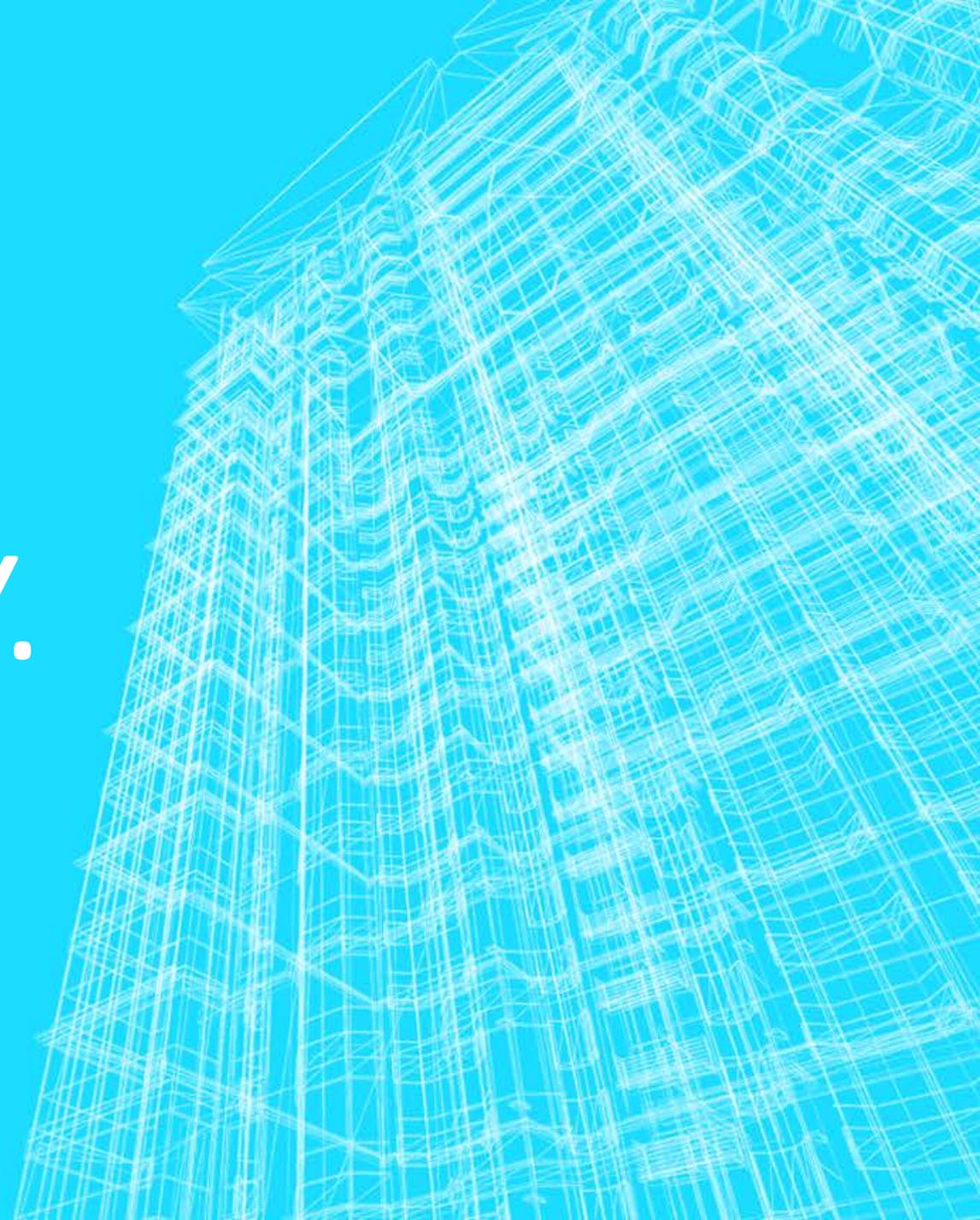
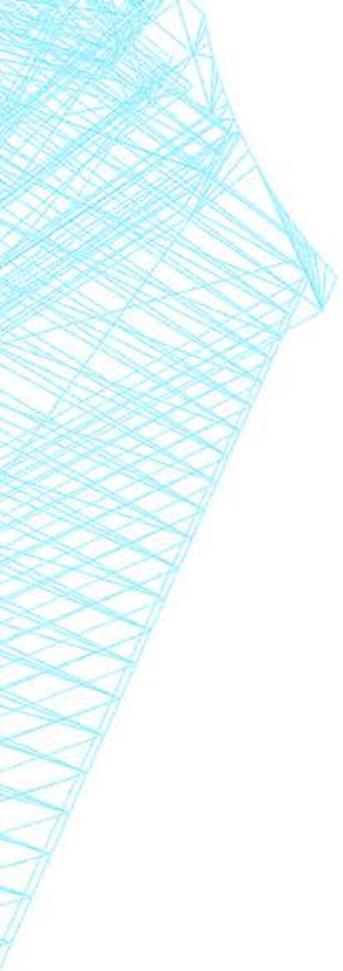


LINUX. VIM. TOY.

