GIT - 분산 버전 관리 시스템

*Mac에는 기본 내장되어 있는 시스템

GitHub - git을 저장하는 클라우드 저장소 (가장 많이 쓰임)

Git 명령어

1.git status - 현재 git의 상태를 보여줌.

2.git log - 현재 저장된 형상들의 로그기록을 보여줌

3.git reset - 과거로 돌아가는 방법

3-1.과감한 방법 - reset 과거로 돌아감과 동시에 그 이후의 형상들은 지워짐

git reset 돌아갈 시점 —hard

3-2.신중한 방법 - revert (취소한 시점이 새로운 캡슐로 생성되어짐)

git revert 취소할 시점

*시점은 git log의 일련번호 앞 6자리를 의미

4.Branch

4-1.새 Branch 생성하기 (자신만의 작업을 수행해보고 싶을 때)

git branch '생성할 branch 이름'

*각각의 branch는 바로 이전의 branch 파일을 모두 끌고옴.

4-2.다른 branch로 이동하기

git checkout '이동할 branch 이름'

git checkout -d '이동할 branch 이름' 생성 후 바로 이동이 가능

git checkout -b '생성할 로컬 branch명' origin/master '원격 branch명'

4-3.branch 병합하기 - 다른 branch의 작업이 괜찮을 경우 master와 결합하기

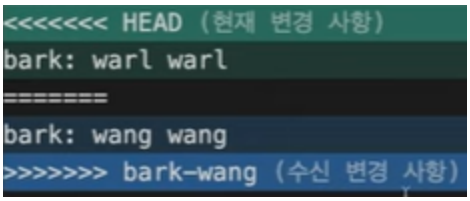
git merge 'master에 병합할 branch 이름'

4-4.branch 삭제 - branch를 다 사용해서 더 이상 필요하지 않은 경우

git branch -d '삭제할 branch 이름'

※각기 다른 branch에서 같은 파일을 수정한 후 병합할 경우(충돌 발생)

아래 사진처럼 충돌이 발생하여 병합이 이루어지지 않는다.



이러한 경우에는 둘 중 한 쪽의 수정 내용을 선택해서 git add -a, git commit까지만 실행해준다.

5. git rebase 'branch 이름' - 다른 branch에서 작업한 내용 가져오기

각 branch의 작업내용을 모두 병렬적으로 정렬하는 방법 (merge와는 다른 방식)

6.git remote - 원격 저장소 확인

7.gitignore - 하나의 임시 저장소로 여기에 올리고 싶지 않은 파일명을 적어두면 업로드하지 않음.

생성법 : touch .gitignore (.log ⇒ 확장자가 log인 경우는 모두 ignore 처리)

8.git fetch - 원격(remote) 저장소의 내용을 로컬 저장소에 불러오는 것(업데이트)

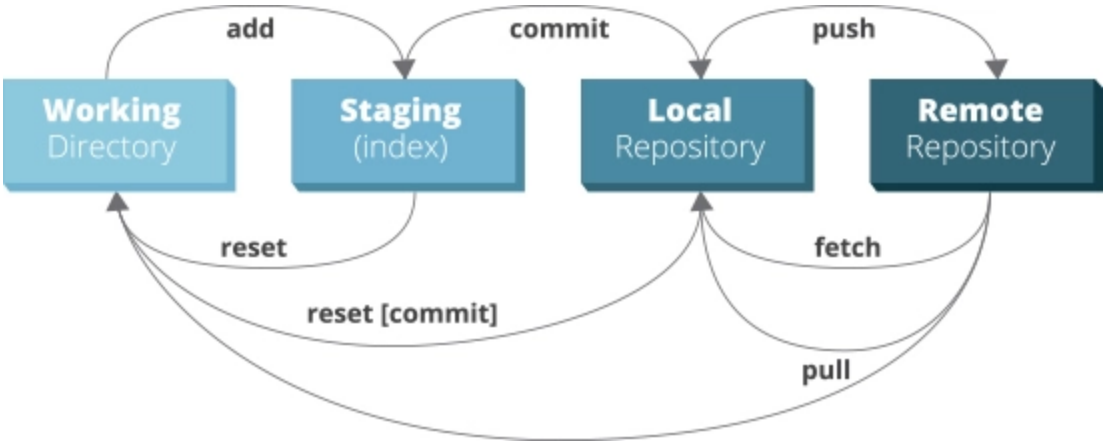
*readme - MarkDown의 역할을 하는 파일 (확장자-.md)

9.git pull - 원격 저장소의 내용을 로컬 저장소에 병합하는 것

*pull과 fetch 차이 - 병합의 유무 차이

*fetch의 경우는 각 저장소의 차이를 확인하고 직접 병합해야 함

※Git to Remote Repository 명령어 요약 시각화※



10.Git Clone - 다른 환경에서 같은 원격 저장소 환경을 사용할 때 사용

git clone '복제할 github URL'