제05장

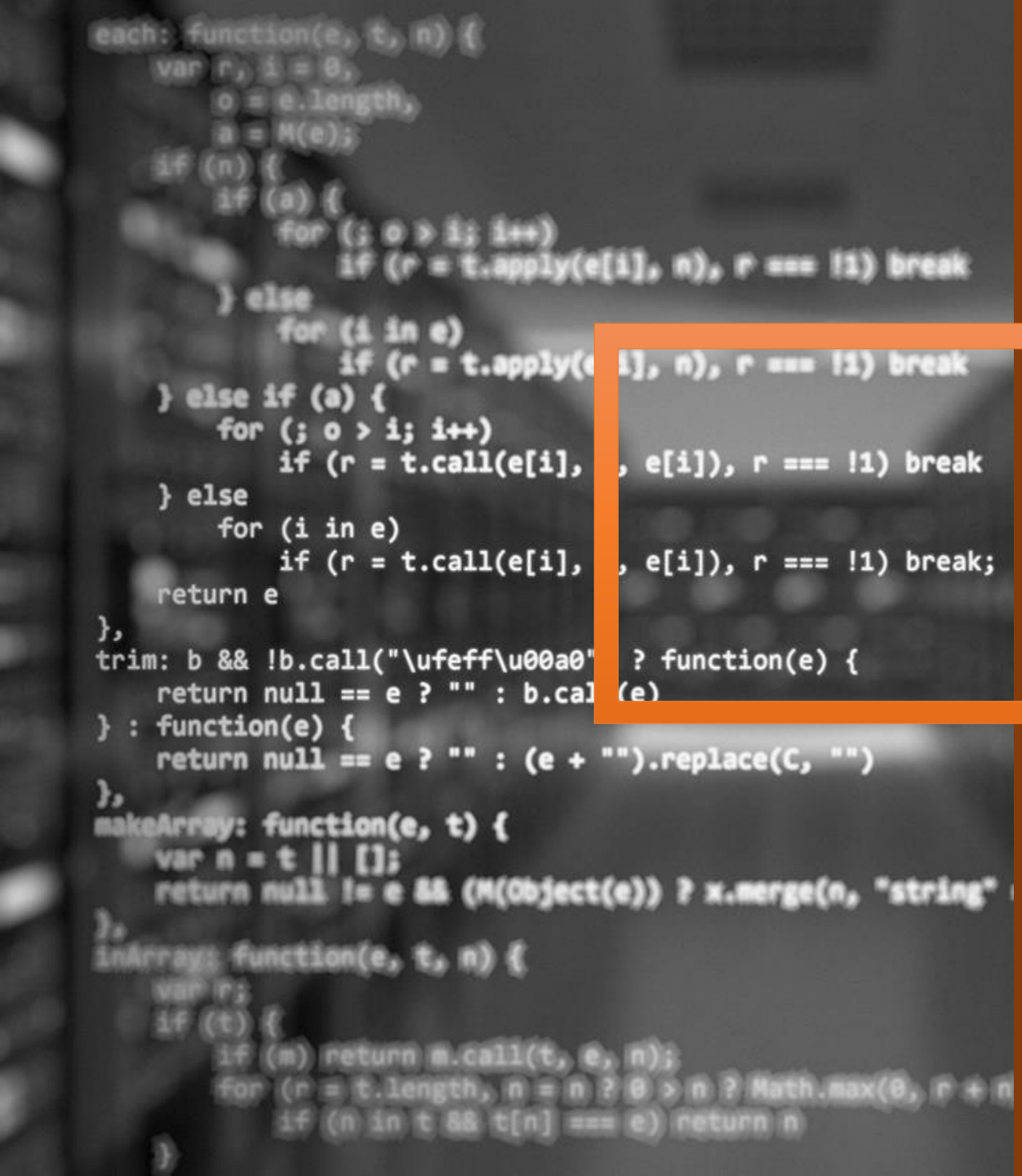
Ubuntu

DevOps

학습목표

1. Ubuntu 운영체제에 대해서 이해할 수 있다.
2. Ubuntu 운영체제의 명령어를 이해하고 사용할 수 있다.

목차



01. Ubuntu 운영체제

Linux

리눅스란?

