



Git convention

▼ 목차

[목차](#)

[커밋 머릿말 규칙](#)

[깃 유용한 명령어\(커밋 되돌리기, 커밋메세지 변경 등..\)](#)

[💡!! remote로 푸시 및 커밋하기 전 확인 사항!!💡](#)

[branch 전략](#)

[feature branch 작성 규칙](#)

[CommitMessage 규칙](#)

[pre-commit-msg 파일 적용방법](#)

[pre-commit-msg 파일](#)

[pre-commit msg 적용 시 자동으로 에픽이 앞에 붙음](#)

[커밋 머릿말 규칙](#)

[Merge Request 작성방법](#)

▼ 커밋 머릿말 규칙

Type	content
Feat	새로운 기능 추가
Fix	버그 수정
Docs	문서 수정
Style	코드 포매팅, 세미콜론 등의 스타일 수정(코드 자체 수정 X)
Refactor	프로덕션 코드 리팩토링
Test	테스트 코드, 테스트 코드 리팩토링
Chore	빌드 과정 또는 보조 기능(문서 생성 기능 등) 수정
Rename	파일 혹은 폴더명을 수정하거나 옮기는 작업만인 경우
Remove	파일을 삭제하는 작업만 수행한 경우
Comment	필요한 주석 추가 및 변경

▼ 깃 유용한 명령어(커밋 되돌리기, 커밋메세지 변경 등..)

git add 취소하기 (파일 상태를 Unstage로 변경하기)

```
git reset HEAD <파일명>

// 파일명 안붙이면 add한 파일 전체가 날아간다 주의!!!!!!!!!!
git reset HEAD
```

git commit 취소하기

```
// commit 취소하고 해당 파일을 staged 상태로 워킹 디렉터리에 보존
git reset --soft HEAD^
// commit 취소하고 해당 파일들을 unstaged 상태로 워킹 디렉터리에 보존
git reset HEAD^

// commit 취소하고 해당 파일들을 unstaged 상태로 워킹 디렉터리에서 삭제
git reset --hard HEAD^
```

commit message 변경하기

```
git commit --amend
```

버전 확인

```
git --version  
  
git --v
```

config

```
git config --global user.name 유저네임  
//유저네임 등록  
  
git config --global user.email 유저이메일  
//유저이메일 등록  
  
git config --global core.editor "code --wait"  
//vs code 를 기본 에디터로 설정하겠다  
  
git config --list  
//git의 global 옵션 조회  
  
git config --global -e  
//gitconfig파일이 열리며 설정한 유저네임과 이메일, 코어정보를 조회 가능하다  
//ctrl + c 써서 빠져나올 수 있음  
  
code 파일명  
//파일을 마우스로 클릭하지않고 명령어로 여는 방법  
//ctrl + shift + p 눌러서 shell Command 찾아서 선택한다
```

CLI

```
git init  
//해당 디렉토리 초기화  
  
git status  
//현재 git의 상태  
  
git add 파일명  
//파일을 스테이지에 올린다  
  
git commit -m "메세지 입력"  
//add 되서 스테이지에 올라간 파일을 커밋한다  
  
git restore --staged 파일명  
//스테이지에 보냈던 파일을 다시 워킹디렉토리로 옮기는 방법  
  
git log  
//해당 디렉토리의 로그를 볼 수 있음  
  
git commit --amend  
//git 커밋 메세지 잘못 입력했을때 이름 바꾸는 방법(사실은 커밋 다시 진행하는것임)  
  
git log --oneline  
//git 로그를 한줄로 제목만 예쁘게 보여줌 ㅎㅎ  
  
git checkout HTML5/main.html  
//디스크드로 인한 명령어로도 쓰인다~  
  
git config --global core.autocrlf true  
//git 들여쓰기 오류 해결 코드
```

