

GIT 실습 1 (GIT 시작하기)

유튜브로 배우는 코딩 Git 강좌 - 김영욱

누구나 쉽게 이해할 수 있는 Git 입문

1. Git 최초 설정

Git 전역으로 사용자 이름과 이메일 주소를 설정

- Git 로컬 설정, GitHub 계정과는 별개
- 협업시 구분하는 정보값
- 프로젝트 마다 다르게 설정 가능

터미널 프로그램 (Git Bash)에서 아래 명령어 실행

```
git config --global user.name "(본인 이름)"
```

```
git config --global user.email "(본인 이메일)"
```

아래의 명령어들로 확인

```
git config --global user.name 본인이름
git config --global user.email 본인이메일
```

기본 브랜치명 변경

- 요즘 master / slave를 사용하지 않으며 GitHub도 main이 기본

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

Git 2.28.0부터 main으로 디폴트 적용이 되므로
Ubuntu 20.04에서는 Git을 업데이트 해야 된다.

```
sudo add-apt-repository ppa:git-core/ppa -y  
sudo apt-get update  
sudo apt-get install git -y
```

에러 발생시

Could not get lock /var/lib/dpkg/lock-frontent

<https://jenakim47.tistory.com/65>

설정 확인

```
git config --global --list
```

2. 프로젝트 생성 & Git 관리 시작

PC의 "C:\src"에 적당한 이름의 폴더를 생성하고

파일탐색기 cmd → code . → **VS Code**로 열람

이후 강의에서도 계속 사용

해당 폴더에서(VS Code 터미널 기본) 아래 명령어 입력

```
git init
```

폴더에 숨김모드로 **.git** 폴더 생성 확인

- 이 폴더를 지우면 Git 관리내역이 삭제됩니다. (현 파일들은 유지)

아래의 파일들 생성

tigers.yaml

```
team: Tigers

manager: John

members:
- Linda
- William
- David
```

lions.yaml

```
team: Lions

manager: Mary

members:
- Thomas
- Karen
- Margaret
```

모든 작업(파일 생성, 수정)마다 파일을 꼭 저장

여기까진 파일에만 저장되고 git에는 저장된 상태가 아님

터미널에 아래 명령어 입력

```
git status
```

- 추적하지 않는(**untracked**) 파일: Git의 관리에 들어간 적 없는 파일(메세지 확인)

소스트리로 해보기

현존하는 저장소 추가

- 소스트리에 폴더를 드래그하거나, 로컬 저장소 추가 (Add)
- 탐색기에서 git폴더 삭제. 북마크만 제거!!

Git이 관리하는 저장소 새로 만들기

- .git 폴더 삭제 후 진행
- 소스트리에 폴더를 드래그하거나, 로컬 저장소 생성 (Create)
- 탐색기에서 git폴더 생성 확인(git init과 같음)

3. Git의 관리에서 특정 파일/폴더를 배제해야 할 경우

a. 포함할 필요가 없을 때

- 자동으로 생성 또는 다운로드 되는 파일들 (빌드 결과물, 라이브러리)
- 꼭 필요한 파일이 아니면 저장소의 용량만 차지함

b. 포함하지 말아야 할 때

- 보안상 민감한 정보를 담은 파일(로그인 정보 등)

.gitignore 파일을 사용해서 배제할 요소들을 지정할 수 있습니다.

.gitignore 사용해보기

폴더에 아래 파일 생성(저장)

secrets.yaml

```
id: admin  
pw: 1234abcd
```

아래 명령어로 상태 확인

```
git status
```

.gitignore 파일 생성(VSCoDe 파일 아이콘 확인)

```
secrets.yaml
```

다시 상태 확인

```
git status
```

.gitignore 내용 삭제 후 다시 확인

협업시 대부분 사용됨

.gitignore 형식

<https://git-scm.com/docs/gitignore> 참조

```
# 이렇게 #를 사용해서 주석
# VSCoDe에서 파일 아이콘 변경되는 것 확인

# 파일이름, 모든 file.c
file.c

# 최상위 폴더의 file.c
# 폴더가 여러개 일 경우 폴더 위치 설정(/folder/file.c)
/file.c

# 모든 .c 확장자 파일(*.yaml)
*.c

# .c 확장자지만 무시하지 않을 파일, 예외
!not_ignore_this.c

# logs란 이름의 파일 또는 폴더와 그 내용들
logs

# logs란 이름의 폴더와 그 내용들
logs/

# logs 폴더 바로 안의 debug.log와 .c 파일들
```

```
logs/debug.log
```

```
logs/*.c
```

```
# logs 폴더 바로 안, 또는 그 안의 다른 폴더(들) 안의 debug.log
```

```
# *가 폴더 1개
```

```
logs/**/*.log
```
