
612 *
613 *
614 *
615 *
616 *
617 *
618 *
619 *
620 *
621 *
622 *
623 *
624 *
625 *
626 *
627 *
628 *
629 *
630 *
631 *
632 *
633 *
634 *
635 *
636 *
637 *
638 *
639 *
640 *
641 *
642 *
643 *
644 *
645 *
646 *
647 *
648 *
649 *
650 *
651 *
652 *
653 *
654 *
655 *
656 *
657 *
658 *
659 *
660 *
661 *
662 *
663 *
664 *
665 *
666 *
667 *
668 *
669 *
670 *
671 *
672 *
673 *
674 *
675 *
676 *
677 *
678 *
679 *
680 *
681 *
682 *
683 *
684 *
685 *
686 *
687 *
688 *
689 *
690 *
691 *
692 *
693 *
694 *
695 *
696 *
697 *
698 *
699 *
700 *
701 *
702 *
703 *
704 *
705 *
706 *
707 *
708 *
709 *
710 *
711 *
712 *
713 *
714 *
715 *
716 *
717 *
718 *
719 *
720 *
721 *
722 *
723 *
724 *
725 *
726 *
727 *
728 *
729 *
730 *
731 *
732 *
733 *
734 *
735 *
736 *
737 *
738 *
739 *
740 *
741 *
742 *
743 *
744 *
745 *
746 *
747 *
748 *
749 *
750 *
751 *
752 *
753 *
754 *
755 *
756 *
757 *
758 *
759 *
760 *
761 *
762 *
763 *
764 *
765 *
766 *
767 *
768 *
769 *
770 *
771 *
772 *
773 *
774 *
775 *
776 *
777 *
778 *
779 *
780 *
781 *
782 *
783 *
784 *
785 *
786 *
787 *
788 *
789 *
790 *
791 *
792 *
793 *
794 *
795 *
796 *
797 *
798 *
799 *
800 *
801 *
802 *
803 *
804 *
805 *
806 *
807 *
808 *
809 *
810 *
811 *
812 *
813 *
814 *
815 *
816 *
817 *
818 *
819 *
820 *
821 *
822 *
823 *
824 *
825 *
826 *
827 *
828 *
829 *
830 *
831 *
832 *
833 *
834 *
835 *
836 *
837 *
838 *
839 *
840 *
841 *
842 *
843 *
844 *
845 *
846 *
847 *
848 *
849 *
850 *
851 *
852 *
853 *
854 *
855 *
856 *
857 *
858 *
859 *
860 *
861 *
862 *
863 *
864 *
865 *
866 *
867 *
868 *
869 *
870 *
871 *
872 *
873 *
874 *
875 *
876 *
877 *
878 *
879 *
880 *
881 *
882 *
883 *
884 *
885 *
886 *
887 *
888 *
889 *
890 *
891 *
892 *
893 *
894 *
895 *
896 *
897 *
898 *
899 *
900 *
901 *
902 *
903 *
904 *
905 *
906 *
907 *
908 *
909 *
910 *
911 *
912 *
913 *
914 *
915 *
916 *
917 *
918 *
919 *
920 *
921 *
922 *
923 *
924 *
925 *
926 *
927 *
928 *
929 *
930 *
931 *
932 *
933 *
934 *
935 *
936 *
937 *
938 *
939 *
940 *
941 *
942 *
943 *
944 *
945 *
946 *
947 *
948 *
949 *
950 *
951 *
952 *
953 *
954 *
955 *
956 *
957 *
958 *
959 *
960 *
961 *
962 *
963 *
964 *
965 *
966 *
967 *
968 *
969 *
970 *
971 *
972 *
973 *
974 *
975 *
976 *
977 *
978 *
979 *
980 *
981 *
982 *
983 *
984 *
985 *
986 *
987 *
988 *
989 *
990 *
991 *
992 *
993 *
994 *
995 *
996 *
997 *
998 *
999 *
1000 *
  
```

[ vi 편집기 ]

# vi 편집기 실습

## □ vi

- vi를 입력하면 vi 편집기가 실행된다.

```
gr120080186@cspro1 ~ $ vi

      VIM - Vi IMproved
            version 6.3.84
      by Bram Moolenaar et al.
Vim is open source and freely distributable

  Help poor children in Uganda!
type  :help iccf<Enter>          for information

type  :q<Enter>                  to exit
type  :help<Enter>  or  <F1>    for on-line help
type  :help version6<Enter>    for version info
```

[ vi 실행 후, 첫 화면 ]

## vi 편집기 실습

## □ vi 시작하기

## □ 사용법

vi 파일명 <enter> 또는  
vi <enter>

**실습**