- 1991년 리누스 토르발스(Linus Torvals)가 개발한 운영체제
- Unix 운영체제를 기반으로 한 운영체제
- 오픈 소스 운영체제
- 오픈 소스이기 때문에 수 많은 종류의 리눅스 배포판이 존재함

주요 리눅스 배포판

- RedHat 계열
  - 패키지 형식은 .rpm이고 패키지 관리자는 yum
  - 대부분 서버용으로 사용
  - CentOS, Amazon Linux, 타이젠 등
- Debian 계열
  - 패키지 형식은 .deb이고 패키지 관리자는 apt
  - 가장 많은 리눅스 배포판과 사용자 수를 가지고 있음
  - Ubuntu, 칼리 리눅스 등

Ubuntu

- **우분투 운영체제**
  - Debian 리눅스 기반으로 개발된 운영체제
  - 데스크탑 버전과 서버 버전이 모두 무료로 제공되는 오픈 소스 운영체제
  - 일반적으로 6개월마다 새로운 버전이 공개
  - 4월에 공개되면 x.04,10월에 공개되면 x.10
  - LTS(Long Term Service)는 2년마다 공개
  - .deb 패키지를 가지며 apt 패키지 관리자를 사용

계정

계정의 종류

- root 계정 : 모든 권한을 가진 특별한 계정
- 시스템 계정 : 리눅스 설치시 기본으로 생성되는 계정
- 사용자 계정 : 실제로 리눅스를 사용하는 계정

계정의 특징

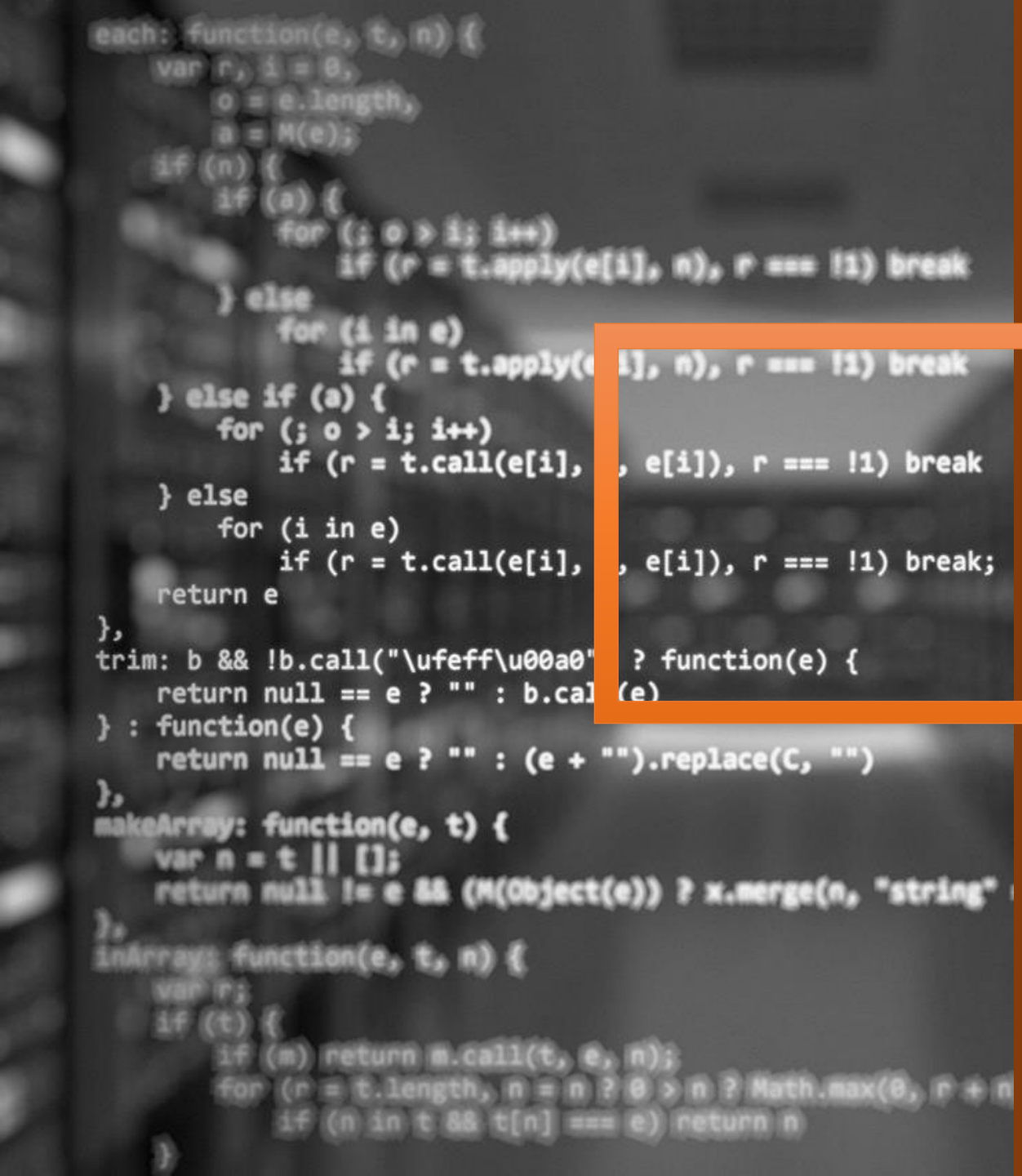
구분	프롬프트	특징
root 계정	#	모든 권한을 가진 계정 UID(User Identity)가 0인 슈퍼 유저 다른 계정을 생성하거나 권한을 부여할 수 있음
시스템 계정	로그인 불가	리눅스 시스템의 필요해 의해서 자동으로 생성되는 계정 adm, bin, daemon 등이 있음
사용자 계정	$	root 계정으로부터 권한을 부여 받아야 사용 가능한 계정 UID(User Identity)가 500~60000 사이인 일반 유저

