



Git 핵심노트

1. Git의 데이터 저장 방법

- (1) Git에서 데이터를 저장하기 위해 사용하는 네 가지 객체
- (2) 데이터의 저장
- (3) 커밋 히스토리

2. 저장소 구조

3. 브랜치

4. 변경사항 되돌리기

- (1) 리셋
- (2) 리버트
- (3) 리셋과 리버트 각각 언제 사용해야할까?

5. 머지

- (1) 머지의 두 가지 방식

6. 리베이스

- (1) 동작
- (2) 인터랙티브 리베이스
- (3) 주의사항

7. 브랜치 관리 전략

1. Git의 데이터 저장 방법

(1) Git에서 데이터를 저장하기 위해 사용하는 네 가지 객체

- 1) 블랍(BLOB): 파일/폴더시스템의 파일에 해당. 파일의 내용만 저장
- 2) 트리(Tree): 파일/폴더 시스템의 폴더에 해당. 블랍, 트리의 권한정보 및 이름을 저장
- 3) 커밋(Commit): 커밋 명령을 실행하면 생성되는 객체

4) 태그(Tag): 특정 커밋을 기억하기 위해 사용하는 객체

(2) 데이터의 저장

- 각 객체는 Key/Value로 구성돼 있음
- Key는 SHA-1 값 (객체의 내용으로 해시를 생성)
- Value는 객체의 내용
- BLOB ← Tree ← Commit ← Tag 형태로 서로 참조하고 있음

(3) 커밋 히스토리

- DAG(directed asyclic graph) 구조로 구성돼 있음.
- 전역적으로 단방향 그래프를 사용

2. 저장소 구조

- `.git` 파일이 Git 저장소
- 저장소 구조는 작업디렉토리 - 스테이징영역(index) - 객체데이터베이스로 구성
 - 작업디렉토리: 소스를 편집하는 공간
 - 스테이징영역(`.git/index`): 변경사항을 선택적으로 적용할 수 있게 해줌
 - 객체데이터 베이스(`.git/objects`): Git에서 데이터 저장을 위해 사용하는 4가지 객체를 저장

3. 브랜치

- 독립된 개발을 보장해 줌
- 커밋의 북마크 (움직이는 포인터, 태그: 고정된 포인터)
- 구현 : 커밋의 해시를 저장하는 41바이트의 파일로 구성
- 종류: 로컬 브랜치(develop), 원격 저장소 추적 브랜치(origin/develop), 원격 저장소 브랜치 (remote develop)

4. 변경사항 되돌리기

리셋과 리버트 두 가지 방법을 제공

(1) 리셋

- 선택한 커밋으로 head를 바꿔 변경사항을 되돌림
- 명령 수행시 커밋히스토리에서 사라지는 내용을 어떻게 처리할지 결정하는 옵션을 지정
 - soft: 사라지는 내용을 index에 둠
 - mixed(default): 사라지는 내용을 working directory에 둠
 - hard: 사라지는 내용을 어디에도 남기지 않고 완전히 제거

(2) 리버트

- 선택한 커밋과 반대되는 내용을 갖는 커밋을 추가해 변경사항을 되돌림
- 커밋히스토리 중간의 커밋을 되돌릴 수 있음

(3) 리셋과 리버트 각각 언제 사용해야할까?

- 제거하려는 변경사항이 원격저장소에 반영된 적이 있으면 리버트 사용(리셋 사용하면 안됨)
- 중간 커밋만 되돌리고 싶을 때 리버트 사용

5. 머지

(1) 머지의 두 가지 방식

- 1) Fast-Forward 머지: 브랜치의 포인터만 변경해 머지
- 2) Three Way 머지: 머지 커밋을 생성해 머지

6. 리베이스

커밋의 베이스를 변경해 커밋히스토리를 변경하는 명령

(1) 동작

리베이스를 수행하면 내부적으로 패치를 생성해 커밋을 재생성

(2) 인터랙티브 리베이스

- 커밋 재생성시 커밋의 메시지, 순서를 변경하거나 삭제, 병합, 쪼개기(스퀴시) 등이 가능함
- 스퀴시: 가장 많이 사용하는 방식

(3) 주의사항

일반적으로 원격저장소에 반영된 커밋히스토리는 리베이스하면 안됨

7. 브랜치 관리 전략

- 브랜치 관리 전략 종류: Gitflow, Github flow, Gitlab flow, One flow, Trunk based development 등
- Gitflow: master, develop, feature, release, hotfix 브랜치 사용



안녕하세요?

저는 깃미남(깃에 미친 남자) 라고 합니다.

충돌처리, 원격저장소, 협업하기 - 풀리퀘스트(Pull Request), 잃어버린 커밋 복원하기, Git 사용 팁 14가지 등 더 많은 Git 지식을 더 자세히 알고 싶으시면 연락주세요.

보다 자세한 내용은 [깃미남 사이트](#)를 참조하세요.