```
gr120080186@csp1 ~ $ vi test.txt
```

[illegible]

# vi 편집기 실습

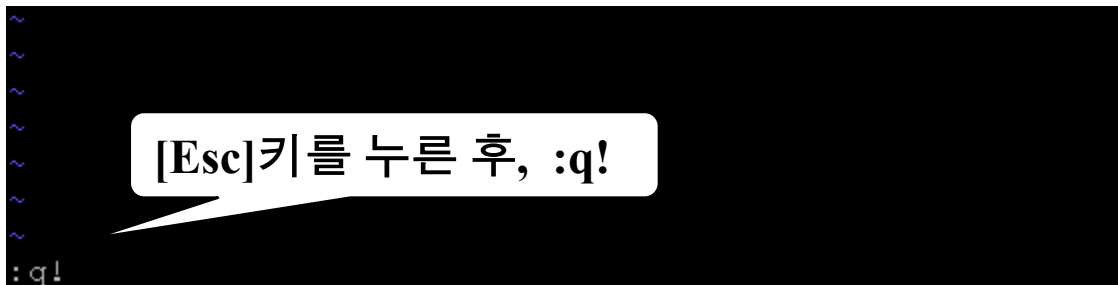
## □ vi 종료하기

- 먼저 [Esc] 키를 누른 후에 아래와 같은 키보드 명령을 입력한다.

명령어	내용
:q	그대로 종료하기
:q!	변경된 내용을 저장하지 않고 종료하기
:wq	변경된 내용을 저장하고 종료하기
:x	변경된 내용을 저장하고 종료하기
ZZ	변경된 내용을 저장하고 종료하기

## □ 실습

- 이전 예제 파일을 '그대로 종료'해보자.



# vi 편집기 실습

## □ vi의 모드(mode)

### □ 명령모드

- 입력된 텍스트에 대해 여러 가지 명령을 내릴 수 있는 모드

### □ 텍스트 입력 모드

- 실제 텍스트를 입력시키는 모드

### □ 커서 이동 모드

- 문장의 위아래나 좌우로 커서를 이동하게 하는 모드

### □ 모드 사이의 전환은 콜론(:)과 [Esc]키, 텍스트 입력 키를 이용한다.



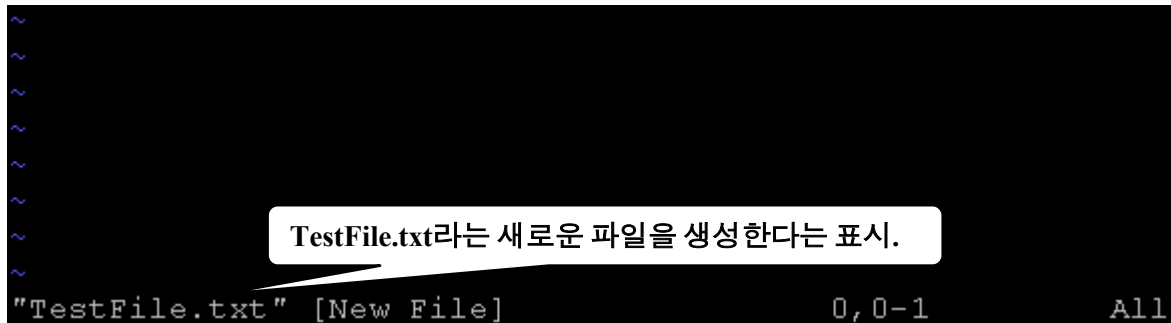
# vi 편집기 실습

## □ vi 실습

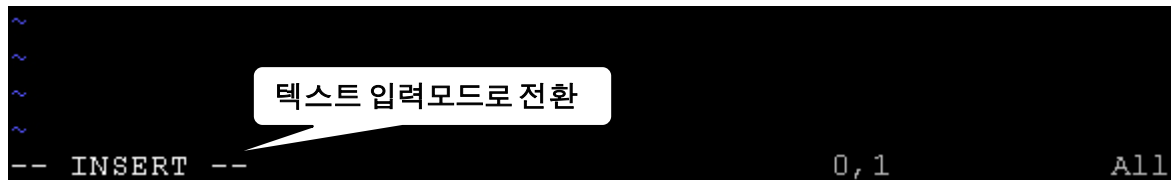
□ 다음의 과정을 하나씩 따라 하며, vi의 세 가지 모드에 대해서 배워보자.

1. vi로 만들게 될 파일의 이름을 입력한다.

```
gr120080186@cspro1 ~ $ vi TestFile.txt
```



2. i나 [Insert]키를 누르면 텍스트 입력모드로 전환한다.



# vi 편집기 실습

3. 이제 실제로 텍스트를 입력해보자. 다음의 내용을 모두 입력 후, [Esc]키를 누른다.

```
Test File

There are so many ways that the Internet has improved our
lives. I think the biggest improvement is that the Internet
has enabled us to find information instantly.
Indeed, online searches have become a huge part of how we
get infomation
~
8, 0-1 All
```

4. 내용을 저장하기 위해서는 저장 명령을 내릴 수 있는 명령 모드로 변경해야 한다.  
:(콜론)을 입력하여 명령모드로 전환한다.