리눅스 주요 디렉터리 구조

디렉터리 경로

- 절대 경로
  - / : 루트 디렉터리를 의미함
  - /로 시작하는 경로는 모두 절대경로를 의미함
  - /부터 전체 경로를 모두 작성해야 함

- 상대 경로
  - /로 시작하지 않는 경로는 모두 상대 경로를 의미함
  - 현재 디렉토리 기준으로 작성함
  - .. : 상위 디렉터리
  - . : 현재 디렉터리
  - ~ : 홈 디렉터리



02. Ubuntu 명령어

명령어 구조

- 명령어 구조

command [-option or --option] [argument1, argument2, ...]

명령어 옵션 인자

- 의미

1. 명령어 : 실행할 명령어 자체를 의미함
2. 옵션 : 명령어의 세부 기능
   ① 여러 개의 옵션을 함께 사용할 수 있음
   ② 대부분의 경우 옵션의 작성 순서는 없음
3. 인자 : 명령어가 필요로 하는 데이터, 파일, 디렉터리, 문자열 등을 의미함

- 예시

grep -i seoul → seoul을 대소문자 구분 없이(-i) 검색(grep)하시오.

명령어 옵션 인자

시스템 기본 명령어

- **su**
  - 로그아웃 없이 임시로 다른 사용자의 id를 사용하는 명령어
  - 주로 일반사용자로 로그인하여 잠시 동안 슈퍼유저 권한의 명령어를 실행할 때 사용
  - 사용자를 지정하지 않으면 root로 실행
  - 형식 : su [옵션] [사용자] [셸변수]

  - root로 변경하기
    - $ su -

시스템 기본 명령어

- timedatectl
  - 날짜, 시간, 타임존에 관한 확인 및 설정
  - 형식 : timedatectl [하위명령] [인자]

  - 날짜, 시간, 타임존 확인
    - $ timedatectl

  - 날짜 변경
    - $ timedatectl set-time "2025-01-01"

  - 날짜, 시간 변경
    - $ timedatectl set-time "2025-01-01 12:00:00"

  - 타임존 변경 (Asia/Seoul)
    - $ timedatectl set-timezone Asia/Seoul

시스템 기본 명령어

- passwd
  - 사용자의 비밀번호를 변경
  - 형식 : passwd [옵션] [사용자]

  - root 계정 비밀번호 변경하기
    - $ passwd root

시스템 기본 명령어

- pwd
  - 현재 작업 중인 경로를 확인
  - 형식 : pwd [옵션]

  - 현재 경로 확인
    - $ pwd

시스템 기본 명령어

- ps
  - 현재 실행 중인 프로세스의 목록 보기
  - 형식 : ps [옵션]

  - 실행 중인 프로세스의 자세한 정보 보기
    - $ ps -f

  - 다른 모든 사용자가 동작시킨 모든 프로세스 확인
    - $ ps -e

시스템 기본 명령어

- kill
  - 실행 중인 프로세스 종료
  - 형식 : kill [옵션] PID

  - PID가 766인 프로세스 종료
    - $ kill 766

  - PID가 766인 프로세스 강제 종료
    - $ kill -9 766

시스템 기본 명령어

- cd
  - 지정한 디렉터리로 이동
  - 형식 : cd 디렉터리명

  - 루트 디렉터리로 이동
    - $ cd /

  - 홈 디렉터리로 이동
    - $ cd ~

  - 상위 디렉터리로 이동
    - $ cd ..