💡 **!! remote로 푸시 및 커밋하기 전 확인 사항!!** 💡

1. .git/hook 폴더에 prepare-commit-msg 파일 넣었는지 확인
 - a. 모르면 클릭 >>
2. 올바른 feature 브랜치에서 작업중인지 확인
3. 기능단위로 커밋했는지 확인
4. 작업이 완료된 local의 feature 브랜치를 깃랩으로 push → 깃랩에서 MR 작성 후 합치기
 - a. conflict가 많은 경우 깃랩에서 자동으로 합쳐지지 않을 수 있다, 이 때는 로컬에서 브랜치 당겨와서 합치고 푸시하기



Merge 할 경우 반드시 Target Branch 확인

- develop branch 가 default branch여서 기본 Target Branch가 develop으로 돼있음
- target branch 확인해서 frontend 혹은 backend 브랜치에 머지할 것

ps. develop branch 에 push나 merge를 제한해놔서 실수로 target branch를 develop으로 하더라도 병합이 안되므로 혹시 실수를 한다면 수정해서 재병합!

branch 전략

- Default Branch : Develop
 - default branch 에 merge가 될 때까지 계속 pull request가 떴서 방지하기 위함
 - 개발 다 끝나고 Master로 변경 예정
- Protected Branches
 - Master
 - 개발이 다 완료되기 전까지 코드 올릴 일 없음
 - Develop
 - 매일/매주 frontend 작업물과 backend 작업물을 병합하는 브랜치
 - push 권한을 no one으로 설정해놔서 혹시 모를 병합 방지
 - Frontend, Backend
 - 오류가 나는 작업물은 올리면 안되는 장소

feature branch 작성 규칙

Frontend, Backend 브랜치에서 feature/[JIRA 에픽코드] 브랜치 파서 작업

- Frontend, Backend 브랜치엔 오류가 나는 작업물은 병합 금지
 - 오류 원인 제공자는 벌칙

feature branch 작성예시
 feature/[JIRA 에픽코드]
 feature/S09P12A710-83

CommitMessage 규칙

```
git commit -m "[머릿말]: [JIRA 에픽명]
```

- [작업내용 1]
- [작업내용 2]
- ...

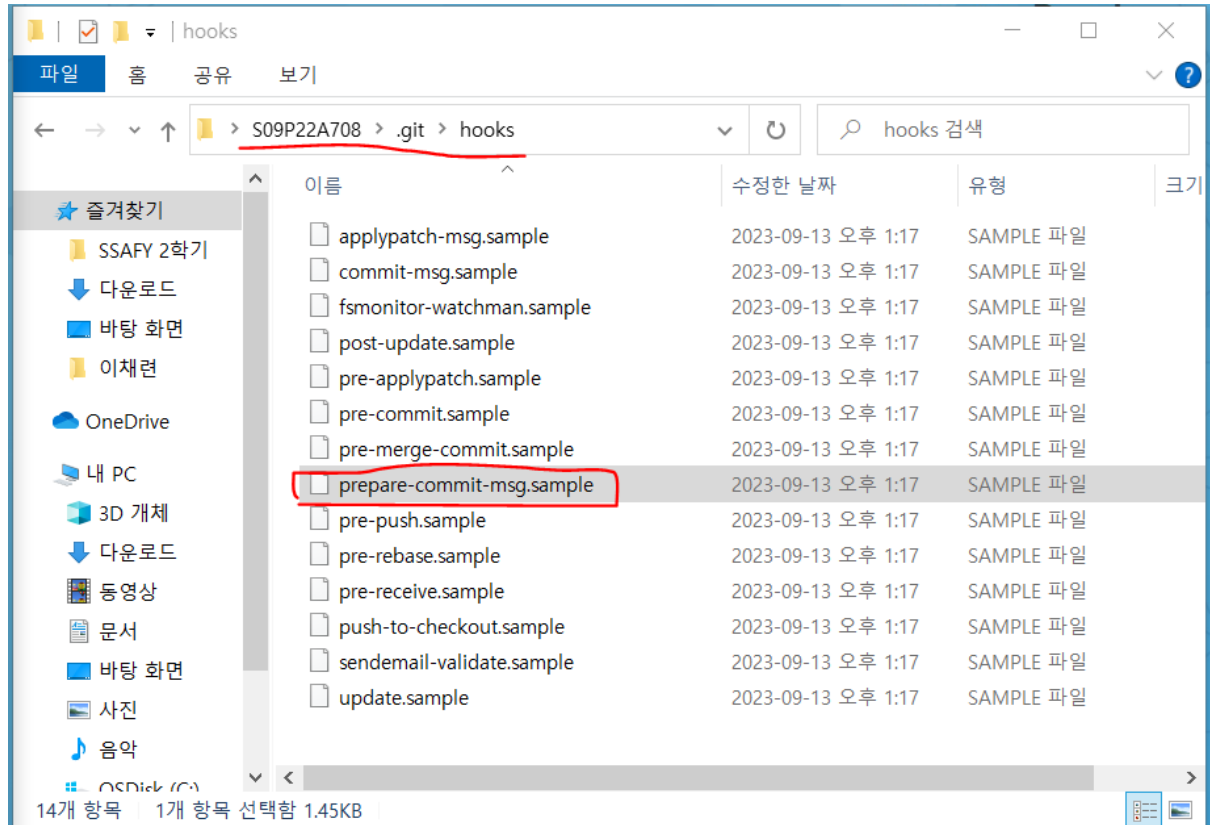
예시

```
$git commit m "Feat: 재무제표 컴포넌트 개발
- 재무제표 Page Component 제작
- 재무제표 Page 디자인 적용"
```