2015-03-25

chocho





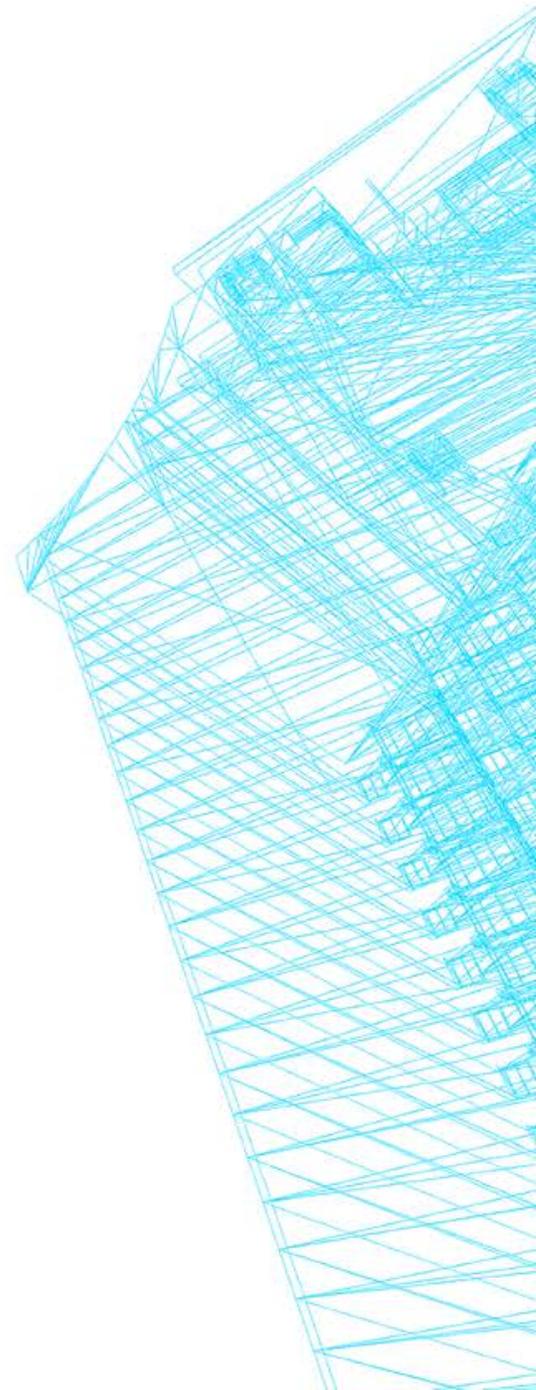
LET'S TALK ABOUT

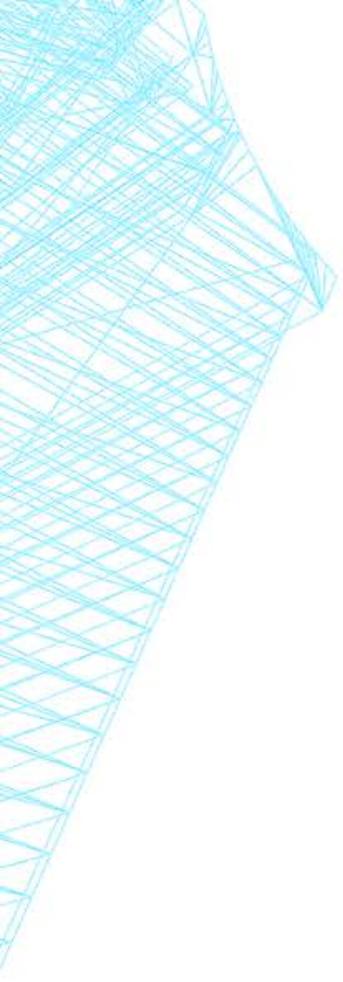
- Linux 리눅스
- Vim 빔
- TOY 장난감

LINUX

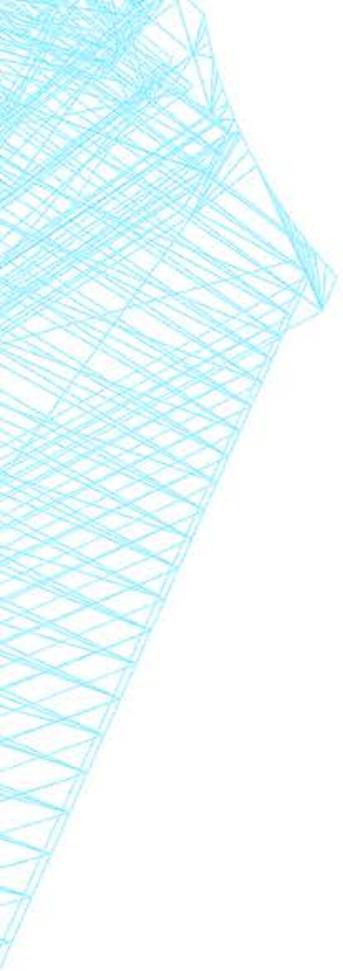
VIM

TOY



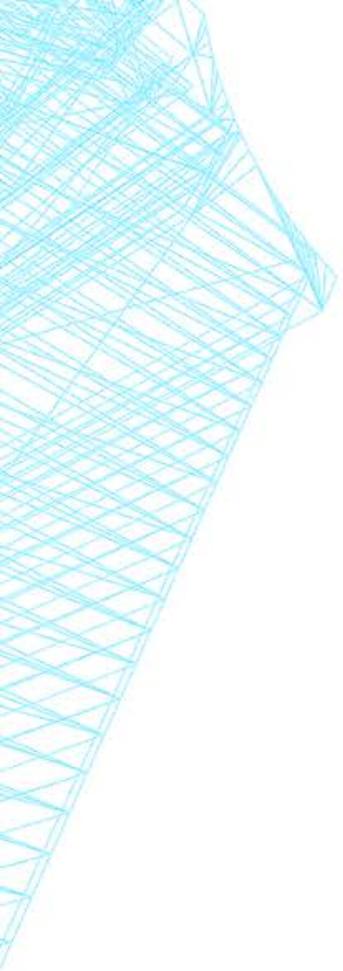


리눅스란?



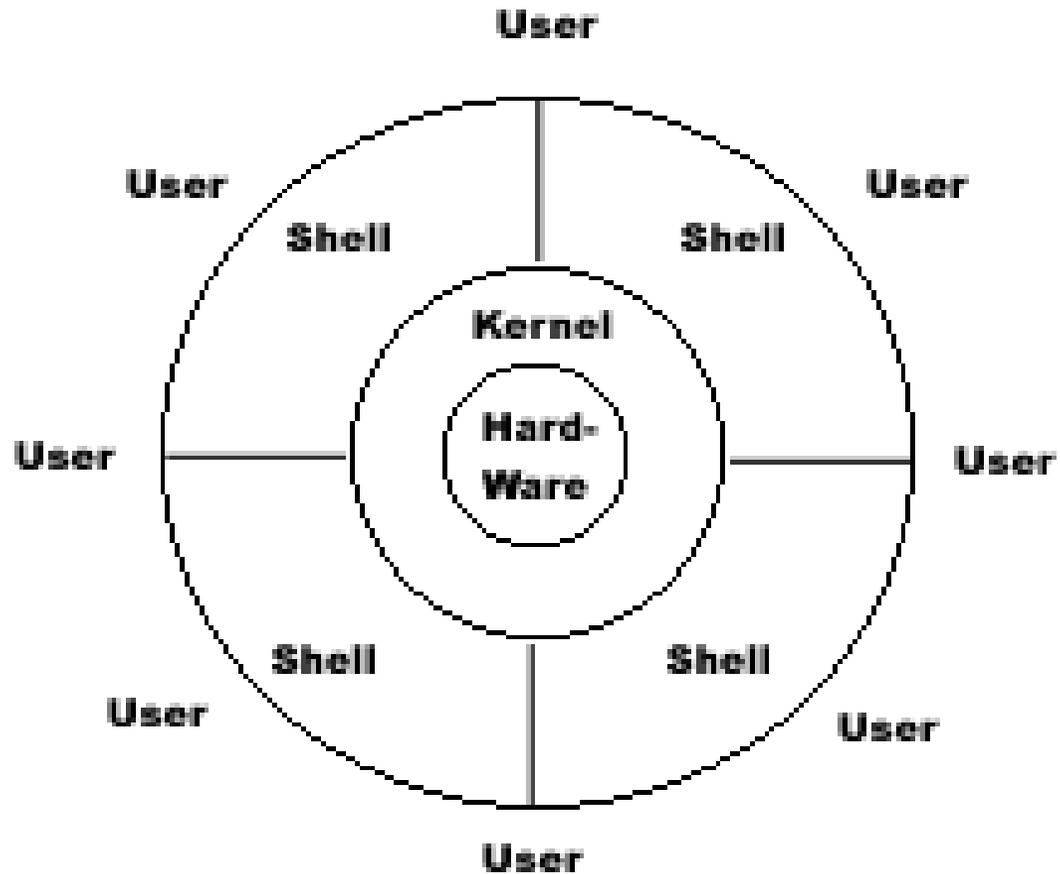
LINUX

- 하나의 운영체제 like Windows or Mac OS
- Linus Torvalds 가 1991년에 개발
- 다양한 종류의 리눅스: Ubuntu, Debian, RedHat, Fedora, etc.
- 가볍고 안정적!
- 서버로 많이 이용
- 대부분 무료로 배포



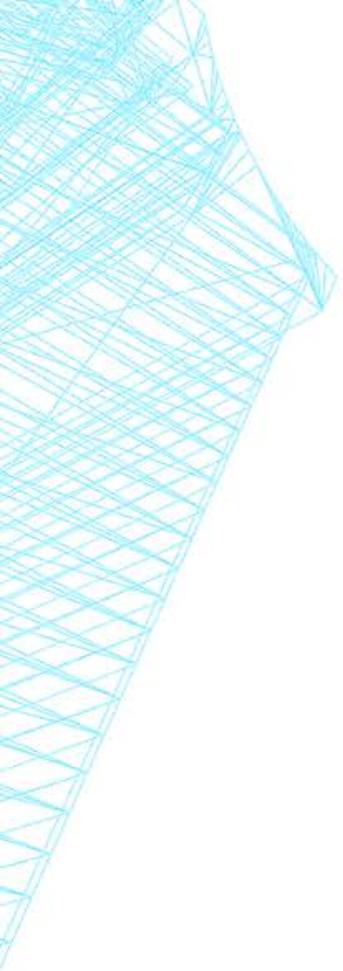
OPERATING SYSTEM

- 하드웨어와 유저간의 인터페이스
- 소프트웨어를 실행하기 위한 플랫폼을 제공
- 시스템, 하드웨어 관리



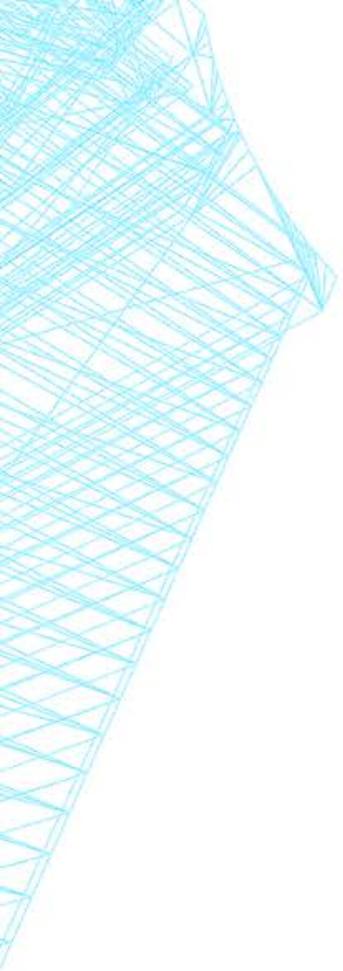
운영체제의 구조

- Kernel 커널
 - = 씨앗, 핵
 - 운영체제의 핵심 부분
 - 프로세스, 메모리, 파일 시스템, 디바이스, 네트워크 등 많은 것을 관리
 - 지속적으로 이용되어야 하므로 보호된 메모리를 이용
- Shell 셸
 - = 껍질, 껍데기
 - 커널을 둘러싼 껍데기
 - 사용자와 운영체제 사이의 층!
 - 명령어를 입력 받아 기계어로 바꾼 후 커널에 전달