시스템 기본 명령어

- mkdir
  - 새로운 디렉터리 만들기
  - 형식 : mkdir 디렉터리명

  - 현재 디렉터리에 data 디렉터리 만들기
    - $ mkdir data

시스템 기본 명령어

- rmdir
  - 비어 있는 디렉터리 삭제
  - 비어 있지 않은 디렉터리는 rmdir 명령으로 삭제 불가능 (rm -rf 명령으로 가능)
  - 형식 : rmdir 디렉터리명

  - 현재 디렉터리에 있는 data 디렉터리 삭제하기
    - $ rmdir data

시스템 기본 명령어

- ls
  - 지정한 디렉터리에 포함된 파일과 디렉터리 목록 보기
  - 형식 : ls [옵션] [디렉터리]

  - 모든 파일 보기(숨김 포함)
    - $ ls -a

  - 자세한 정보 출력
    - $ ls -l

  - 파일 끝에 파일 종류를 함께 출력(실행파일:*, 디렉터리:/, 일반파일:표시 없음)
    - $ ls -F

  - 하위 디렉터리까지 모두 출력
    - $ ls -R

시스템 기본 명령어

- rm
  - 지정한 파일을 삭제
  - 형식 : rm [옵션] 파일명

  - 디렉터리 삭제
    - $ rm -r DIR

  - 강제 삭제
    - $ rm -f FILE

  - 삭제 여부 확인
    - $ rm -i FILE

시스템 기본 명령어

- echo
  - 주어진 텍스트를 화면에 출력하는데 사용
  - 형식 : echo [옵션] 텍스트

  - Hello world 출력
    - $ echo Hello world

  - (Hello world) 출력 (특수문자 출력 시 큰 따옴표 필요)
    - $ echo "(Hello world")

  - 변수 선언 후 변수 값 출력
    - $ NAME=kim
    - $ echo $NAME

  - Hello world 파일 출력 (리다이렉션 : > 파일 덮어쓰기, >> 파일 이어쓰기)
    - $ echo Hello world > file.txt

시스템 기본 명령어

- cat
  - 텍스트 파일을 한 번에 보기
  - 형식 : cat [옵션] 파일명

  - 홈 디렉터리의 .profile 파일 열기
    - cat ~/.profile

시스템 기본 명령어

- more
  - 텍스트 파일을 한 화면씩 보기
  - 형식 : more [파일명]

  - 홈 디렉터리의 .profile 파일을 한 화면 단위로 보기
    - $ more ~/.profile

      space : 다음 화면을 보여 줌
      b : 이전 화면을 보여 줌
      enter : 다음 라인을 보여 줌
      /검색어 : 검색어를 찾아서 보여 줌

시스템 기본 명령어

- head
  - 텍스트 파일의 상단 부분 출력하기
  - 형식 : head [옵션] 파일명

  - 홈 디렉터리의 .profile 파일 위에서 10 라인만 보기
    - $ head ~/.profile

  - 홈 디렉터리의 .profile 파일 위에서 5 라인만 보기
    - $ head -5 ~/.profile

시스템 기본 명령어

- tail
  - 텍스트 파일의 하단 부분 출력하기
  - 형식 : tail [옵션] 파일명

  - 홈 디렉터리의 .profile 파일 아래에서 10 라인만 보기
    - $ tail ~/.profile

  - 홈 디렉터리의 .profile 파일 아래에서 5 라인만 보기
    - $ tail -5 ~/.profile

시스템 기본 명령어

- cp
  - 파일이나 디렉터리를 복사
  - 형식 : cp [옵션] 원본 복사본

  - 현재 디렉터리의 README.md 파일을 홈 디렉터리에 복사
    - $ cp README.md ~

  - 현재 디렉터리의 README.md 파일을 /app 디렉터리에 복사할 때 덮어쓰기 여부 묻기
    - $ cp -i README.md /app

