깃의 3대목적

1. 버전관리
2. 백업
3. 협업
4. 버전관리

리포짓 만들기

* 리포짓으로 만들 디렉토리가서
* git init -> 데스크탑에 리포짓 만듬

버전 생성

* Working Tree: 버전이 되기전 파일
  + [untracked]
* Staging Area : 버전으로 만들려는 파일들 add 명령어를 통해 올림
  + git add -> [tracked]
* Repository : .git 파일 또는 버전이 저장되는곳 , commit 을 통해서 만들어짐
* 순서: 파일 수정 -> add 파일: staging에 올리기 -> commit : 버전 생성

버전간의 차이점 비교

* git diff : 현재 워킹트리와 마지막 버전의 차이점을 보여줌
* git reset –hard : 마지막 버전으로 돌아감

checkout을 통한 시간여행

* git checkout [log hash] 로 특정 버전으로 넘어갈수 있음
  + 지우면서 돌아가는 것이 아님 HEAD만 옮기는 것
  + 가장 최신으로 돌아가고 싶으면 git checkout master 명령어 입력

버전삭제 – git reset

* git reset –hard [log hash] : 해당 해쉬 버전으로 reset하겠다.
  + 그 버전의 앞에 있던 기록들 삭제

버전 되돌리기 – git revert

* git revert [log hash] : 자동으로 log hash 는 revert 됬다는 커밋을 추가해주고 해당 log hash의 수정값이 사라짐
  + a -> b -> c(최신) git revert c;  
    a -> b -> c -> c revert ~~ 가 되는거임
  + a로 돌아가고 싶다면 역순으로 c b 에 대해서 revert 해준다.
  + revert b을 하면 b에서 수정된것만 삭제를 하기 때문에 중간 log를 할려면 충돌이 발생할 확률이 높아서 사용되지 않는다.