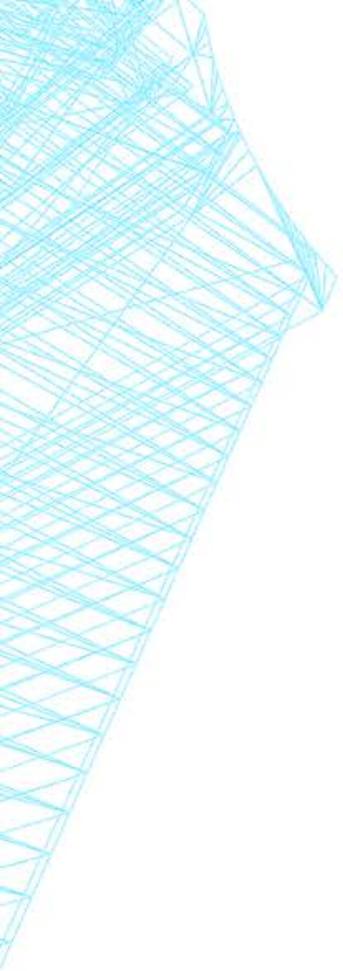
리눅스의 디렉토리 구조

- /bin
- /boot
- /dev
- /etc
- /home
- /media
- 등.



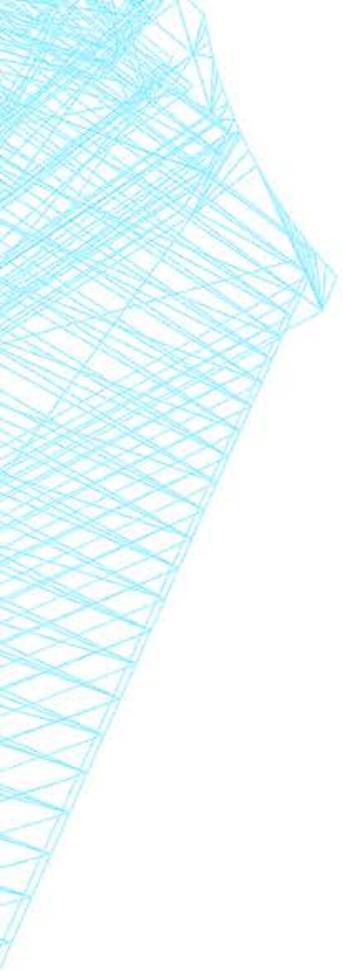
리눅스의 디렉토리 구조

- /bin
- /boot
- /dev
- /etc
- **/home - /home/[아이디] non-wheel**들은 여기서 작업
- /media
- 등.



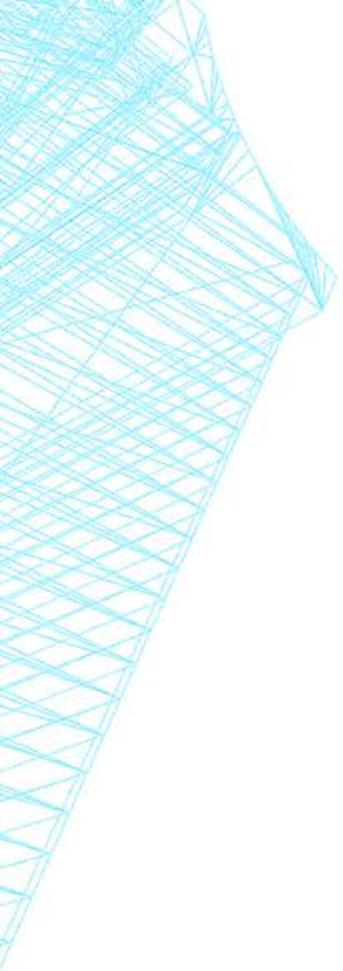
LINUX COMMANDS 명령어

- 계정 관련
 - passwd
- 매뉴얼
 - man
- 경로 관련
 - pwd
 - cd
 - ls
- 파일 관련
 - cp, rm, mv
 - mkdir, rmdir
 - tar
- 출력
 - cat, more, less
 - grep
 - vi
- 알리아스
 - alias
- 원격 접속
 - ssh, scp
- 파일 디스크립터
 - 와일드카드 *
 - 파이프
 - 리디렉션



passwd (password)

- 비밀번호 변경
- 사용법
 - passwd 를 친다.
 - 원래 비밀번호를 친다.
 - 새 비밀번호를 친다.
 - 끝



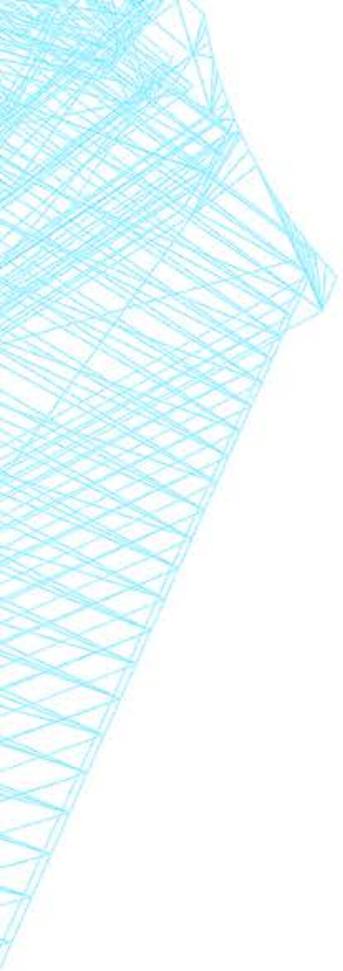
man (manual)

- 명령어 사용 설명서/매뉴얼
- 사용법: man [명령어]

경로 관련

- pwd (print **w**orking **d**irectory)
 - 현재 작업중인 디렉토리를 표시
- cd (c**h**ange **d**irectory)
 - 작업중인 디렉토리 변경
 - 사용법: cd [타겟 디렉토리]
 - . = 현재 디렉토리
 - .. = 상위 디렉토리
 - ~ = /home/[자기 아이디]
- ls (**l**ist)
 - 현재 작업중인 디렉토리의 **파일들을 나열**
 - 사용법: ls [옵션] [파일/디렉토리]

- 옵션:
 - -l (long)
 - 권한, owner, group 등 자세한 정보 표시
 - -a (all)
 - 숨겨진 파일까지 보여줌
 - 숨겨진 파일은 파일 이름이 .로 시작
 - -R (recursive)
 - 디렉토리 안의 디렉토리 안의 ... 디렉토리까지 전부 보여줌
 - -d (directory)
 - 디렉토리 내부를 리스팅하지 않고, 디렉토리 자체 정보 출력