```
Test File

There are so many ways that the Internet has improved our
lives. I think the biggest improvement is that the Internet
has enabled us to find information instantly.
Indeed, online searches have become a huge part of how we
get infomation
~
:
8, 0-1 All
```

# vi 편집기 실습

5. 이제 “변경된 내용을 저장하고 종료하기” 명령인 wq를 입력한다.

```
Test File

There are so many ways that the Internet has improved our
lives. I think the biggest improvement is that the Internet
has enabled us to find information instantly.
Indeed, online searches have become a huge part of how we
get infomation

~
~
:wq
```

6. TestFile.txt 파일이 생성되었는지 확인해보자.

```
gr120080186@cspro1 ~ $ ls
TestFile.txt
gr120080186@cspro1 ~ $ cat TestFile.txt
Test File

There are so many ways that the Internet has improved our
lives. I think the biggest improvement is that the Internet
has enabled us to find information instantly.
Indeed, online searches have become a huge part of how we
get infomation

gr120080186@cspro1 ~ $
```

# vi 편집기 실습

7. 파일의 내용을 수정하려면 파일을 처음 만들 때처럼 다음과 같이 입력하면 된다.

```
gr120080186@cspro1 ~ $ vi TestFile.txt
Test File

There are so many ways that the Internet has improved our
lives. I think the biggest improvement is that the Internet
has enabled us to find information instantly.
Indeed, online searches have become a huge part of how we
get infomation
```

# vi 편집기 실습

## □ 다양한 vi 명령

- 이제부터 명령 모드에서 텍스트 입력 모드로 전환하는 키와
- 각 모드에서 사용되는 명령 키를 알아보자.

## □ 입력 키의 종류

- 명령모드에서 텍스트 모드로 전환하려면 **[Insert]**키를 누르면 된다.
- 이 밖에도 vi 편집기에서는 여러 가지 기능키를 제공한다.

a	커서 위치의 다음 칸부터 입력하기
A	커서가 있는 줄의 끝부터 입력하기
i	커서 위치부터 입력하기(키보드의 [Insert]키도 같은 기능을 합니다.)
I	커서가 있는 줄의 맨 앞에서부터 입력하기
o	커서 바로 아래에 줄을 만들고 입력하기(open line)
O	커서 바로 위에 줄을 만들고 입력하기
s	커서가 있는 단어를 지우고 입력하기
S	커서가 있는 행을 지우고 입력하기

★ 대문자와 소문자 키의 기능이 다르다는 것을 잊지 말자.

# vi 편집기 실습

## 이동 키의 종류

- [Esc]키를 이용하면 커서 이동모드로 돌아온다.
- 커서의 이동은 윈도우와 마찬가지로 방향키(←, →, ↓, ↑)를 사용하면 된다.

### ❖ 방향키가 안되는데요?

커서 이동키로 이동하는 것도 가능하지만 어떤 리눅스 시스템에서는 Vi를 실행했을 때 커서 이동키가 작동하지 않습니다. 그럴 경우에는 h (←), l (→), j (↓), k (↑)를 이용해 방향을 이동하세요.

## 그 밖의 커서 이동키 열람

글자 단위	h	한 칸 왼쪽으로 이동
	L	한 칸 오른쪽으로 이동
	J	한 줄 아래로 이동
	K	한 줄 위로 이동
단어 단위	w, W	다음 단어의 첫 글자로 이동(W는 특수문자 무시)
	b, B	이전 단어의 첫 글자로 이동(B는 특수문자 무시)
	e, E	단어의 마지막 글자로 이동(E는 특수문자 무시)
줄 단위	^	그 줄의 첫 글자로 이동
	\$	그 줄의 마지막 글자로 이동
	0	(숫자) 그 줄의 처음으로 이동
	+	다음 줄의 첫 글자로 이동
	-	윗줄의 첫 글자로 이동

# vi 편집기 실습

## □ 삭제 키의 종류

### □ 기본 삭제 키

- ▣ [Del]키 : “커서 이동 모드”에서, 한 글자 삭제
- ▣ [Backspace]키 : “텍스트 입력모드”에서, 한 글자 삭제

### □ 추가 삭제 명령키 (커서 이동모드에서)

---

- x     커서 위치의 글자 삭제
  - X     커서 바로 앞의 글자 삭제
  - dw    한 단어를 삭제
  - d0    커서 위치부터 줄의 처음까지 삭제
  - D     커서 위치부터 줄의 끝까지 삭제
  - dd    커서가 있는 줄을 삭제
  - dj    커서가 있는 줄과 그 다음 줄을 삭제
  - dk    커서가 있는 줄과 그 앞줄을 삭제
-

# vi 편집기 실습

## 수정 키의 종류

### 수정 명령키 (커서 이동모드에서)

r	커서 위치의 한 글자 수정하기
R	커서 위치부터 [Esc]를 누를 때까지 다른 글자로 수정하기. 단, 같은 줄에만 해당
s	커서 위치의 한 글자를 여러 글자로 수정하기
ch	커서 바로 앞의 한 글자를 여러 글자로 수정하기
cw	커서 위치의 한 단어를 수정하기
c0	커서 위치부터 줄의 처음까지 수정하기
C	커서 위치부터 줄의 끝까지 수정하기
cc	커서가 있는 줄을 수정하기
cj	커서가 있는 줄과 그 다음 줄을 수정하기
ck	커서가 있는 줄과 그 앞줄을 수정하기
~	대문자를 소문자로, 소문자를 대문자로 수정한다.