  - /home 디렉터리를 하위 디렉터리까지 모두 포함하여 /tmp 디렉터리에 복사
    - $ cp -r /home /tmp

  - 현재 디렉터리의 README.md 파일을 /app 디렉터리에 강제로 복사 (기존 파일은 제거함)
    - $ cp -f README.md /app

시스템 기본 명령어

- mv
  - 파일이나 디렉터리를 이동
  - 파일이나 디렉터리의 이름을 변경
  - 형식 : mv [옵션] 원본 이동위치

  - 현재 디렉터리의 my.cnf 파일을 홈 디렉터리의 /app 디렉터리로 이동
    - $ mv my.cnf ~/app

  - 현재 디렉터리의 my.cnf 파일 이름을 my.ini 로 변경
    - $ mv my.cnf my.ini

시스템 기본 명령어

- grep
  - 지정한 패턴을 검색한 결과를 출력
  - 형식 : grep [옵션] 패턴 [대상]

  - 홈 디렉터리의 .bashrc 파일에서 'user' 검색하기
    - $ grep 'user' ~/.bashrc

  - 홈 디렉터리의 .bashrc 파일에서 대소문자 구분 없이 'user' 검색하기
    - $ grep -i 'user' ~/.bashrc

  - 홈 디렉터리의 .bashrc 파일에서 'user'를 제외하고 검색하기
    - $ grep -v 'user' ~/.bashrc

  - 홈 디렉터리의 .bashrc 파일에서 'user'와 'rm' 검색하기
    - $ grep -e 'user' -e 'rm' ~/.bashrc

시스템 기본 명령어

- 파이프 (|)
  - 먼저 실행한 명령의 출력 결과를 나중에 실행하는 명령에 입력으로 사용
  - 형식 : 명령 | 명령

  - 파일과 디렉터리 목록을 한 화면씩 출력하기
    - $ ls | more

  - 실행 중인 모든 프로세스 중에서 mysql 검색하기
    - $ ps -ef | grep 'mysql'

  - ~/.bashrc 파일을 열고 'alias' 검색하기
    - $ cat ~/.bashrc | grep 'alias'

시스템 기본 명령어

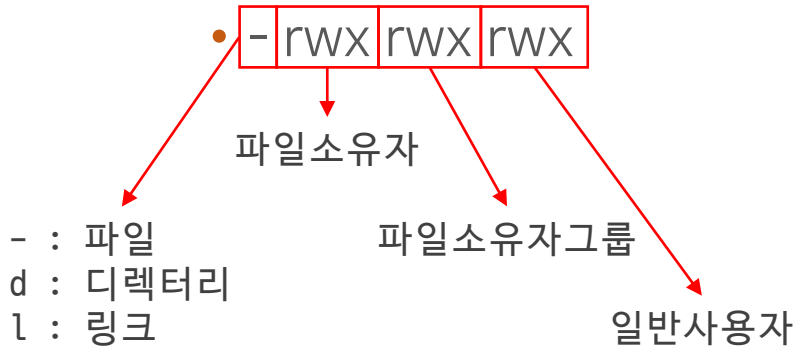
모드

- 모드 확인
  - $ ls - l

- 사용자 종류
  - 파일 소유자 (user)
  - 파일 소유자 그룹 (group)
  - 일반 사용자 (others)
  - 모든 사용자 (all)

- 권한 종류
  - 읽기 (read), 설정값 : 4
  - 쓰기 (write), 설정값 : 2
  - 실행 (execute), 설정값 : 1

모드 구성



시스템 기본 명령어

- chmod
  - 사용자별로 파일이나 디렉터리의 읽기/쓰기/실행 권한을 지정함
  - 형식 : chmod [옵션] [모드] [파일/디렉터리]

  - 파일 소유자에게 실행할 수 있는 권한 추가하기
    - $ chmod u+x FILE

  - 파일 소유자를 제외한 모든 사용자의 모든 권한 제거하기
    - $ chmod go-rwx FILE

  - 모든 사용자에게 모든 권한 설정하기
    - $ chmod a=rwx FILE
    - $ chmod rwxrwxrwx FILE
    - $ chmod 777 FILE

시스템 기본 명령어

- usermod
  - 사용자의 홈, 디렉터리, 그룹, UID, GID 등 사용자 정보를 변경
  - 형식 : usermod [옵션] 사용자계정