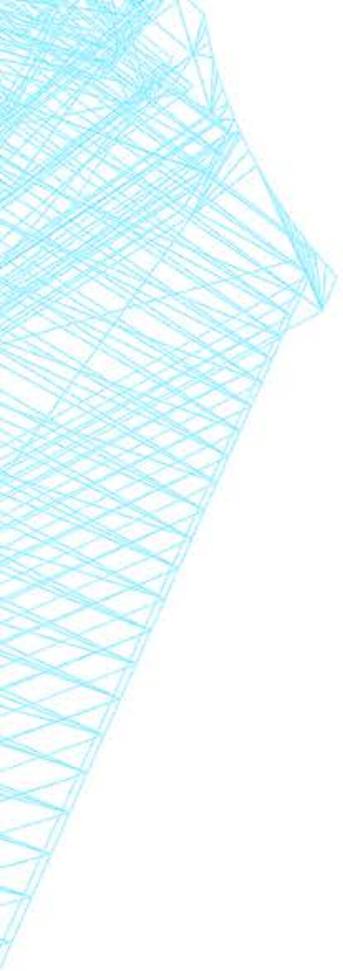
파일 관련

- cp (**copy**)
 - 파일을 복사
 - 사용법: cp [option] [source] [target]
 - 옵션:
 - -r (recursive): 하위파일까지 전부 복사
- mv (**move**)
 - 파일을 이동 or 잘라내기+붙이기
 - 사용법: mv [option] [source] [target]
- rm (**remove**)
 - 파일을 삭제
 - 사용법: rm [option] [target]
 - 옵션:
 - -r (recursive): 하위파일까지 전부 삭제
- mkdir (**make directory**)
 - 새 디렉토리 만들기
 - 사용법: mkdir [option] [target]
 - 옵션:
 - -p : 하위 디렉토리 줄줄이 생성
- rmdir (**remove directory**)
 - 비어있는 디렉토리 삭제
 - 사용법: rmdir [option] [target]
 - 옵션:
 - -p: 하위 디렉토리 모조리 삭제
 - 단, 모든 하위 디렉토리가 비어있어함

파일 관련

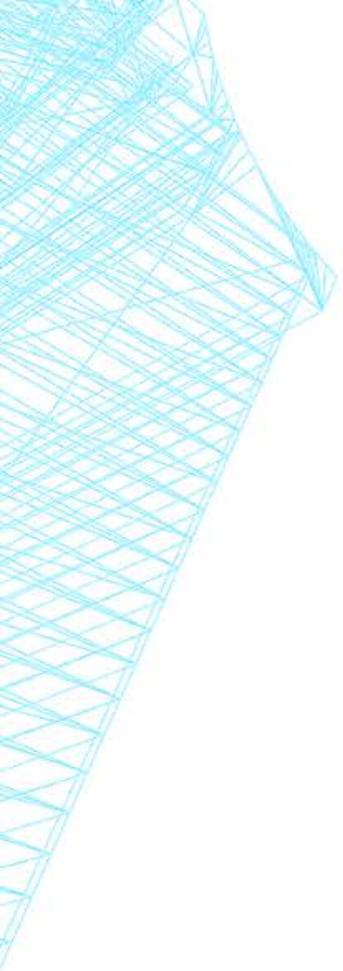
- tar
 - 여러 파일을 모으고 압축
 - 사용법: tar [cvxzf] filename file1 file2 ...
 - tar.gz - 파일을 모으고 압축까지 한번에
 - tar cvzf alzip.tar.gz a.a b.b c.c d.d
 - alzip.tar.gz 파일로 a.a b.b c.c d.d를 모으고 압축
 - tar.gz 풀기
 - gunzip alzip.tar.gz -> tar xvf alzip.tar
 - 를 한방에! tar xvzf alzip.tar.gz

- -c (create): 압축
- -v (verbose): 진행상황 화면에 보여주기
- -x (extract): 압축 풀기
- -z (gzip): gzip으로 압축/해제
- -f (file): 압축 결과를 파일로 저장



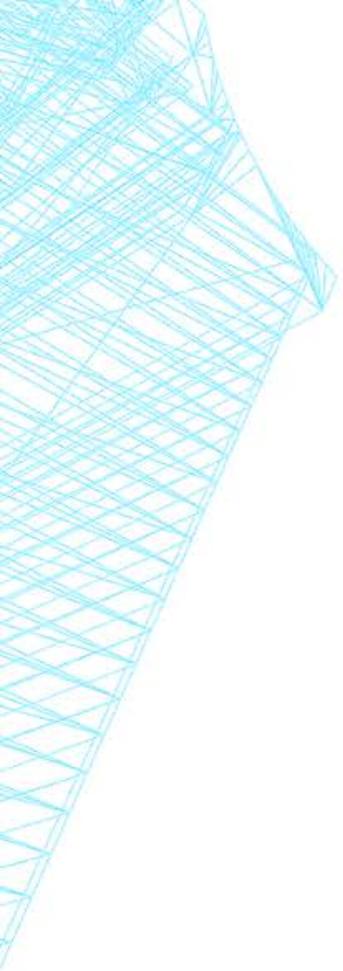
출력

- cat (con**cat**enate)
 - 파일 내용 읽기, 합치기할 때 씀
 - 사용법
 - cat [filename] - 파일 읽기
 - cat [a] [b] > [c]
 - a와 b의 파일 내용을 합친 것을 c로 저장
 - cat -n [filename]
 - 줄 번호가 나오게 파일 내용을 출력
- more
 - 내용을 앞에서부터 화면에 꼭 차게 출력
 - 엔터 - 한 줄씩 이동
 - 스페이스바 - 한 페이지씩 이동
 - q - 나가기
- less
 - more보다 더 많은 기능 지원
 - 사용법: less [filename]
 - /[keyword]: [keyword] 검색
 - n - 이전 키워드 찾기
 - N - 다음 키워드 찾기
 - :[line#]: [line#]번 줄로 이동
 - v - 편집
 - q - 나가기
- wc(**w**ord **c**ount)
 - 줄 수, 단어 수, 바이트 수를 세줌
 - 사용법: wc [filename]



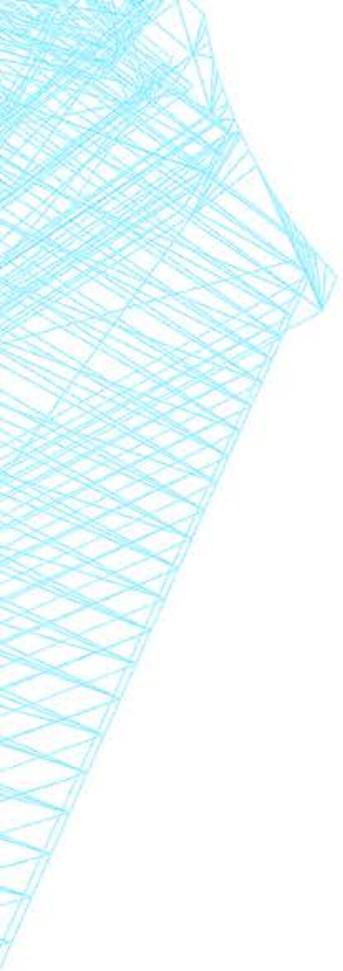
출력

- grep (global/regular expression/print)
 - 텍스트 검색 기능
 - 주어진 regular expression과 맞는 줄을 찾아 출력
 - grep [option] [keyword] [file]
 - 옵션:
 - -i: 대소문자 구분 X
 - -w: 독립적인 단어로 검색
- vi (vim)
 - 편집기/파일 생성(?)
 - vim 세미나에서!



