



Session 3

고급 Merge(Advanced Merge)

- merge 충돌을 해결하는 방법

- merge 취소

```
git merge --abort
```

- 공백 무시하기

```
git merge -Xignore-space-change {branch}
```

- 수동으로 merge하기

- 충돌 파일 checkout

- merge 로그

- combined diff 형식

- merge 되돌리기

- refs 수정

```
git reset --hard HEAD~
```

- 커밋 되돌리기 ; 모든 변경사항을 취소하는 새로운 커밋을 만드는 revert

```
git revert -m 1 HEAD
```

다른 방식의 merge

- out/their 선택하기
 - `git merge -Xours(Xtheirs)` 옵션 추가 → 충돌이 뜨지 않음
- 서브트리 merge ; 프로젝트 두개가 있을 때 한 프로젝트를 다른 프로젝트의 하위 디렉토리로 매핑하여 사용하는 것

Rerere

- 기록한 해결책 재사용하기
 - 충돌이 났을 때 각 코드 덩어리를 어떻게 해결했는지 기록해두었다가 나중에 같은 충돌이 나면 기록을 참고하여 자동으로 해결
 - 긴 호흡의 branch를 깔끔하게 merge 하고 싶을 때
 - 브랜치를 rebase 할 때

```
git config --global rerere.enabled true
```

```
git rerere statue #충돌 난 파일 확인  
git rerere diff #해결 중인 상태 확인
```

Submodules

- git 저장소 안에 다른 git 저장소를 디렉토리로 분리해 넣는 것
 - 다른 독립된 git 저장소를 clone 해서 내 git 저장소에 포함할 수 있으며 각 저장소의 커밋은 독립적으로 관리

```
git submodule add {추가할 저장소 URL}
```

- 서브모듈을 포함한 프로젝트 clone

```
git submodule init #서브모듈 정보를 기반으로 로컬 환경설정 파일 준비
git submodule update #서브모듈 프로젝트 checkout
git clone --recursive {URL} #서브모듈을 자동으로 초기화하고 업데이트
```

- 서브모듈을 포함한 프로젝트 작업

- 서브모듈 update

```
#서브모듈 디렉토리에서
git fetch
git merge origin/master

git diff --submodule #업데이트된 서브모듈과 서브모듈에 추가된 커밋

#또는
git submodule update --remote DbConnector
```

- 서브모듈 수정사항 공유하기

```
git push --recurse-submodules=check #서브모듈의 로컬 커밋이
git push --recurse-submodules=on-demand $git이 push를 대
```

- 서브모듈 merge

```
git rev-parse HEAD
git branch try-merge
git merge try-merge
```

- git submodule foreach → 모든 서브모듈에게 동일한 명령어 적용

Bundling

- git push 명령으로 올려보낼 모든 것을 하나의 바이너리 파일로 만들어 이메일로 보내거나 usb로 다른 사람에게 보내서 다른 저장소에 풀어서 사용

```
git bundle create repo.bundle HEAD master #bundle 생성
```

```
git clone repo.bundle repo #bundle 가져오기
```

Replace

- 어떤 개체를 읽을 때 항상 다른 개체로 보이게 하는 명령어

```
git replace {이전 commit} {new commit}
```

Git Attributes

- binary files
 - word 파일 diff 하기

```
*.docx diff=word. #"word" 필터 사용
```

- commit/checkout 필터

```
git config --global filter.indent.clean indent  
git config --global filter.indent.smudge cat
```