

2주차

- revision selection: 7.1
- interactive staging: 7.2
- stash: 7.3
 - push
 - apply & pop
 - drop
- bisect: 7.5
- rebase deep dive: 7.6
 - commit 합치기
 - commit 분리하기
 - `-onto`
- reset & checkout는 어떻게 작동할까?: 7.7

SHA-1 해시?

입력 데이터를 고정된 길이의 고유한 40자리(16진수) 문자열로 변환하는 해시 알고리즘. Git은 이 해시 알고리즘을 사용해 커밋의 내용(변경된 파일 내용, 커밋 메시지, 시간 등)을 바탕으로 고유한 커밋 ID를 만든다.

=각 커밋 고유하게 식별하는 긴 문자열(고유ID)

SHA-1 줄여 쓰기

해시가 저장소 내에서 유일하다면 앞부분 4글자만으로도 커밋을 나타낼 수 있다.

`$ git log --abbrev-commit` : 짧고 중복되지 않는 해시 값을 보여준다.

```
$ git log --abbrev-commit --pretty=oneline
ca82a6d changed the version number
085bb3b removed unnecessary test code
a11bef0 first commit
```

브랜치로 가리키기

어떤 커밋이 브랜치의 가장 최신 커밋이라면, 커밋을 SHA-1 값이 아닌 브랜치 이름으로 가리키기.

```
$ git show topic1
```

 → topic1 브랜치의 최근 커밋 보는 것.

RefLog로 가리키기

Reflog=브랜치, HEAD가 지난 몇 달 동안에 가리켰었던 커밋의 기록(로그)

```
git reflog
```

 : Reflog 볼 수 있음.

```
$ git reflog
734713b HEAD@{0}: commit: fixed refs handling, added gc auto, updated
d921970 HEAD@{1}: merge phedders/rdocs: Merge made by the 'recursive' strategy.
1c002dd HEAD@{2}: commit: added some blame and merge stuff
1c36188 HEAD@{3}: rebase -i (squash): updating HEAD
95df984 HEAD@{4}: commit: # This is a combination of two commits.
1c36188 HEAD@{5}: rebase -i (squash): updating HEAD
7e05da5 HEAD@{6}: rebase -i (pick): updating HEAD
```