

19~21

오픈소스 소프트웨어의 의미

소프트웨어의 소스코드를 자유롭게 읽고, 수정 및 재배포가 가능

오픈소스 소프트웨어 개발을 지원하는 서버인 깃허브에서의 협업 방식

github, gitlab, bitbucket

대표적인 오픈소스 소프트웨어

파이썬, 사이킷런, 파이토치, 텐서플로, MySql, MongoDB 등

오픈소스 소프트웨어의 저작권 이해

- 의무 강도
GPL - LGPL - Apache - BSD - MIT 순으로 의무 강도가 낮아짐]
- GPL : 2차 저작물 소스코드 공개 의무 있음
- LGPL : 특정 조건에서 2차 저작물 소스코드 공개 의무 없음
- Apache, BSD, MIT : 2차 저작물 소스코드 공개 의무 없음
-

리처드 스톤먼

LAMP : 대부분의 웹을 지원하는 서비스 스택 모델

- Linux : 오픈소스 운영 체제(OS)이자 세계 최대 규모의 오픈소스 프로젝트
- Apache : 초기 웹에서 핵심 역할을 한 오픈소스 크로스 플랫폼 웹 서버
- MySQL : 대부분의 데이터베이스 기반 웹 애플리케이션에서 사용하는 오픈소스 관계형 데이터베이스 관리 시스템
- PHP : 소프트웨어 개발에 사용되는 범용 스크립팅 언어 (가끔 Perl 또는 Python 언어로 대체됨)

stash : 작업 디렉토리와 스테이징 영역을 숨김(stash)에 저장하고 작업 폴더를 정리

- **\$ git stash**
- **\$ git stash -m '메시지'**
- **\$ git stash save**
- **\$ git stash save '메시지'**

옵션

- **- -keep -index, -k** : 스테이징 영역은 제외하고 작업 폴더만 저장
- **- -include -untracked, -u** : untracked 파일도 포함해 저장

최근 또는 지정된 임시저장소 내용을 가져와 반영하고 **삭제**

- **\$ git stash pop**
- **\$ git stash pop stash@{n}**

최근 또는 지정된 임시저장소 내용을 가져와 반영, **작업 디렉토리만 반영, stash 목록은 그대로**

- **\$ git stash apply** : 기본으로 작업 디렉토리 내용만 다시 복사해 활용
- **\$ git stash apply stash@{n}**

최근 또는 지정된 임시저장소 내용을 가져와 반영, **작업 디렉토리와 스테이징 영역도 반영, stash 목록은 그대로**

- **\$ git stash apply - -index**
- **\$ git stash apply - -index stash@{n}**

임시저장 목록 보기

\$ git stash list

stash@{0} (가장 최신 것)

stash@{1}

stash@{2}

...

커밋 자료와 최신 stash 항목 간의 차이로 표시

- \$ git stash show
- \$ git stash show -p : 파일 내용의 차이까지 보이기

커밋 자료와 해당 stash 항목 간의 차이로 표시

- \$ git stash show stash@{n}
- \$ git stash show stash@{n} -p

최근 임시저장 내용을 삭제

\$ git stash drop

지정된 임시저장 내용을 삭제

\$ git stash drop stash@{n}

모든 stash 목록을 모두 제거

\$ git stash clear

stash

- 커밋할 필요 없이 파일의 변경 사항을 숨기거나 비밀리에 저장할 수 있는 도구
 - 작업 디렉토리와 스테이징 영역의 원래의 내용을 stash 스택에 저장
 - 작업 디렉토리와 스테이징 영역의 값을 깃 저장소의 마지막 커밋 내용으로 복사
- 따로 안전한 곳에 저장했다가 다시 가져올 수 있는 기능
 - 저장 내용 : 작업 디렉토리 내용과 스테이지 내용

LIFO(Last In - First Out)

- push : 자료 넣는것 (가장 예전 것이 밑에 쌓이게 됨) / pop : 자료 꺼내는것

untracked 파일 삭제

작업 디렉토리에 있는 파일을 말함. 리눅스 명령어 rm f

깃 명령어 :

- \$ git clean (바로 삭제되지 않음)
- \$ git clean -f : 무조건 삭제
- \$ git clean -i : 대화형 / 숫자나 첫 알파벳으로 선택