```
There are many ways that the Internet has improved our
lives. I think the most important is that the Internet
has enable s 입력 후, ...를 여러번 입력
Email have innovated the way we communicate with one
another.....
Indeed, online searches have become a part of how we
get infomationNi!!!!!!
```

r을 입력 후, n을 대문자로 변경



# vi 편집기 실습

## □ 복사와 붙여넣기 키의 종류

- 텍스트의 삭제나 교체도 임시 기억 장소인 버퍼(buffer)에 저장된다.

---

yw	커서 위치부터 단어의 끝까지 복사
y0	커서 위치부터 줄의 처음까지 복사
y\$	커서 위치부터 줄의 끝까지 복사
yy	커서가 있는 줄 복사
yj	커서가 있는 줄과 그 다음 줄을 복사
yk	커서가 있는 줄과 그 앞줄을 복사
p	커서의 다음 위치에 붙여넣기
P	커서가 있는 위치에 붙여넣기

---

```
iNDEED, ONLINE SEARCHES HAVE BECOME A PART OF HOW WE GET IN
FOMATIOn!!!!!!!!!!
iNDEED, ONLINE SEARCHES HAVE BECOME A PART OF HOW WE GET IN
FOMATIOn!!!!!!!!!!
iNDEED, ONLINE SEARCHES HAVE BECOME A PART OF HOW WE GET IN
FOMA!!!!!!!!!!
```

yy를 이용하여 커서가 있는 줄을 복사합니다. 그리고 p를  
두번 눌러 붙여넣기를 실행합니다.

# vi 편집기 실습

## ❑ 취소 키의 종류

### ❑ 작업 취소 명령

---

u	작업 취소(undo)
U	그 줄에 행해진 작업 모두 취소
.	조금 전에 했던 명령을 반복

---

- ❑ 텍스트에서  
“information”이라는 단어를  
찾으려면,

