

형상 관리를 위한 Git & Github

Chapter 2

Git 브랜치 전략

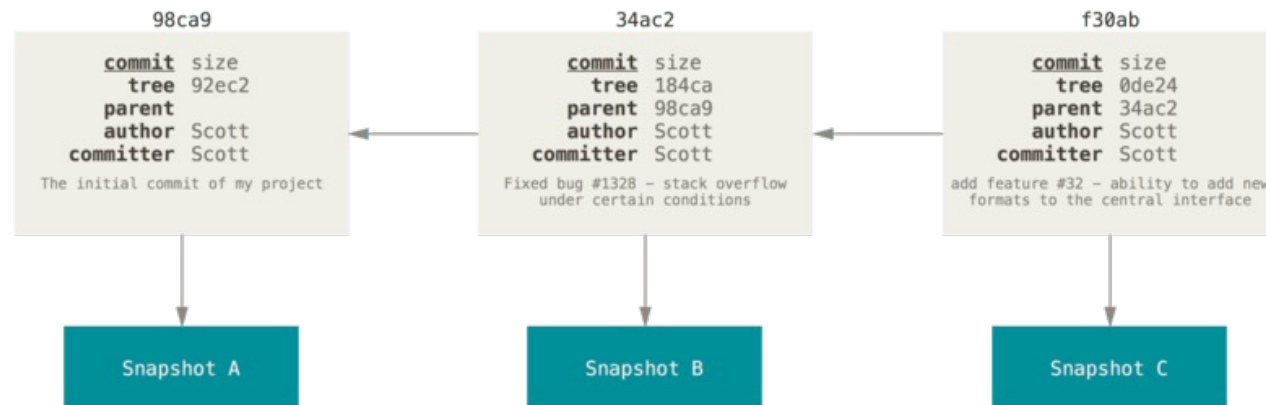
Chapter 2

Git 브랜치 전략

Git 브랜치 전략(1)

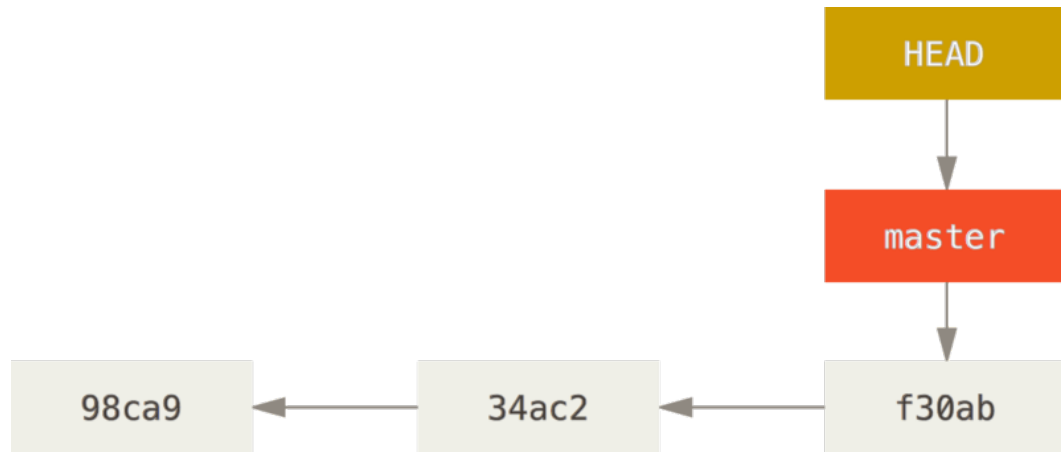
Git 브랜치

- Git은 변경 사항을 기록하는 것이 아니라 일련의 **스냅샷**을 기록
- Commit을 하면 데이터의 스냅샷에 대한 포인터 등을 포함한 **Commit Object** 생성
- 브랜치는 Commit 사이를 가볍게 이동하는 **포인터**



