**2주차 oss**

**1. 개요**

- 프로그램의 코드, 소스가 공개된 소프트웨어 (예 : 텐서플로우, 파이토치)

- 특징 : 소스코드 공개 , 자유롭게 수정, 제작, 배포 , 저작권자의 권익 보호, 공동연구

- 미래의 공동연구를 보장하기 위해 소스코드가 공유되어야 하는 사상을 가진 소프트웨어

- 공개 소프트웨어 : 무료(예 안드로이드) / 비공개 소프트웨어 : 유료(예 윈도우)

**2. 버전관리 개요**

- 파일의 추가 및 수정 이력(추적) 관리 / 파일 집합에 대한 변경사항을 추적, 관리

- 누가, 언제, 어디를, 어떻게, 어느 파일을, 저장소가 여러 개인 겨우

- 기능 : 저장, 백업, 추적관리(충돌, 삭제, 추가 일시 확인)

- 명령어 인터페이스 방식(CLI) : 텍스트 입력, 결과 (예 git Bash)



- main : 현재 작접 저장소 브랜치 이름

- 그래픽 사용자 인터페이스(GUI) : 그래픽 대화화면에서 마우스와 텍스트 입력방식으로 명령-> 결과 인터페이스 (예: 소스트리)

- 커밋

- 저장소의 현 상태를 저장하는 행위 / 파일 집합의 변경내용을 깃저장소에 저장

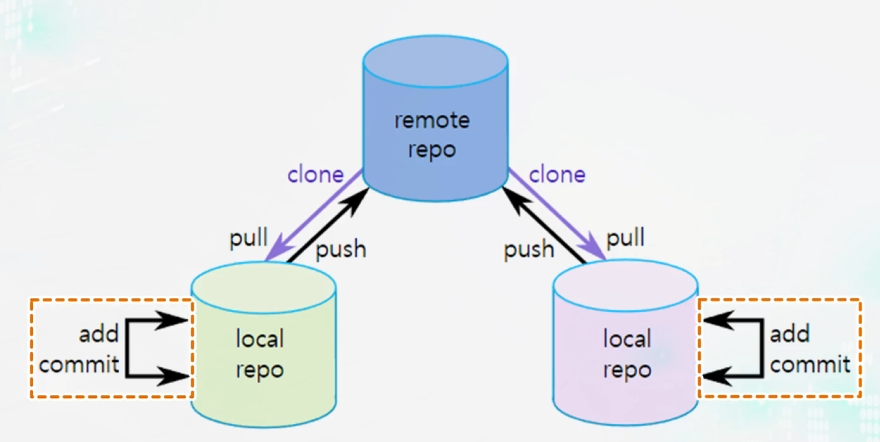
- 저장소(Git repository) : 연속된 커밋 관리 / 가장 최근 커밋 : HEAD

- 원격 저장소 : 전용서버 관리, 공유 / 지역 저장소 : 개인 전용 저장소

- 저장소 명령 : Clone(원격 저장소를 지역장소에 복사) / Pull(지역 저장소로 내리기)

Push(원격 저장소로 올리기)

add,commit(파일을 저장소에 저장, 순서대로, 커밋요약할때 커밋메시지저장(요약))



- 종류 : CVS, SVN, GIT, Mercurial, Bararr

3. 깃과 깃허브개요

- 깃

- 리누스 토발즈가 개발 / 주니오 하마노가 소프트웨어의 유지보수

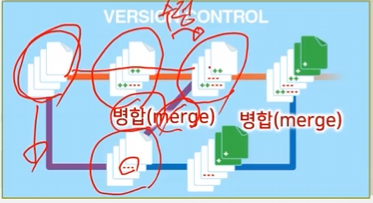
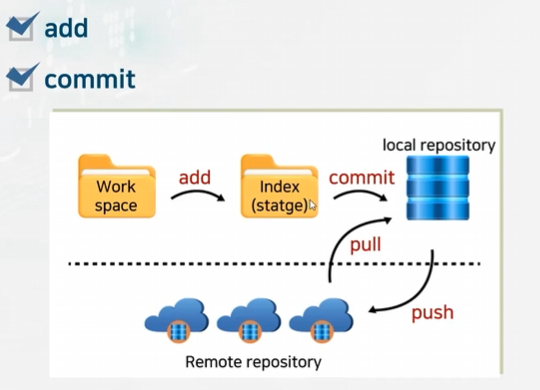
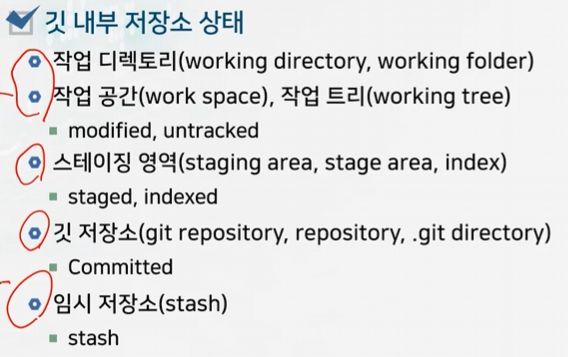
- 컴퓨터 파일의 변경을 추적하는데 사용되는 버전 관리 시스템

- 장점 : 모든 개발자는 지역시스템에 코드의 전체 사본을 소유

- 특징 : 여러 개발자 작업 / 코드 변경사항을 추적 / 코드관리에 분산 버전 제어 도구가사

용/ 여러 개의 평형 분기를 통해 비선형개발을 지원(branch)

- 소프트웨어 : git Bash(CLI) : 명령형 인터페이스 / Git Gui



- 브랜치 : 분기

- 분기를 이용한 프로젝트의 비선형 개발 작업흐름

- 새로운 수정을 할 수 있는 또 다른 버전의 작업흐름 / 파일 집합을 복사해 독립적으로 다시 개발

- 깃허브(RAILS 사용)

- 버전 관리를 위한 서비 저장소, 프로젝트 개발을 위한 협업 관리 서비스 / 웹 저장소

- 전문가 사이 소통, 협업, 통합, 자동화 지원/소스 공유, 협업 소프트웨어 빌드 플랫폼(sns)

- 분산 환경 소스코드 버전 관리 도구인 GIT의 글로벌 웹 호스팅 서비스

- 오픈소스 / 무료(공개 저장소만) 유료(비공개 저장소) / 중요 개발운영(DevOps) 기능

- 기능 : 이슈, 레이블, 마일스톤, 프로젝트, 끌어오기요청(PR), 분기, 커밋