영어단어

Rudimentary – 초보적인

**revert – 돌아가다**

meticulously – 꼼꼼하게

cohesive – 결합력 있는

precludes – 배제하다

disparity – 상이

inherent – 고유의

Repository는 모든 버전 제어 파일 및 기록된 변경 사항을 담고 있음

Commit은 편집 내용과 변경 사항을 저장. 파일의 스냅샷과 같음

Push는 변경사항을 repository에 update하는 것

Pull은 repository의 local version을 current version으로 update한다.

즉, 공유 repository는 여러 사람 혹은 개인이 여러 컴퓨터로 변경한 사항들을 update하고 Pull은 위에서 업데이트 한 내용을 내 local version에 적용하는 것

Staging은 파일 변경에 대한 commit을 분리해서 실행할 수 있게 도와줌

(그래서 한 번에 3개 커밋을 한다던지 하는 일은 없음)

R스튜디오와 깃헙저장소 연결하기

R스튜디오 프로젝트 생성 -> 글로벌 옵션 -> GIT -> SSH RSA 키 생성 복붙 -> 깃허브 프로필에서 SSH키 등록 (연결 완료) -> R에서 테스트 파일 생성 -> 오른쪽 상단 환경 탭에서 해당 파일 STAGED에 올리고 COMMIT -> PUSH -> 깃헙 저장소에 TESTING.R 파일 확인

처음부터 R 프로젝트를 버전컨트롤로 만들수 있고, 일반 프로젝트로 만들어도 깃과 연결할 수 있음 깃 배쉬에서 해당 프로젝트 디렉토리로 이동 후 stage -> 커밋 -> 푸쉬하고 깃허브에서 동일명의 레포지터리 만들고 ㅇremote 연결 -> push하면 연결됨