Git 브랜치

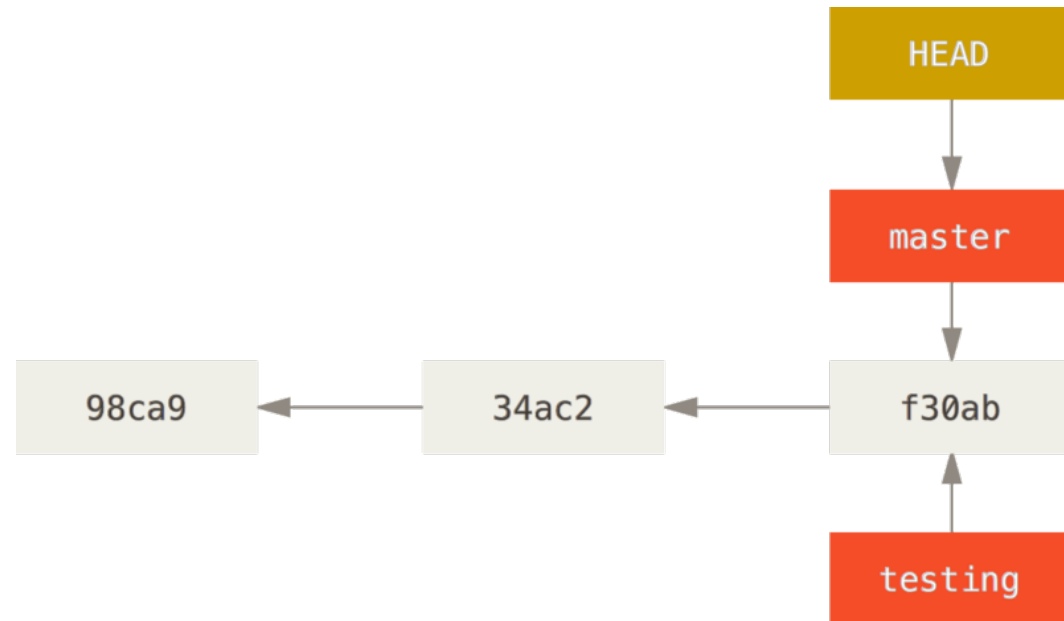
HEAD : 지금 작업하는 로컬 브랜치를 가리키는 포인터



Git 브랜치

0. 브랜치 생성하기

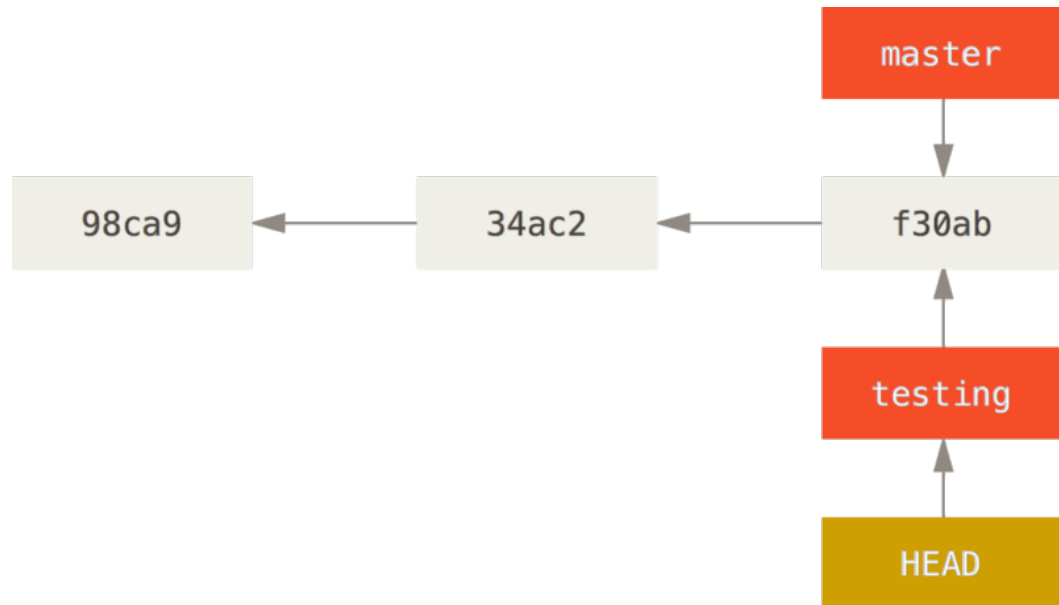
\$ git branch testing



Git 브랜치

1. 브랜치 이동하기