  - 사용자 user를 docker 그룹으로 변경
    - $ usermod -g docker user

  - 사용자 user를 docker 그룹에 추가로 속하게 함
    - $ usermod -aG docker user

시스템 기본 명령어

- chown
  - 파일이나 디렉터리의 소유자나 소유자 그룹을 변경할 수 있음
  - 형식 : chown [옵션] 소유자[:소유자그룹] [파일/디렉터리]

  - FILE의 소유자를 root로 변경하기
    - $ chwon root FILE

  - FILE의 소유자그룹을 root로 변경하기
    - $ chown :root FILE

  - FILE의 소유자와 소유자그룹을 root로 변경하기
    - $ chown root:root FILE

  - DIR와 모든 하위 디렉터리를 포함하여 소유자를 root로 변경하기
    - $ chown -R root DIR

시스템 기본 명령어

- chgrp
  - 파일이나 디렉터리의 사용자 그룹을 변경
  - 형식 : chgrp [옵션] 그룹 [파일/디렉터리]

  - FILE의 사용자 그룹을 min으로 변경하기
    - $ chgrp min FILE

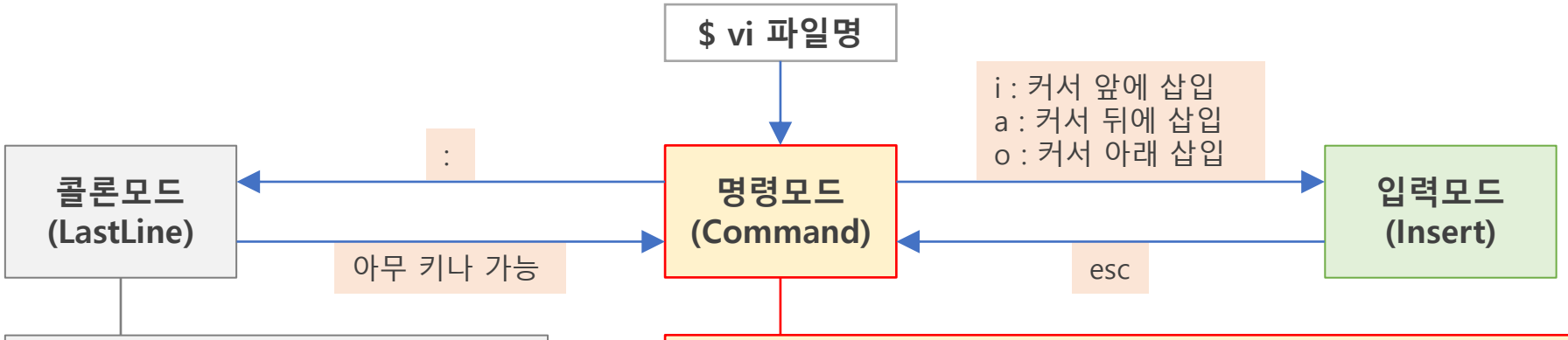
  - DIR와 모든 하위 디렉터리를 포함하여 사용자 그룹을 root로 변경하기
    - $ chgrp -R root DIR

시스템 기본 명령어

- vi
  - vi 편집기를 열어 파일을 보거나 새로 만들거나 기존 파일을 편집할 수 있음
  - 형식 : vi 파일명

  - .profile 파일 열기
    - $ vi ~/.profile

vi 에디터



:wq 저장하고 닫기
:w 새파일명 다른 이름으로 저장
:! ls 저장 여부 확인 위해
shell 명령 ls 입력
:q 그냥 닫기
:q! 저장 안하고 닫기
:r 읽을파일 현재 커서 아래에
파일 내용 붙이기
:set nu 행 번호 보기(set number)
vi 닫으면 설정 없어짐

1. 커서이동
   k, j : 상, 하
   h, l : 좌, 우
   ^, $: 라인 처음, 라인 끝
   w, b : 다음 단어, 이전 단어
   G : 마지막 행

2. 취소(undo)
   u (무한으로 사용 가능)

3. 삭제(delete)
   (삭제된걸 붙이기 가능)
   x : 글자 삭제
   dw : 단어 삭제
   dd : 행 삭제
   d^ : 현재 행 처음까지 삭제
   d$: 현재 행 끝까지 삭제
   dG : 문서 끝까지 삭제

