Git 프로젝트의 세 가지 단계

- Git 디렉터리 : Git이 프로젝트의 메타데이터와 객체 데이터베이스를 저장하는 곳을 말한다.

- 워킹 디렉토리 : 프로젝트의 특정 버전을 Checkout한 것이다.

- Staging Area : Git 디렉터리에 있다. 단순한 파일이고 곧 커밋할 파일에 대한 정보를 저장한다.

Git으로 하는 일은 기본적으로 아래와 같다.

1. 워킹 디렉터리에서 파일을 수정한다.

2. Staging Area에 파일을 Stage해서 커밋할 스냅샷을 만든다.

3. Staging Area에 있는 파일들을 커밋해서 Git 디렉터리에 영구적인 스냅샷으로 저장한다.

( Git status로 확인 )

Unmodified(잡히지 않는다.)

Tracted File Modified(빨간색)

Working Directory Staged(초록색)

(Tracted File은 이미 스냅샷에 포함돼 있던 파일이다.)

Untracted File(= before add) (빨간색)

git diff

- 단순히 파일이 변경됐다는 사실이 아니라 어떤 내용이 변경됐는 지 살펴보려면 git status명령이 아니라 git diff 명령을 사용해야 한다.

- 이 명령은 워킹 디렉터리에 있는 것과 Staging Area에 있는 것을 비교한다.

- Unstaged 상태인 것들만 보여준다. 수정한 파일을 모두 Staging Area에 넣었다면 git diff 명령은 아무것도 출력하지 않는다.

파일 삭제하기

1. Staging Area에서만 제거하고 워킹 디렉터리에 있는 파일은 지우지 않고 남겨두려면

git rm –cached <file>을 사용하자.

2. git rm 명령은 워킹 디렉토리에 있는 파일도 삭제한다.

git rm –f <file>

3. 워킹 디렉토리에 있는 파일을 삭제하면 deleted 메시지가 뜬다.(status)

이 때 git rm <file> 명령을 실행하면 Staged 상태가 된다.(commit에 등록을 위해)

파일 상태를 Unstage로 변경하기

git reset HEAD <file> 특정파일 1개 Unstage

**git reset** : Clears your index, leaves your working directory untouched. (simply unstaging everything) (stage 비우기)

**git reset --hard**: Clears your index, reverts all changes in your working directory to the last commit. (the same thing as svn revert would do)

Index = 바로 다음에 커밋할 것들이다. = Stagin Area

HEAD = 현재 브랜치를 가리키는 포인터 + 그 브랜치에 담긴 커밋 중 가장 마지막 커밋

Modified 파일 되돌리기

$Git checkout -- <filename>

-> 원래 파일로 덮어썼기 때문에 수정한 내용은 전부 사라진다.

-> 변경한 내용을 쉽게 버릴 수는 없고 하지만 당장은 되돌려야만 하는 상황이라면 stash와 branch를 사용하자.

리모트 저장소

히스토리 단장하기

이 모든 것은 다른 사람들과 코드를 공유하기 전에 해야 한다.