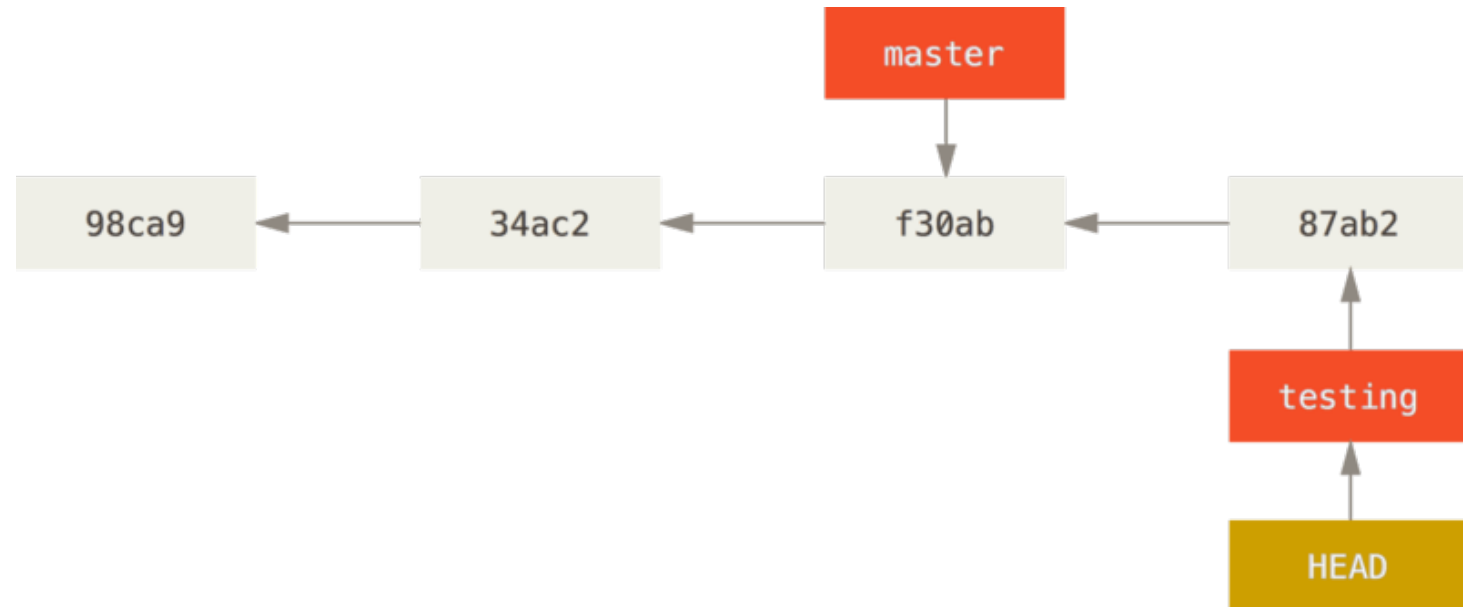
\$ git checkout testing



Git 브랜치

2. 이동한 브랜치에서 commit 생성

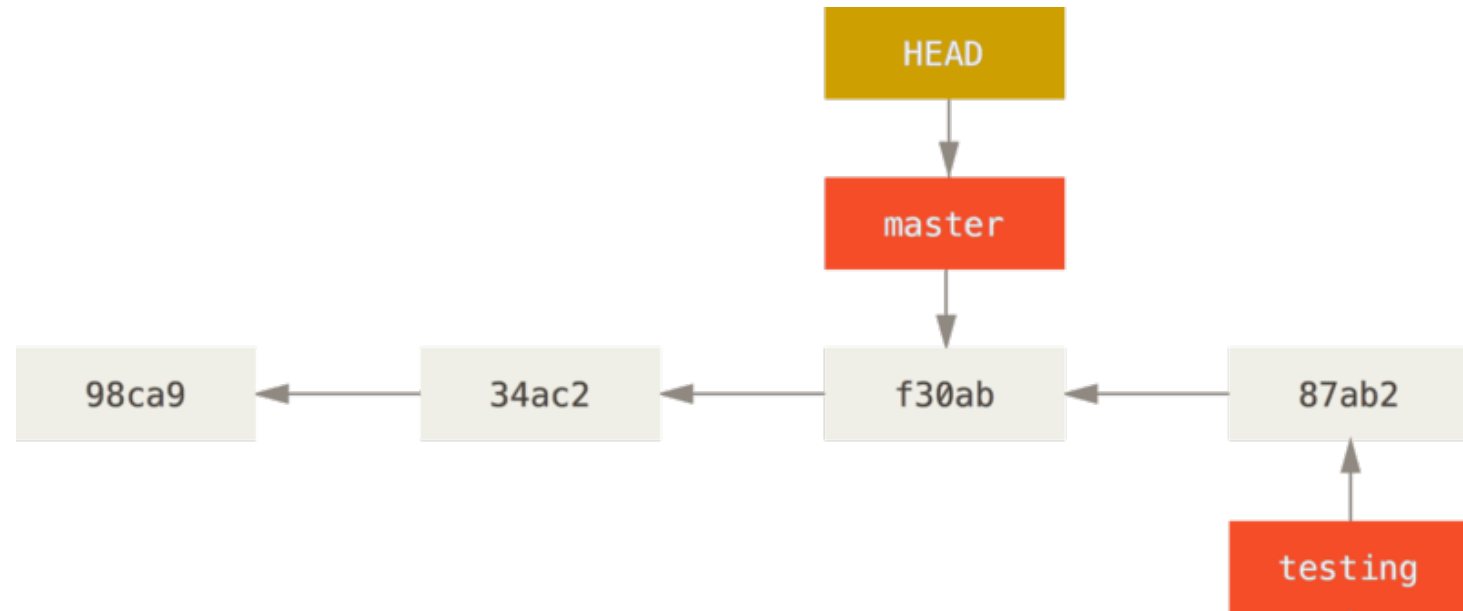
\$ git commit -a -m 'made a change in testing'