```
~  
/information
```

## ❑ 검색 키의 종류

### ❑ 검색 명령

---

/pattern	텍스트에서 앞으로 패턴 검색
>pattern	텍스트에서 뒤로 패턴 검색
n	앞 또는 이전 검색 반복
N	반대방향으로 이전 검색 반복
/	전 검색을 앞으로 반복
?	전 검색을 뒤로 반복

---

# FileZilla

- 서버에 있는 파일을 업로드/다운로드를 하기 위한 프로그램
- 다운로드 : <https://filezilla-project.org/>

The screenshot shows the FileZilla homepage. On the left is a sidebar with navigation links: Home, FileZilla (Features, Screenshots, Download, Documentation, FileZilla Pro), FileZilla Server (Download), Community (Forum, Project page, Wiki), General (FAQ, Support, Contact, License, Privacy Policy, Trademark Policy), Development (Source code, Nightly builds, Translations, Version history, Changelog, Issue tracker), and Other projects (libfilezilla, Octochess). Sponsors for NDC and Network Data Center North Inc. are listed at the bottom of the sidebar. The main content area has a 'Promotion' banner with the FileZilla logo. Below it is the 'Overview' section, followed by 'Quick download links' which contains two buttons: 'Download FileZilla Client' (All platforms) and 'Download FileZilla Server' (Windows only). A red box highlights the 'Quick download links' section. Below the buttons is a 'News' section with two entries: '2021-01-18 - FileZilla Client 3.52.2 released' and '2021-01-08 - FileZilla Client 3.52.0.5 released', each with a list of bugfixes and minor changes.

Home

**FileZilla**

- Features
- Screenshots
- Download
- Documentation
- FileZilla Pro

**FileZilla Server**

- Download

**Community**

- Forum
- Project page
- Wiki

**General**

- FAQ
- Support
- Contact
- License
- Privacy Policy
- Trademark Policy

**Development**

- Source code
- Nightly builds
- Translations
- Version history
- Changelog
- Issue tracker

**Other projects**

- libfilezilla
- Octochess

Sponsors:

**NDC** NETWORK DATA CENTER NORTH INC.

**Promotion:**

**Overview**

Welcome to the homepage of FileZilla®, the free FTP solution. The *FileZilla Client* not only supports FTP, but also FTP over TLS. We are also offering *FileZilla Pro*, with additional protocol support for WebDAV, Amazon S3, Backblaze B2, Dropbox, Microsoft OneDrive. Last but not least, *FileZilla Server* is a free open source FTP and FTPS Server. Support is available through our [forums](#), the [wiki](#) and the [bug and feature request trackers](#). In addition, you will find documentation on how to compile FileZilla and nightly builds for multiple platforms in the developer's area.

**Quick download links**

**Download FileZilla Client**  
All platforms

**Download FileZilla Server**  
Windows only

Pick the client if you want to transfer files. Get the server if you want to make files available for others.

**News**

2021-01-18 - FileZilla Client 3.52.2 released

**Bugfixes and minor changes:**

- When restoring the main window position at startup, detect if the title bar would become hidden behind taskbars or