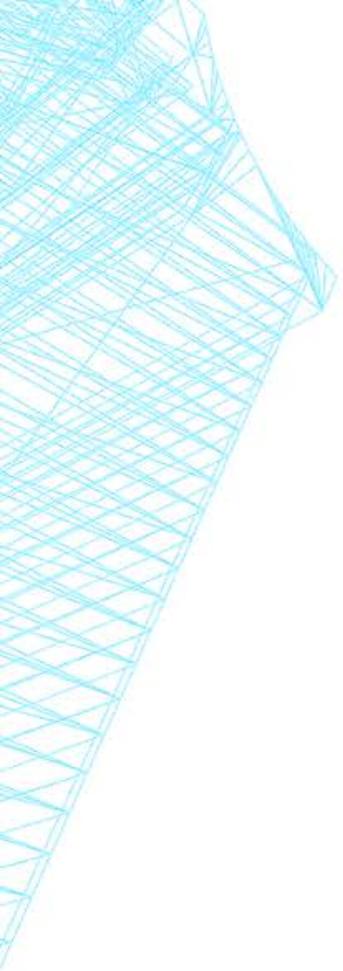
원격 접속

- ssh (secure shell)
 - 원격 컴퓨터에 접속하여 작업할 수 있게 해준다.
 - ssh [user@computer]
 - [user@]부분이 비어있으면 같은 아이디로 접근 시도
- scp (secure copy)
 - 암호화된 명령을 통해 파일을 전송
 - scp [file] [user@computer:directory]
 - [user@]부분이 비어있으면 같은 아이디로 접근 시도



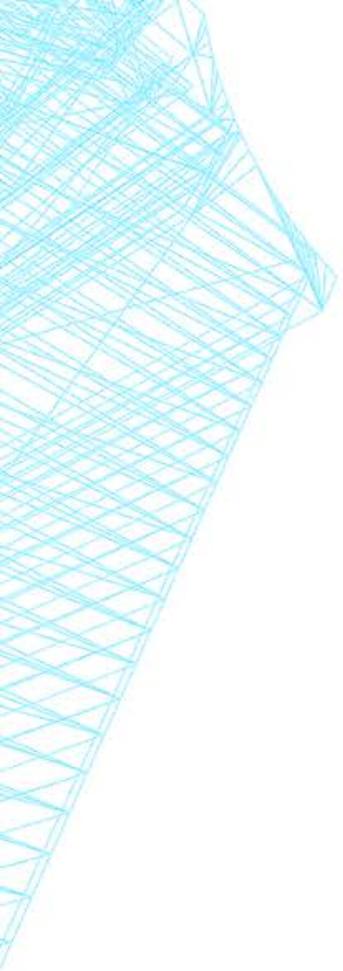
alias

- alias
 - 너무 길어 치기 귀찮은 명령어를 한방에!
 - 예시:
 - sparcs 서버에서 bit 서버로 접속하고 싶다!
 - “ssh chocho@bit.sparcs.org” 여러번치기 귀찮
 - alias bit= “ssh chocho@bit.sparcs.org”
 - 행복^~^



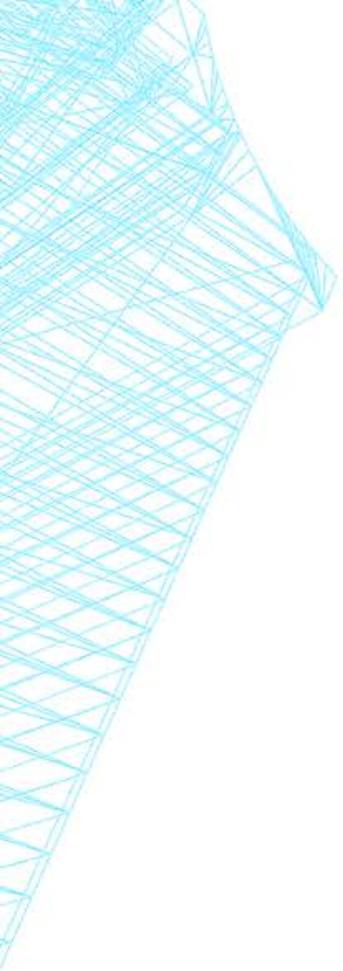
Wildcard *?

- *
- 0개 이상의 문자를 상징
- 예: cho*
 - cho, choi, chocho, chohyun, chohyunsung, ...
- ?
- 딱 1개의 문자를 상징
- 예: ?parcs
 - sparcs, oparcs, pparcs, ...



Wildcard *?

- *
- 0개 이상의 문자를 상징
- 예: cho*
 - cho, choi, chocho, chohyun, chohyunsung, ...
- ?
- 딱 1개의 문자를 상징
- 예: ?parcs
 - sparcs, oparcs, pparcs, ...



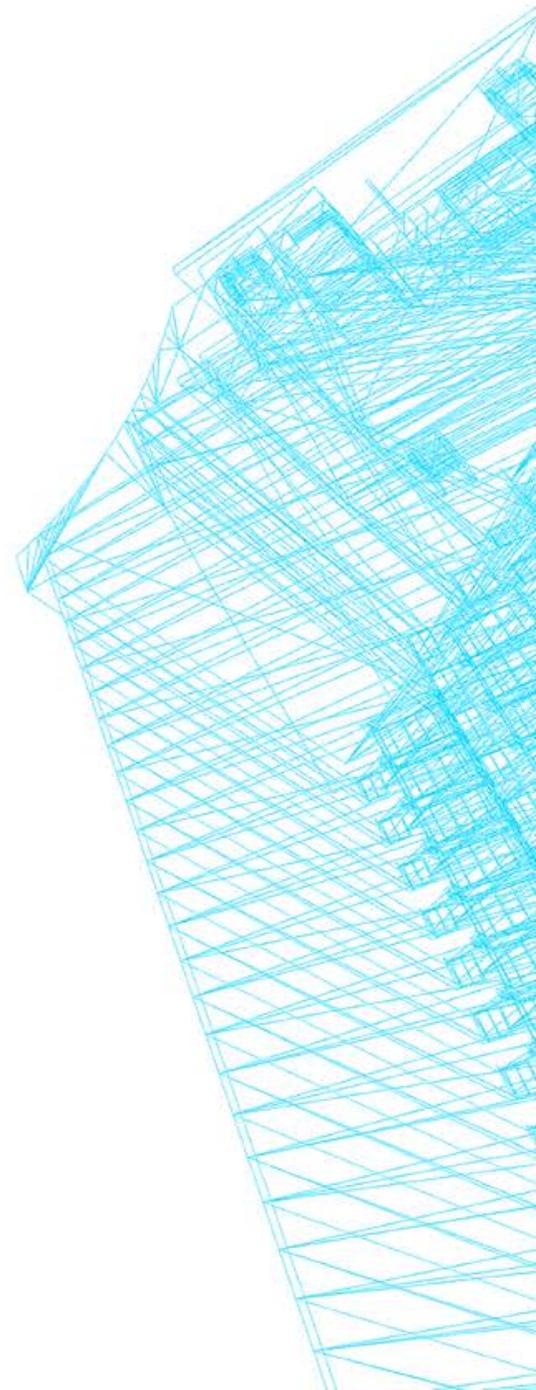
Pipe & Redirection

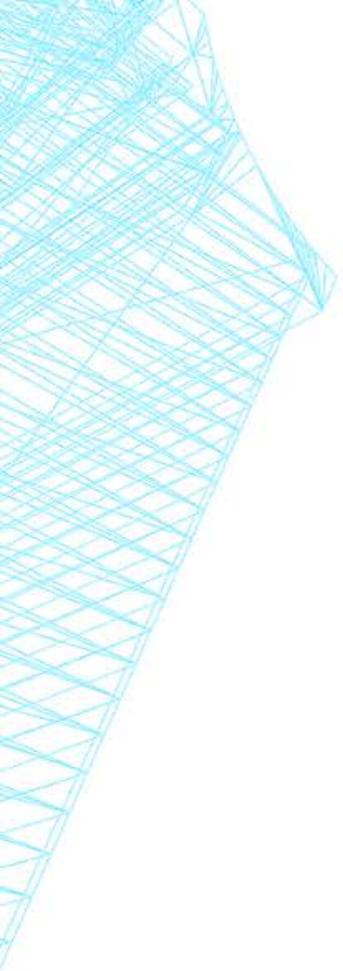
- Pipe |
 - `[command1] | [command2]`
 - `[command1]`의 output이 `[command2]`의 input으로 연결
 - 예: `ps aux | grep bash`
- Redirection >
 - redirect standard output
 - 예: `ls > list.txt`

LINUX

VIM

TOY





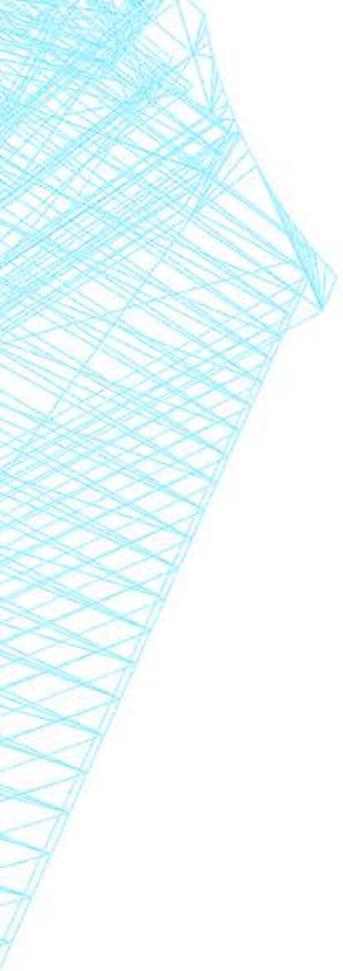
vim 이란?

