

# Git

Source Code Management(SCM) Tool: 코드 관리 도구

Version Control System(VCS): 버전 관리 시스템

버전을 통해 코드를 관리하는 도구

## 버전 관리를 하는 이유

1. 언제든지 과거로 돌아갈 수 있음
2. 수정 이력을 확인할 수 있음

## Git 명령어

### 0. 폴더 생성

git 저장소로 만들 폴더 생성

#### 1. `git init`

git 프로젝트(저장소)를 시작(폴더 단위)

```
Initialized empty Git repository in C:/Users/student/git_basic/.git/
```

1. `(master)`
2. `.git` 폴더 생성

#### (중요) git 프로젝트를 종료

- `.git` 폴더 삭제

#### 2. `git status`

git의 현재 상태를 출력

- 최초상태

```
On branch master: master라는 브랜치에 있음
```

```
No commits yet: 아직 commit이 없음
```

```
nothing to commit (create/copy files and use "git add" to track)  
: commit할 게 없음
```

- 파일/폴더 추가시 (`touch a.txt`)

...

Untracked files: (추적되지 않은 파일)

(use "git add <file>..." to include in what will be committed)

a.txt (빨강)

nothing added to commit but untracked files present (use "git add" to track)

: commit하기 위해 추가된 파일이 없음, 그러나 추적되지 않은 파일은 있음

- add 이후(`git add a.txt`)

...

Changes to be committed:

commit할 변화들

(use "git rm --cached <file>..." to unstage)

new file: a.txt (초록)

- commit 이후(`git commit -m "first commit"`)

On branch master

nothing to commit, working tree clean

### 3. `git add` [파일명]

commit하기 위한 stage에 파일 추가

### 4. `git commit -m "커밋메시지"`

스냅샷 & 버전 생성

### commit의 구성 요소

- committer(author): commit을 찍은 사람
- commit datetime(date): commit을 찍은 시간
- commit message: commit에 대한 내용 (필수)

### 5. `git config`

- `git config --global user.email "이메일"`: 사용자 이메일 설정
- `git config --global user.name "이름"`: 사용자 이름 설정
- `git config --global user.email`: 사용자 이메일 출력
- `git config --global user.name`: 사용자 이름 출력

## 6. `git log`

commit 목록(log) 출력

- `git log --oneline`: 한줄 출력