- SFTP: Fixed preserving of timestamps on downloads

2021-01-08 - FileZilla Client 3.52.0.5 released

**Bugfixes and minor changes:**

# FileZilla

FileZilla Client Interface showing a connection to a remote server via SFTP.

**Host Information:**

- Host (H): sftp://gr120210194@cspro.sogang.ac.kr
- User (U): gr120210194
- Password (W): [Masked]
- Port (P): [Blank]
- Quick Connect (Q): [Checked]

**Status:** Connected to cspro.sogang.ac.kr

**Local Site:** C:\Users\wasf\W

**Remote Site:** /sogang/grad/gr120210194

**Local Site Contents:**

- Temp
- Users
- All Users
- asf
- Default
- Default User
- Public
- Windows
- E: (새 볼륨)
- F: (새 볼륨)
- H:

**Remote Site Contents:**

- sogang
  - grad
    - gr120210194

**File List (Local):**

파일명	크기	파일 유형	최종 수정
...			
.anaconda		파일 폴더	2019-06-27 오후 ...
.conda		파일 폴더	2019-06-28 오전 ...
.idlerc		파일 폴더	2019-05-22 오후 ...
.PyCharmCE2019.1		파일 폴더	2019-05-22 오후 ...
.Xilinx		파일 폴더	2018-11-27 오후 ...
3D Objects		파일 폴더	2020-07-28 오전 ...
Anaconda3		파일 폴더	2019-01-28 오후 ...
AppData		파일 폴더	2021-03-03 오전 ...
Application Data		파일 폴더	2021-01-26 오후 ...
Contacts		파일 폴더	2020-07-28 오전 ...
Cookies		파일 폴더	2020-07-28 오전 ...
Desktop		파일 폴더	2021-02-26 오후 ...
Documents		파일 폴더	2020-12-01 오후 ...
Downloads		파일 폴더	2021-03-03 오전 ...

9 파일 및 33 디렉터리. 총 크기: 7,533,660 바이트

**File List (Remote):**

파일명	크기	파일 유형	최종 수정	권한	소유자/그룹
...					
.cache		파일 폴더	2021-03-02 ...	drwx-----	gr1202101...
.local		파일 폴더	2021-03-02 ...	drwx-----	gr1202101...
prac1		파일 폴더	2021-03-02 ...	drwxr-xr-x	gr1202101...
prac1final		파일 폴더	2021-03-02 ...	drwxr-xr-x	gr1202101...
python		파일 폴더	2021-03-03 ...	drwxr-xr-x	gr1202101...
.bash_history	879	BASH_HIS...	2021-03-02 ...	-rw-----	gr1202101...
.bash_logout	220	BASH_LOG...	2021-03-02 ...	-rw-r--r--	gr1202101...
.bashrc	3,771	BASHRC ...	2021-03-02 ...	-rw-r--r--	gr1202101...
.profile	655	PROFILE ...	2021-03-02 ...	-rw-r--r--	gr1202101...
.viminfo	3,678	VIMINFO ...	2021-03-03 ...	-rw-----	gr1202101...
hello	28	파일	2021-03-02 ...	-rwxrwxrwx	gr1202101...
test.py	23	Python File	2021-03-03 ...	-rw-r--r--	gr1202101...

7 파일 및 5 디렉터리. 총 크기: 9,254 바이트

**Bottom Bar:**

- Server/Local File: [Blank]
- Direction: [Blank]
- Remote File: [Blank]
- Size: [Blank]
- Options: [Blank]
- Status: [Blank]

**Footer:**

- 대기 파일
- 전송 실패
- 전송 성공
- 대기열: 비었음
- 35

# FileZilla

FileZilla interface showing a connection to a remote server via SFTP.

**Host Information:**

- Host (H): sftp://gr120210194@cspro.sogang.ac.kr
- User (U): gr120210194
- Password (W): [Masked]
- Port (P): [Blank]
- Quick Connect (Q): [Checked]

**Status:** Connected to cspro.sogang.ac.kr

**Local Site:** C:\Users\wasf\W

**Remote Site:** /sogang/grad/gr120210194

**Local Site Contents:**

- Temp
- Users
- All Users
- asf
- Default
- Default User
- Public
- Windows
- E: (새 볼륨)
- F: (새 볼륨)
- H:

**Remote Site Contents:**

- sogang
  - grad
    - gr120210194

**Local Site File List:**

파일명	크기	파일 유형	최종 수정