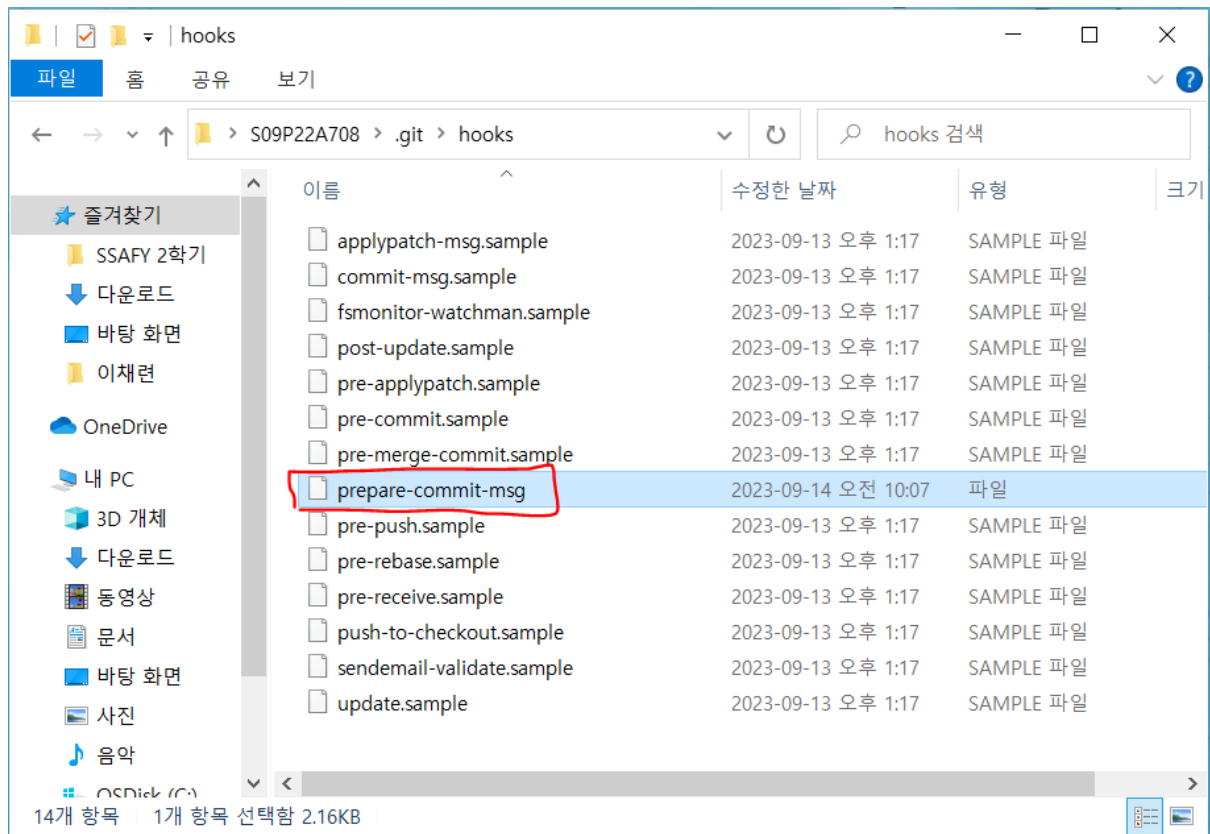
tip 커밋메시지 작성 시 수행하는 방법
첫 줄의 커밋메시지를 입력한 다음 따옴표를 닫지 않고 엔터를 치면됨다