- UNIX 환경의 대표적인 편집기 (editor)
- 대부분 Linux의 기본 에디터
- other editors: emacs, nano, ed
- 여러 OS에서 사용 가능
 - iOS, 안드로이드 등

세가지 모드

- 파일모드
- 명령모드
- 입력모드

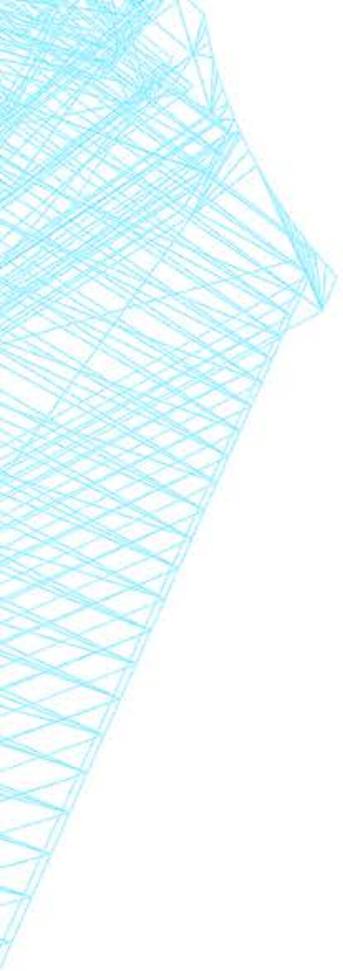




입력모드

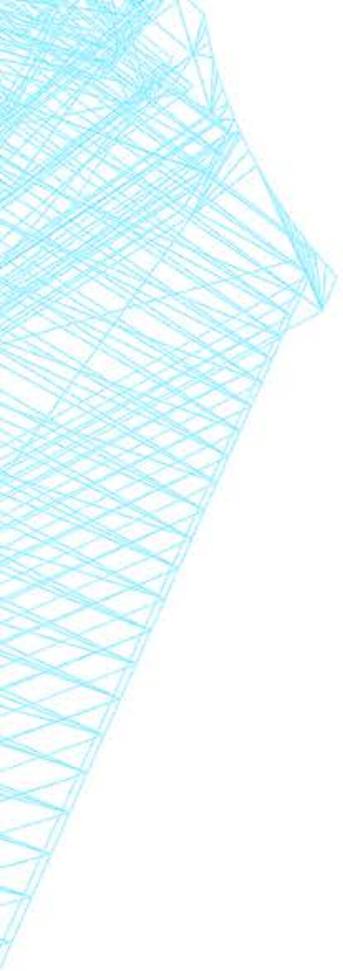
- 명령모드에서 i, a, o 등을 입력했을 때
- ESC키를 누르면 명령모드로 전환

- 내용 입력이 가능한 상태



입력 | 입력모드

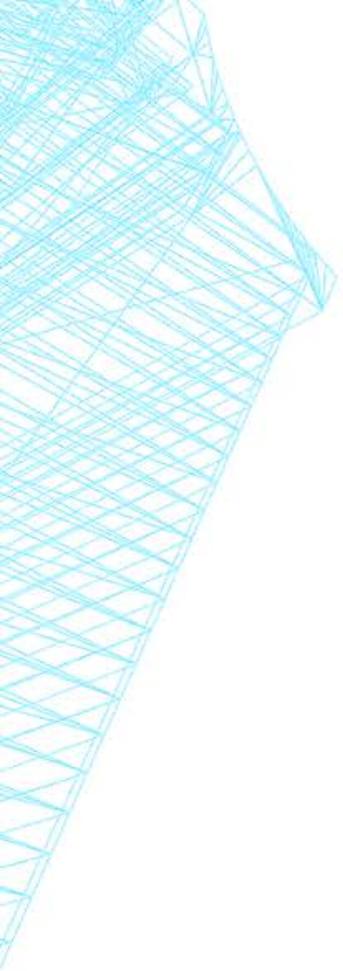
- i insert 현재 위치부터 입력
- a append 현재 위치 다음부터 입력
- I 현재 줄의 맨 앞에서부터 입력
- A 현재 줄의 맨 뒤에서부터 입력
- o open line 현재 줄 아래에 줄을 만들고 입력
- O 현재 줄 위에 줄을 만들고 입력



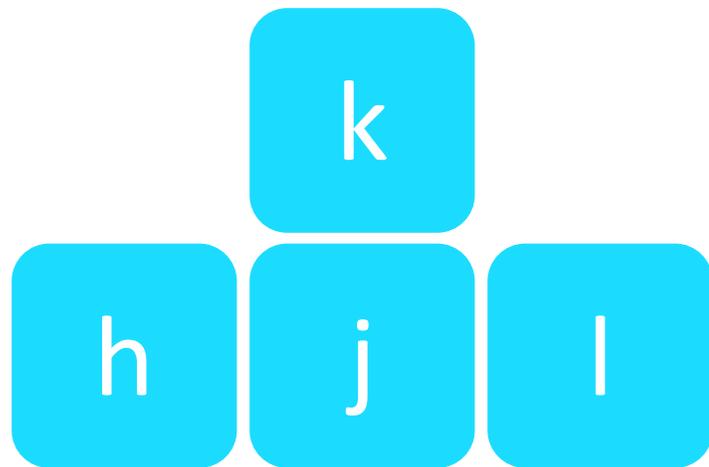
명령모드

- 처음 vim을 실행했을 때!
- 입력모드에서 ESC키를 눌렀을 때!

- 편집에 필요한 각종 명령이 가능한 상태
 - 이동, 수정, 삭제, 검색 등



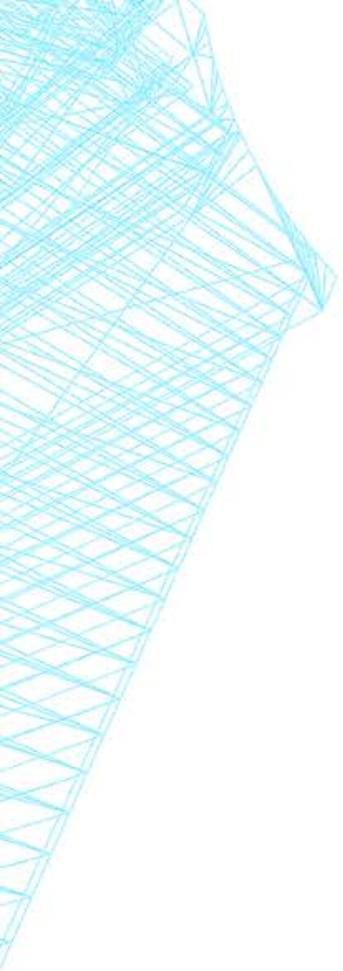
이동 | 명령모드



이동 | 명령모드

※ “의미“ 기준이라는 것은 “ ” . / % < > 등등을 기준으로 구분

- w : 다음 단어(의미 기준)의 첫 글자로 이동
- W : 다음 단어(공백 기준)의 첫 글자로 이동
- b : 이전 단어(의미 기준)의 첫 글자로 이동
- B : 이전 단어(공백 기준)의 첫 글자로 이동
- e : 다음 단어(의미 기준)의 끝 글자로 이동
- E : 다음 단어(공백 기준)의 끝 글자로 이동
- O : 현재 줄의 처음으로 이동
- ^ : 현재 줄의 첫 글자로 이동
- \$: 현재 줄의 마지막 글자로 이동
- +, <Enter> : 다음 줄의 첫 글자로 이동
- - : 이전 줄의 첫 글자로 이동
- () : 이전/다음 문단(빈 줄 포함)의 첫 글자로 이동
- { } : 이전/다음 문단의 첫 글자로 이동
- [,] : 이전/다음 함수({, })의 처음으로 이동
- H : 현재 화면의 맨 위로 커서 이동
- M : 현재 화면의 가운데로 커서 이동
- L : 현재 화면의 맨 아래로 커서 이동