4. 붙이기(paste)
   p, P : 커서 뒤에, 앞에 붙이기

5. 복사
   yw : 단어 복사
   yy : 행 복사
   y^ : 현재 행 처음까지 복사
   y$: 현재 행 끝까지 복사
   yG : 문서 끝까지 복사

6. 수정
   r : 한 글자 수정(replace)
   cw : 한 단어 수정(change word)
   cc : 한 행 수정
   c^ : 현재 행 처음까지 수정
   c$: 현재 행 끝까지 수정
   cG : 문서 끝까지 수정

7. 검색
   /검색어 : 검색어 찾기
   n : 다음 찾기(next)
   N : 이전 찾기

시스템 기본 명령어

- source
  - 스크립트 파일의 수정 사항을 바로 적용
  - 형식 : source [파일명]

  - .profile 파일의 변경사항을 재로그인 없이 곧바로 적용하기
    - $ source ~/.profile

시스템 기본 명령어

- dpkg
  - .deb 패키지의 설치, 삭제, 정보 제공 (패키지 설치 시 의존 문제를 해결하지 않음)
  - 형식 : dpkg [옵션] [패키지]

  - 설치된 패키지 목록 확인
    - $ dpkg --list

  - test 패키지 설치 (설치 시에는 .deb 확장자 명시)
    - $ dpkg -i test.deb

  - test 패키지 삭제
    - $ dpkg -r test

  - test 패키지 상태 확인
    - $ dpkg -l test

시스템 기본 명령어

- apt
  - .deb 패키지의 설치, 삭제, 정보 제공 (패키지 설치 시 의존 문제를 해결함)
  - 형식 : apt [옵션] 하위명령 [패키지]

  - 패키지 목록 업데이트 (/etc/apt/sources.list 파일 업데이트)
    - $ apt update

  - test 패키지 설치 (-y 옵션으로 자동으로 yes 입력)
    - $ apt install test

  - 이름에 test가 포함된 패키지 검색
    - $ apt search test

  - test 패키지와 설정파일 모두 삭제 (apt remove는 패키지만 삭제하고 설정파일은 남김)
    - $ apt purge test

시스템 기본 명령어

- tar
  - 여러 개의 파일을 하나의 파일로 묶거나 풀 때 활용
  - 형식 : tar [옵션] [파일]

  - 아카이브 파일 생성 (묶기)
    - $ tar -cvf 아카이브

  - 아카이브 파일 해제 (풀기)
    - $ tar -xvf 아카이브

  - 아카이브 파일 종류에 따른 옵션 변경

확장자	추가 옵션	예시
gzip	-z	-cvzf, -xvzf 등
bzip2	-j	-cvjf, -xvjf 등
xz	-J	-cvJf, -xvJf 등

시스템 기본 명령어

- tar 옵션
  - -f : 대상 아카이브 지정 (필수 옵션)
  - -c : tar 아카이브 생성 (파일 묶을 때)
  - -x : tar 아카이브에서 파일 추출 (파일 풀 때)
  - -v : 처리 과정을 자세하게 나열
  - -z : gzip 파일 적용 옵션
  - -j : bzip2 파일 적용 옵션
  - -C : 대상 디렉터리 경로 지정

시스템 기본 명령어

- wget
  - 웹에서 파일 다운로드를 수행
  - 형식 : wget [옵션] [URL]

  - http 프로토콜 test.tar.gz 다운로드
    - $ wget http://example.org/test.tar.gz

  - https 프로토콜 test.tar.gz 다운로드 (인증서 유효성 검사 무시)
    - $ wget --no-check-certificate https://example.org/test.tar.gz

  - test.tar.gz 백그라운드에서 다운로드
    - $ wget -b http://example.org/test.tar.gz

시스템 기본 명령어

- curl
  - 클라이언트와 서버가 URL을 이용하여 데이터를 송수신함 (Client URL)
  - 형식 : curl [옵션] [URL]

  - GET 요청 (-X GET 생략 가능)
    - $ curl http://localhost:8080/users

  - POST 요청 (JSON 전송)
    - $ curl -d '{"key": "value"}' -H "Content-Type: application/json" ₩
      -X POST http://localhost:8080/users

  - DELETE 요청
    - $ curl -X DELETE http://localhost:8080/users/1

  - PUT 요청 (파라미터 전송)
    - $ curl -d 'param1=value2&param2=value2' -H "Content-Type: application/json" ₩
      -X PUT http://localhost:8080/users/1