..		파일 폴더	
.anaconda		파일 폴더	2019-06-27 오후 ...
.conda		파일 폴더	2019-06-28 오전 ...
.idlerc		파일 폴더	2019-05-22 오후 ...
.PyCharmCE2019.1		파일 폴더	2019-05-22 오후 ...
.Xilinx		파일 폴더	2018-11-27 오후 ...
3D Objects		파일 폴더	2020-07-28 오전 ...
Anaconda3		파일 폴더	2019-01-28 오후 ...
AppData		파일 폴더	2021-03-03 오전 ...
Application Data		파일 폴더	2021-01-26 오후 ...
Contacts		파일 폴더	2020-07-28 오전 ...
Cookies		파일 폴더	2020-07-28 오전 ...
Desktop		파일 폴더	2021-02-26 오후 ...
Documents		파일 폴더	2020-12-01 오후 ...
Downloads		파일 폴더	2021-03-03 오전 ...

9 파일 및 33 디렉터리. 총 크기: 7,533,660 바이트

**Remote Site File List:**

파일명	크기	파일 유형	최종 수정	권한	소유자/그룹
..		파일 폴더			
.cache		파일 폴더	2021-03-02 ...	drwx-----	gr1202101...
.local		파일 폴더	2021-03-02 ...	drwx-----	gr1202101...
prac1		파일 폴더	2021-03-02 ...	drwxr-xr-x	gr1202101...
prac1final		파일 폴더	2021-03-02 ...	drwxr-xr-x	gr1202101...
python		파일 폴더	2021-03-03 ...	drwxr-xr-x	gr1202101...
.bash_history	879	BASH_HIS...	2021-03-02 ...	-rw-----	gr1202101...
.bash_logout	220	BASH_LOG...	2021-03-02 ...	-rw-r--r--	gr1202101...
.bashrc	3,771	BASHRC ...	2021-03-02 ...	-rw-r--r--	gr1202101...
.profile	655	PROFILE ...	2021-03-02 ...	-rw-r--r--	gr1202101...
.viminfo	3,678	VIMINFO ...	2021-03-03 ...	-rw-----	gr1202101...
hello	28	파일	2021-03-02 ...	-rwxrwxrwx	gr1202101...
test.py	23	Python File	2021-03-03 ...	-rw-r--r--	gr1202101...

7 파일 및 5 디렉터리. 총 크기: 9,254 바이트

**Bottom Bar:**

- Server/Local File: [Blank]
- Direction: [Blank]
- Remote File: [Blank]
- Size: [Blank]
- Options: [Blank]
- Status: [Blank]

**Footer:**

- 대기 파일
- 전송 실패
- 전송 성공
- 대기열: 비었음
- 36

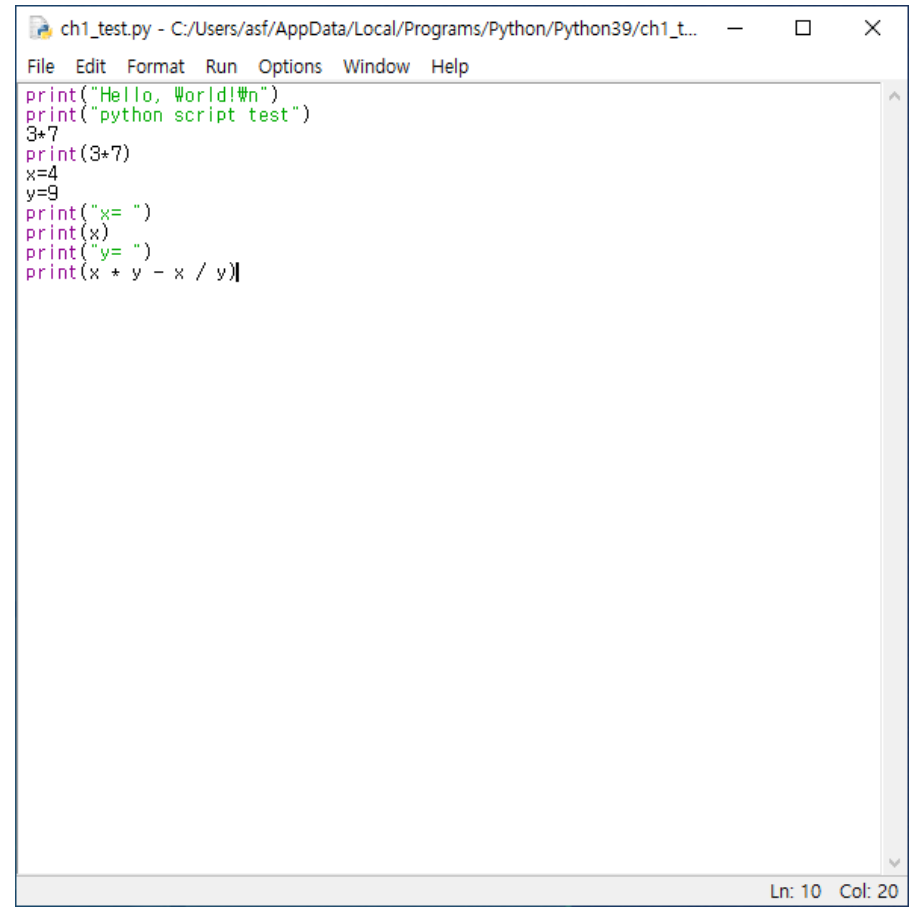
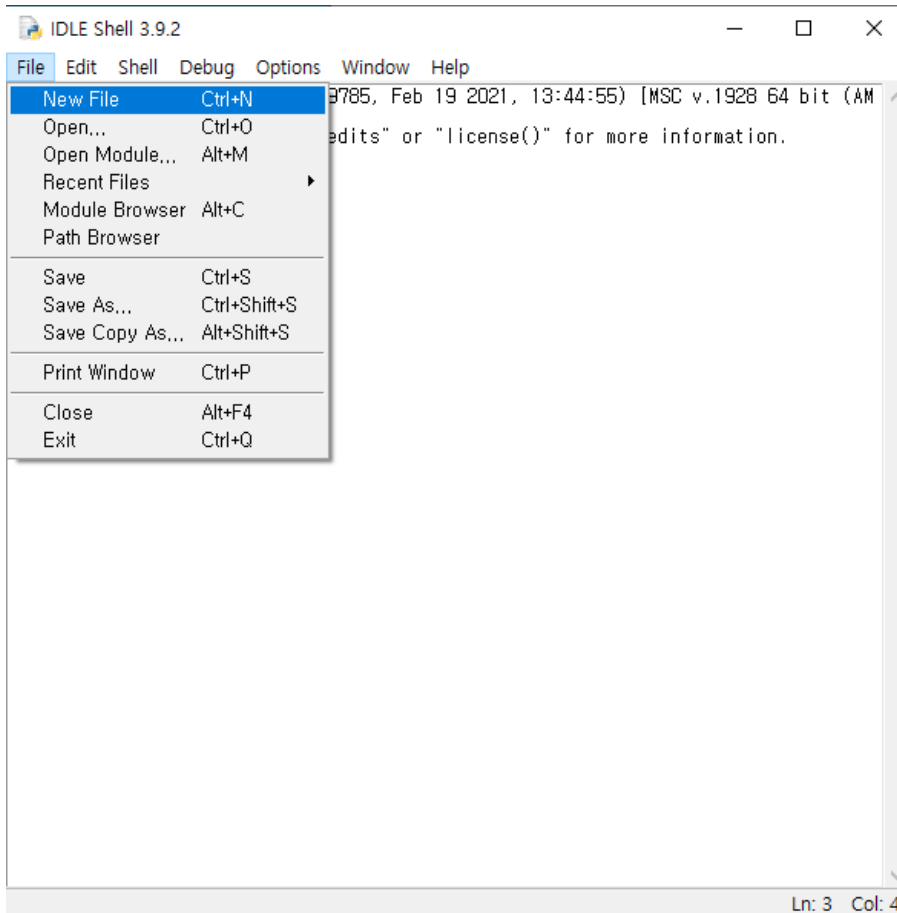
# Python 설치

- Python.org 에서 다운로드
- 서버가 아닌 PC에서 프로그래밍 할 시 사용

The screenshot shows the Python.org homepage. The navigation bar includes links for About, Downloads, Documentation, Community, Success Stories, News, and Events. The 'Downloads' menu is open, showing options like All releases, Source code, Windows, Mac OS X, Other Platforms, License, and Alternative Implementations. The 'Windows' option is highlighted with a red box, and a sub-menu is displayed showing 'Python 3.9.2' as the selected version. A note states: 'Note that Python 3.9+ cannot be used on Windows 7 or earlier.' Below this, it says 'Not the OS you are looking for? Python can be used on many operating systems and environments. View the full list of downloads.' The footer text reads: 'Python is a programming language that lets you work quickly and integrate systems more effectively. >>> [Learn More](#)'

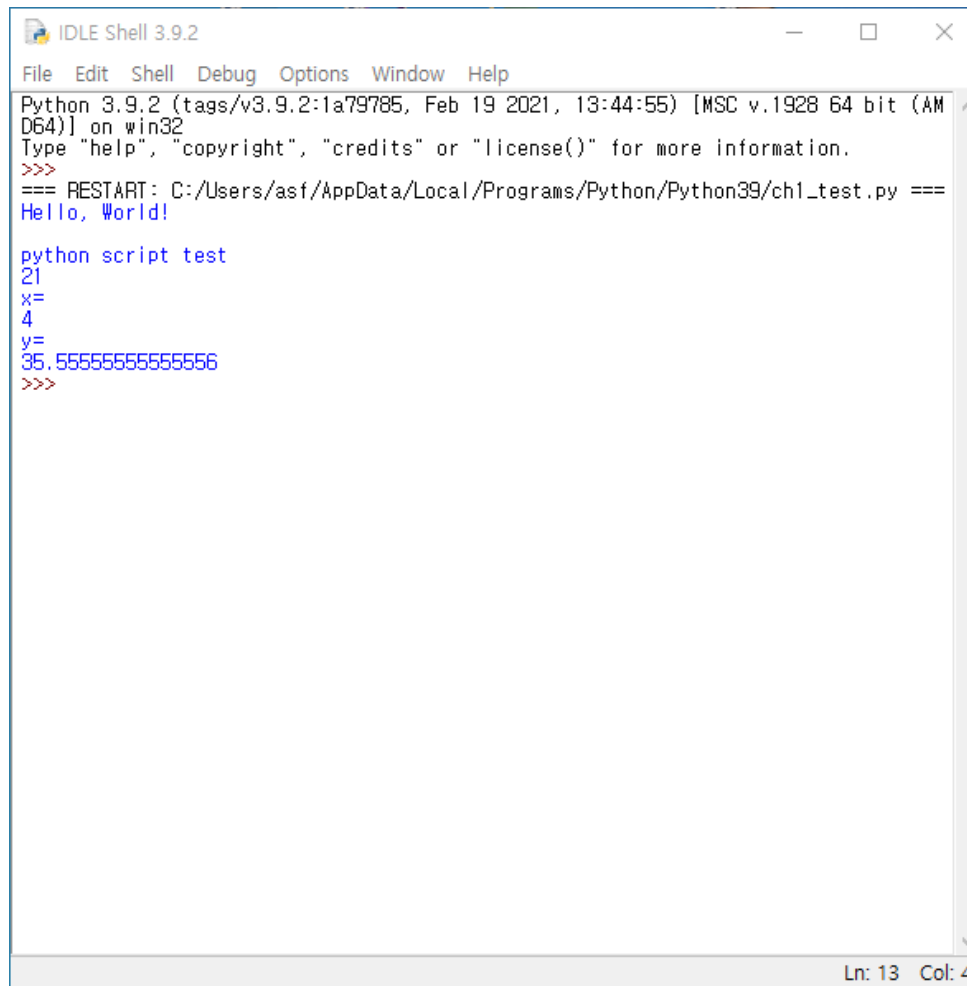
# Python 설치

- 검색->IDLE 실행 -> File탭에서 New File 클릭
- 프로그래밍 후 F5 눌러서 저장 및 실행



# Python 설치

## □ 실행화면



The screenshot shows the IDLE Shell 3.9.2 window. The title bar reads 'IDLE Shell 3.9.2'. The menu bar includes 'File', 'Edit', 'Shell', 'Debug', 'Options', 'Window', and 'Help'. The main text area displays the following content:

