



week3



1. 깃 설치 ⇒ 딱히 공부할 거 없었다

=====

2. 깃 첫 설치 후 해야 할 설정

=====

3. visual studio code 와 리눅스 명령

-vs 개요와 설치

-리눅스 기초 명령: pwd, ls, cat, echo, touch, cp, rm, mv

-리눅스 기초 문자 : 파이프(|), 리다이렉트 : > >> <

2. 깃 설정

Summary

» 깃 설정 명령 구조

- ◆ \$ git config --설정범위 설정변수 설정값
- ◆ 설정 범위
 - [--system | --global | --local]
- ◆ --global 인 경우 설정 파일
 - C:\Users\[사용자-계정]\.gitconfig

Summary

» 지역 저장소 생성 전 다음 설정 반드시 수행

- ◆ 사용자 설정

- \$ git config --global user.name hskang
- \$ git config --global user.email hsakng@gmail.com

- ◆ 맥(lf)과 윈도(crlf) 간의 자동 변환

- \$ git config --global core.autocrlf true

- ◆ 뉴라인 경고 발생 없애기(옵션)

- \$ git config --global core.safecrlf false

- ◆ 기본 편집기 설정

- \$ git config --global core.editor 'code --wait'

- ◆ 깃 저장소 기본 브랜치 이름 설정

- \$ git config --global init.defaultBranch main

Summary

» git repository 생성 방법

- ◆ \$ git init

- ◆ \$ git init .
- 현재 디렉토리를 git repository로 만들기 위해서 사용

- ◆ \$ git init basic
- 현재 폴더 하부에 폴더 basic을 생성하고 git repository로 만들기 위해서 사용

3. vscode 와 리눅스

Summary

» Vscode 개요와 설치

- ◆ 깃 버전관리와 파일 관리
- ◆ 시스템 버전 설치 권장
 - 탐색기에서 파일 또는 디렉토리에서 바로 'vscode로 열기' 메뉴 체크
- ◆ 활동바(툴바) 활용

» 리눅스 명령

- ◆ 깃 CLI에서 함께 명령 사용
 - \$ pwd
 - \$ ls -al
 - \$ touch a.txt
 - \$ echo aaa > a.txt
 - \$ cat a.txt

깃 설정

배시에서 처음 하는 것 : 사용자의 이름과 전자 메일 설정

⇒kimsinyoung / dpqlc7008@naver.com

설정 명령과 설정 범위

설정 명령 구조

- \$ git config --설정범위 설정변수 설정값

설정 범위

- [--system | --global | --local]

▪ system

▪ global

▪ local

▶ 모든 사용자

▶ 현재 사용자의 모든 저장소

▶ 현재 사용자의 현재 저장소

설정 명령 사용

```
PC@DESKTOP-482NOAB MINGW64 /c/[smart Git]
```

```
$ git config --global user.name ai7dnn
```

```
PC@DESKTOP-482NOAB MINGW64 /c/[smart Git]
```

```
$ git config --global user.email ai7dnn@gmail.com
```

```
PC@DESKTOP-482NOAB MINGW64 /c/[smart Git]
```

```
$ git config --global init.defaultBranch main
```

전역 설정

줄바꿈 설정



저장소 생성 전에 반드시 설정 확인 후 시작

✓ 다음 설정 명령

- 자신이 사용하는 컴퓨터라면 1회 설정 이후에 설정 파일에 저장
 - 수정하지 않는다면 이후 다시 설정할 필요 없음

주요 6가지 설정

\$ git config --global user.name ai7dnn	사용자 이름
\$ git config --global user.email ai7dnn@gmail.com	사용자 전자메일
\$ git config --global core.autocrlf true	줄바꿈 자동변환
\$ git config --global core.safecrlf false	줄바꿈 안전 설정
\$ git config --global core.editor 'code --wait'	기본 편집기 설정
\$ git config --global init.defaultBranch main	기본 브랜치 이름

저장소 생성

현재 있는 디렉토리를 깃 저장소(git repository)로 만들기 위해 사용

⇒ git init / git init .