이동 | 명령모드

- gg : 파일 맨 처음으로 이동
- [줄 번호] + gg : 해당 줄로 이동
- G : 파일 맨 마지막으로 이동
- **:[줄 번호] <Enter> 를 이용해도 해당 줄로 이동 가능**

삭제 | 명령모드

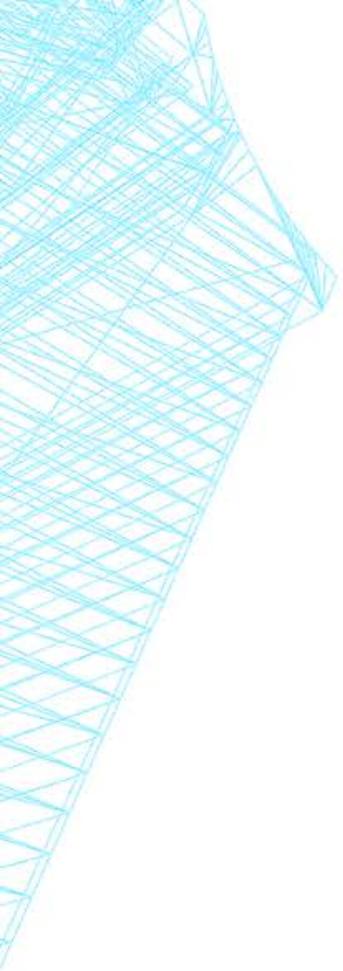
- x, dl : 현재 위치의 글자 삭제
- X, dh : 현재 위치 바로 앞의 글자 삭제
- d + <이동키> : 현재 위치부터 <이동키>까지 움직인 부분을 삭제
 - dw(단어), d0(처음까지), dj(그 줄과 다음 줄), dk
- D, d\$: 현재 위치부터 줄의 끝까지 삭제
- dd : 커서가 있는 줄을 삭제

편집 | 명령모드

- v : 글자 단위로 블록 지정, 이동명령어 이용
- V : 줄 단위로 블록 지정

블록 지정한 후,

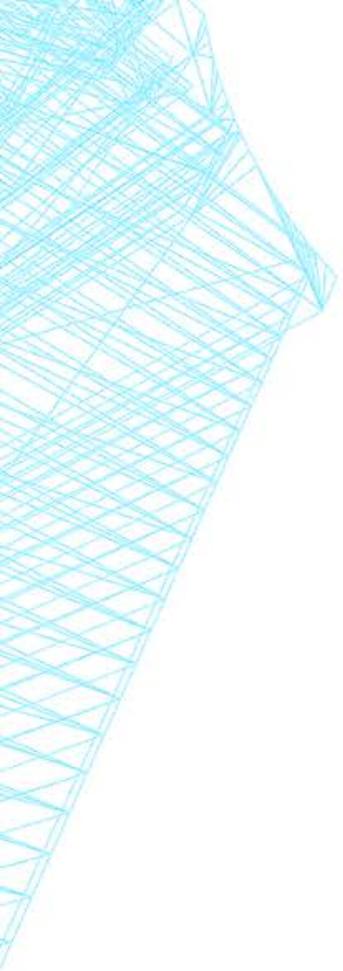
- x, d, c : 잘라내기 (클립보드에 저장)
- y + <이동키> : 현재 위치부터 <이동키>까지 움직인 부분을 복사
- yy : 현재 줄을 복사
- p : 현재 위치의 다음에 붙여넣기
- P : 현재 위치에 붙여넣기



스크롤 | 명령모드

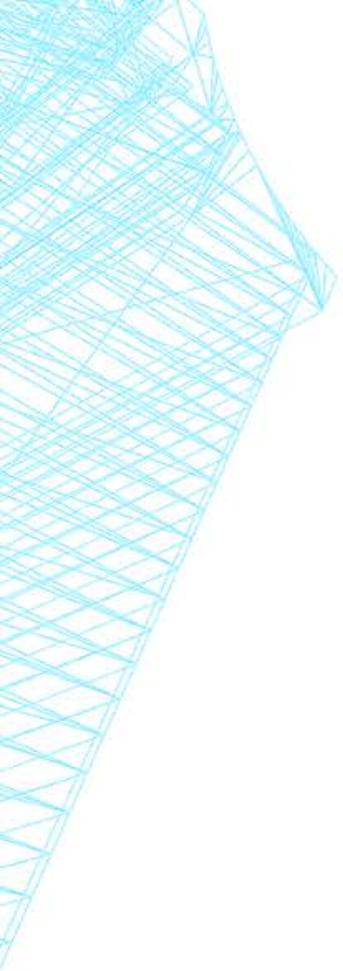
- Ctrl + F : 한 화면 다음으로 스크롤
- Ctrl + B : 한 화면 전으로 스크롤
- Ctrl + D : 반 화면 다음으로 스크롤
- Ctrl + U : 반 화면 전으로 스크롤
- Ctrl + E : 한 줄 다음으로 스크롤
- Ctrl + Y : 한 줄 전으로 스크롤

- z + <Enter>, ., - : 현재 줄을 화면의 맨 위, 가운데, 맨 아래로 스크롤



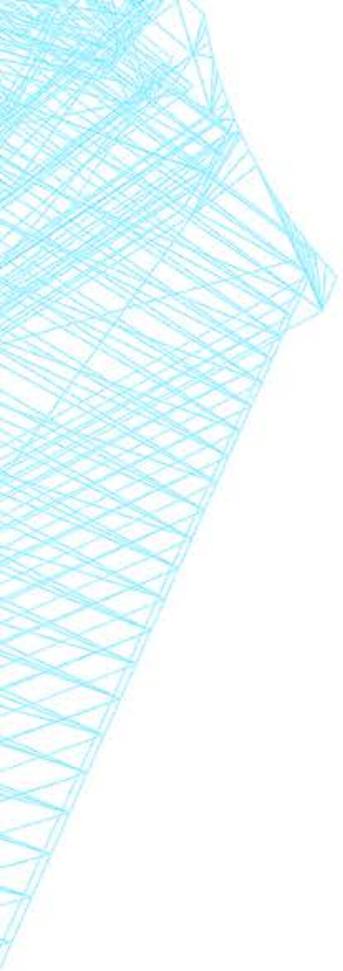
파일모드

- 명령모드에서 : 키 입력
- 파일 저장, 읽기, 각종 치환, vim 설정



파일 | 파일모드

- :w write file 파일 저장
- :w filename save as filename 다른 이름으로 저장
- :q exit 나가기
- :wq or :x save & exit 저장하고 나가기
- :q! exit, discard changes 저장하지 않고 나가기
- :help [명령어] ([명령어]에 대한) 도움말



검색/치환 | 파일모드

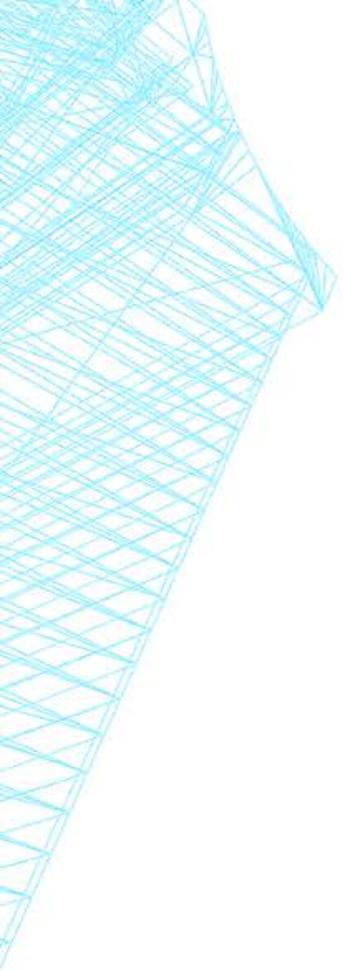
- `/[string]`
- `/[string]wc`
- `?[string]`
- `:s/[old]/[new]`