```
Python 3.9.2 (tags/v3.9.2:1a79785, Feb 19 2021, 13:44:55) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
=== RESTART: C:/Users/asf/AppData/Local/Programs/Python/Python39/ch1_test.py ===
Hello, World!

python script test
21
x=
4
y=
35.55555555555556
>>>
```

The status bar at the bottom right indicates 'Ln: 13 Col: 4'.



# 실습1 - Hello World

- Unix 상에서 test.py라는 python파일을 만든 후 실행을 하여 강의 자료에 있는 내용을 출력하시오.
  - test.py는 vi를 이용하여 생성.
  - 컴파일은 python3를 사용.

# 실습1 - Hello World

- vi를 실행시켜 빈 파일을 생성

```
grl20210194@cspiro:~$ vi test.py
```

# 실습1 - Hello World

[illegible]

# 실습1 - Hello World

- Insertion 명령어를 수행해서 입력대기 상태로 바꾼다
  - i 를 입력
- 소스코드를 입력 후 esc를 눌러 입력을 종료하고 파일을 저장하고 종료한다.
  - 커맨드를 입력하기 위해서는 입력모드가 아닐 때 :를 입력한다.
  - 저장은 w, 종료는 q이다.
  - 저장 후 바로 종료하기 위해서는 wq를 입력하면 된다.
  - 저장하지 않고 종료하려면 q!를 입력한다.

# 실습1 - Hello World

- 작성된 파일을 python3를 사용하여 실행한다.
- python3 뒤에 파일 이름을 붙이면 실행된다.

```
grl20210194@cspiro:~$ python3 test.py
hello, world!
python script test
21
grl20210194@cspiro:~$
```