pre-commit-msg 파일 적용방법

1. 프로젝트 상위 > .git > hooks 폴더에 prepare-commit-msg.sample 파일 제거



2. prepare-commit-msg 파일 추가



pre-commit-msg 파일

- 파일은 mm 단체메시지 방에 핀으로 고정해놨습니당

pre-commit msg 적용 시 자동으로 에픽이 앞에 붙음

```
git commit -m "Feat: [FE] 메인 페이지 제작
- 재무제표 Page Component 제작
- 재무제표 Page 디자인 적용"

!!!

S0922PA708-322 Feat: [FE] 메인 페이지 제작
- 재무제표 Page Component 제작
- 재무제표 Page 디자인 적용
```

커밋 머릿말 규칙

Type	content
Feat	새로운 기능 추가
Fix	버그 수정
Docs	문서 수정
Style	코드 포매팅, 세미콜론 등의 스타일 수정(코드 자체 수정 X)
Refactor	프로덕션 코드 리팩토링
Test	테스트 코드, 테스트 코드 리팩토링
Chore	빌드 과정 또는 보조 기능(문서 생성 기능 등) 수정
Rename	파일 혹은 폴더명을 수정하거나 옮기는 작업만인 경우
Remove	파일을 삭제하는 작업만 수행한 경우

Type	content
Comment	필요한 주석 추가 및 변경

Merge Request 작성방법

MR 제목

예시 : S09P12A710-572 Feat: 유저 설정 api 연동, 게시물 불러치러 구현완료

1 PR 타입(하나 이상의 PR 타입을 선택해주세요)

- ☐ 기능 추가
- ☐ 기능 수정(삭제)
- ☐ 코드 개선
- ☐ 기타

2 이슈 번호: ex) ex9P12A710-117(하위 이슈)

3 변경 사항: ex) 로그인 시, 구글 소셜 로그인 기능을 추가했습니다.

4 테스트 결과: ex) 베이스 브랜치에 포함되기 위한 코드는 모두 정상적으로 동작해야 합니다. 결과물에 대한 스크린샷, GIF, 혹은 라이브 데모가 가능하도록 샘플API를 첨부할

예시

S09P22A708-415 Feat: [FE] MainPage - 컴포넌트 제작

Edit

Code ▾

⋮

Merged 박재민 requested to merge feature/S09P22A708-415 into frontend just now

Overview 0

Commits -

Pipelines 0

Changes -

1 PR 타입(하나 이상의 PR 타입을 선택해주세요)

- ☐ 기능 추가
- ☐ 기능 수정(삭제)
- ☐ 코드 개선
- ☒ 기타

2 이슈 번호: S09P22A708-311

3 변경 사항: ex) 로그인 시, 구글 소셜 로그인 기능을 추가했습니다.

4 테스트 결과: ex) 베이스 브랜치에 포함되기 위한 코드는 모두 정상적으로 동작해야 합니다. 결과물에 대한 스크린샷, GIF, 혹은 라이브 데모가 가능하도록 샘플API를 첨부할 수도 있습니다.