파일 내에서 [string] 검색

대소문자 구분 X

밑에서부터 검색

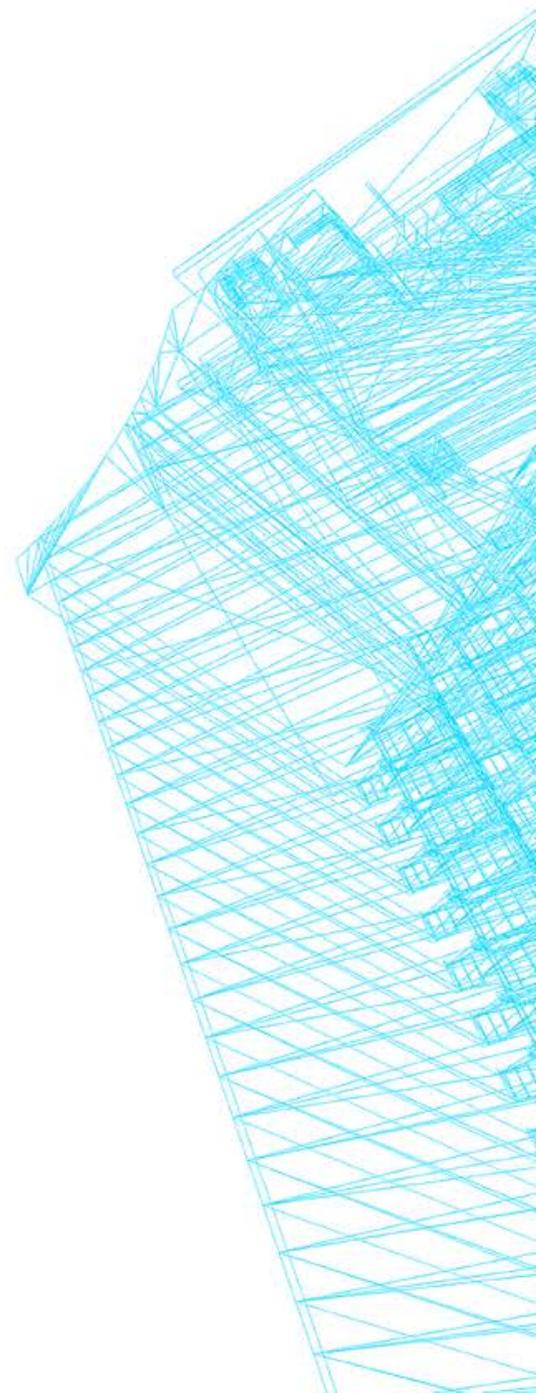
old 를 new 로 변경



more about VIM

- [vimtutor](#)
- sparcs.org/seminar
- [VIM adventures](#)

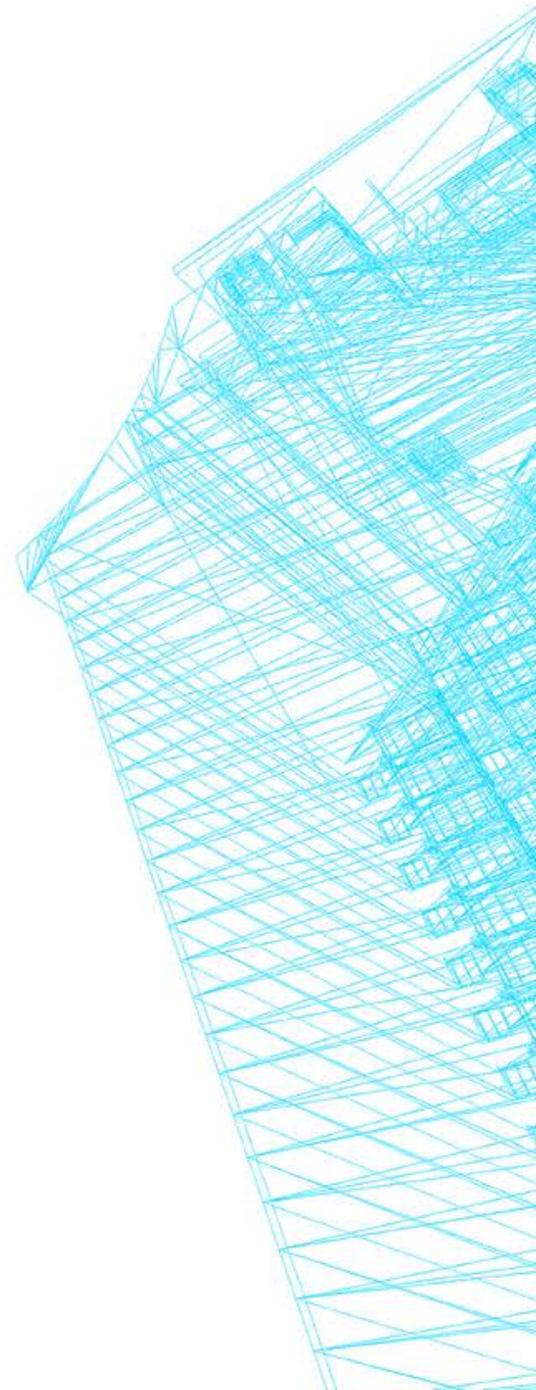
LINUX
VIM
TOY

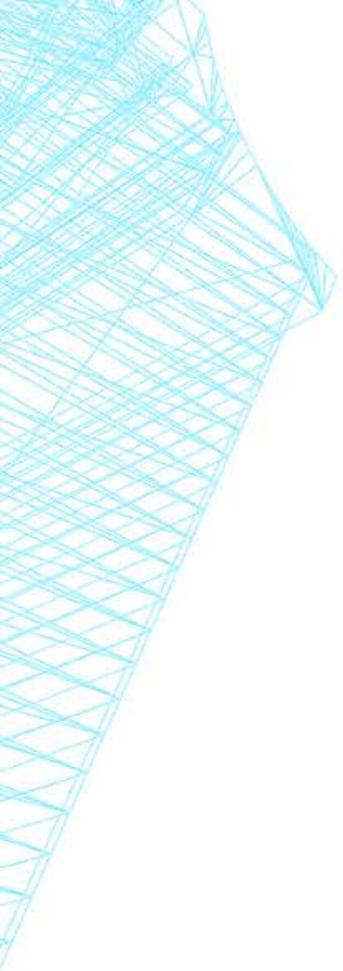


이동수(EGG) 회원의

2014 SPRING TOY SEMINAR

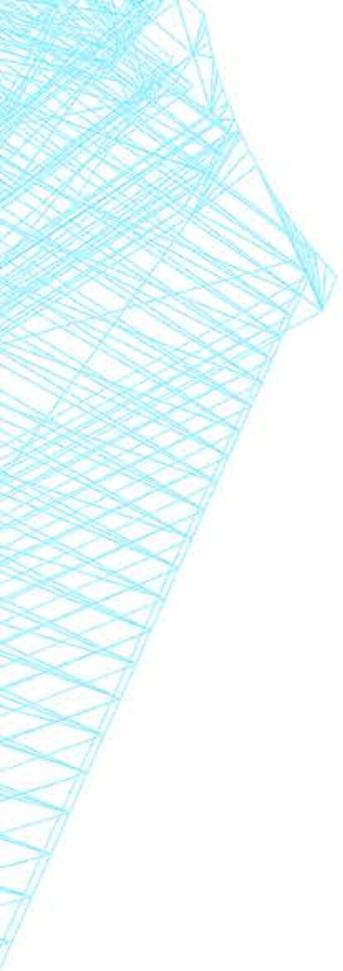
참조





숙제

1. nugu 를 사용해 자신을 제외한 스팍스 회원 5명의 이름, 전화번호, 이메일 획득!
2. 획득한 정보를 자신의 홈디렉토리 /home/[자기아이디]/h/e/l/l/o/w/o/r/l/d에 텍스트 파일로 저장 (vim 사용)
3. mailpatcher 를 사용해 이름 옆에 아이디도 덧붙이기
4. vim을 잘 활용해 같은 내용 한 줄 간격으로 5번 복사!



References

- 2014 가을 신입생 세미나 - 리눅스/장난감 by [suckzoo](#)
- 2014 봄학기 신입생 세미나 - TOY by [egg](#)
- 2013 봄 신입생 세미나 #4 Vim by [yumgyun](#)
- 2012 봄 신입생 세미나 - 4. VIM by [podo](#)