현재 폴더 하부에 폴더 basic을 생성하고 git Repositoryh 로 만들기 위해서 사용 ⇒ git init basic

git init basic

- 현재 폴더 하부에 폴더 basic을 생성하고 git repository로 만들기 위해서 사용

PC@DESKTOP-482NOAB MINGW64 /c/[smart Git]

\$ git init basic

Initialized empty Git repository in c:/[smart Git]/basic/.git/



저장소 생성 확인

- ✓ 생성된 깃 저장소 폴더 basic으로 이동

- 하부에 폴더 .git를 확인

```
PC@DESKTOP-482NOAB MINGW64 /c/[smart Git]
$ cd basic                                         현재 브랜치(색상)

PC@DESKTOP-482NOAB MINGW64 /c/[smart Git]/basic (main) ⚡
$ ls -al
total 8
drwxr-xr-x 1 PC 197121 0 Dec 12 16:43 ./          현재 디렉토리
drwxr-xr-x 1 PC 197121 0 Dec 12 16:43 ../         부모 디렉토리
drwxr-xr-x 1 PC 197121 0 Dec 12 16:43 .git/        숨김으로 표시
                                                        되는 폴더
```

vscode와 리눅스 명령어

06. 비주얼 스튜디오 코드와 리눅스 명령

2

리눅스 명령

긱 실습을 하기 위한 리눅스 명령어 준비 (1/3)

폴더 관련 명령

- \$ pwd
 - Print Working Directory
 - 현재 폴더 표시
- \$ cd
 - Change Directory
- \$ mkdir dname
 - Make Directory
- \$ ls
 - File or folder list
 - 다양한 옵션
 - -l: 파일의 상세정보
 - -a: 숨김 파일 표시
 - -al: 위 두개를 조합

2 리눅스 명령

깃 실습을 하기 위한 리눅스 명령어 준비 (2/3)

파일 관련 명령: 생성

- \$ touch fname
 - 빈 파일 fname 생성

touch a b c

표준 출력으로 결과 표시 echo

- \$ echo git bash
- \$ echo 'print()'
- 이후 문자열을 컴퓨터 터미널에 출력(display)하는 명령어
- 단순한 문자열은 가능하나 괄호 등이 있는 경우, 작은 또는 큰 따옴표 사용

파일 a, b, c



싱글쿼트나 더블 쿼트를 쓰면 print명령어가 먹는다. 그냥 쓰면 그냥 print라고 echo가 작동한다.

깃 실습을 하기 위한 리눅스 명령어 준비 (2/3)

파일 관련 명령: 내용 보이기, 복사, 이름 바꾸기

- \$ cat fname
 - 파일 내용 보기
- \$ cp a b
 - 파일 a를 b로 복사
- \$ mv f1 f2
 - 파일 f1을 f2로 이름 수정

cat a b c



깃 실습을 하기 위한 리눅스 명령어 준비 (2/3)

✓ 파일 및 폴더 삭제

- \$ rm fname

- 파일 fname 삭제

- \$ rm -rf dname

- 하부에 서브폴더와 파일이 있어도 폴더 삭제, 옵션 사용

- -f : 강제로 파일이나 디렉토리를 삭제

- -r : 디렉토리 내부의 모든 내용을 삭제



명령어 ls

✓ 현재 위치의 파일 목록 조회

list

- ls -l

- 파일의 상세정보

- ls -a

- 숨김 파일 표시

- ls -t

- 파일들을 생성 시간 순(제일 최신 것부터)으로 표시

- ls -rt

- 파일들을 생성 시간 순(제일 오래된 것부터)으로 표시

- ls -f

- 파일 표시 시 마지막 유형에 나타내는 파일명을 끝에 표시

- '/' : 디렉터리

- '*' : 실행파일

- '@' : 링크 등등

2 리눅스 명령

명령어 cat

cat (Concatenate)

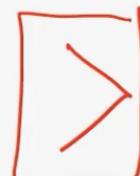
- 파일의 내용을 화면에 출력
- cat file1
 - file1의 내용을 출력
- cat file1 file2
 - file1과 file2의 내용을 출력
- 파이프(pipe) 기호인 | (vertical var)의 의미
 - 앞 명령의 표준 출력을 뒤 명령의 표준입력으로 처리
- cat file1 file2 | more
 - file1과 file2의 내용을 페이지별로 출력
- cat file1 file2 | head
 - file1과 file2의 내용을 처음부터 10번째 줄까지만 출력
- cat file1 file2 | tail
 - file1과 file2의 내용을 끝에서부터 10번째 줄까지만 출력



Redirection 명령어 연산 > >> <

'>', '>>' : 화면의 출력 결과를 파일로 저장

- '>' 기호
 - 기존에 있는 파일 내용을 지우고 저장
- '>>' 기호
 - 기존 파일 내용 뒤에 덧붙여서 저장
- '<' 기호
 - 파일의 데이터를 명령에 입력



 **Redirection 명령어 연산 >><** **활용 사례**

- `echo aaa>a.txt`
 - (새) 파일 a.txt에 문자열 aaa 복사(replace)
- `echo bbb>>a.txt`
 - (새) 파일 a.txt에 문자열 bbb 추가
- `cat file1 firle2 > file3`
 - file1, file2의 명령 결과를 합쳐서 file3라는 파일에 저장
- `cat file4 >> file3`
 - file3에 file4의 내용 추가
- `cat < file1`
 - file1의 결과 출력