Git 브랜치

3. master 브랜치로 돌아가기

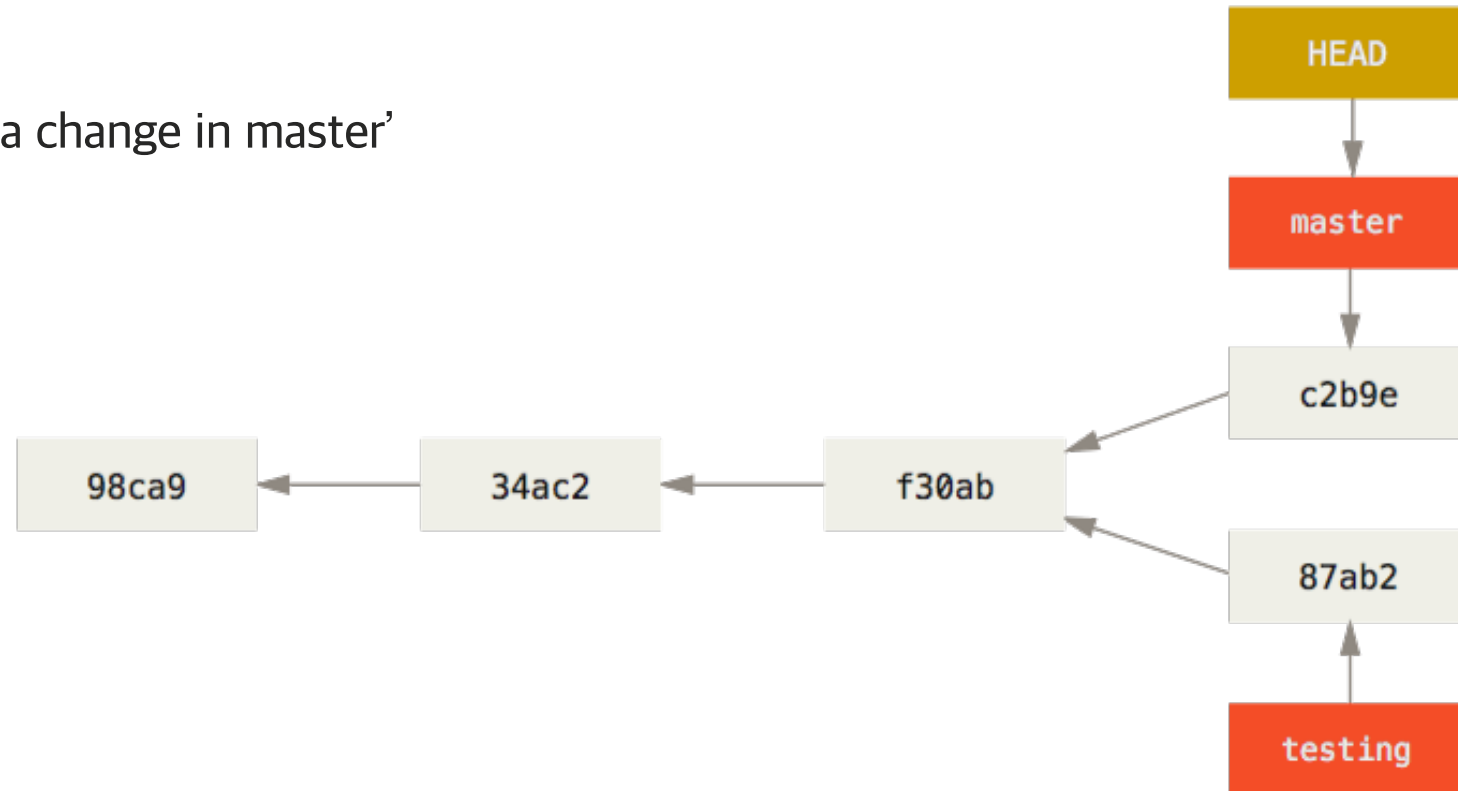
\$ git checkout master



Git 브랜치

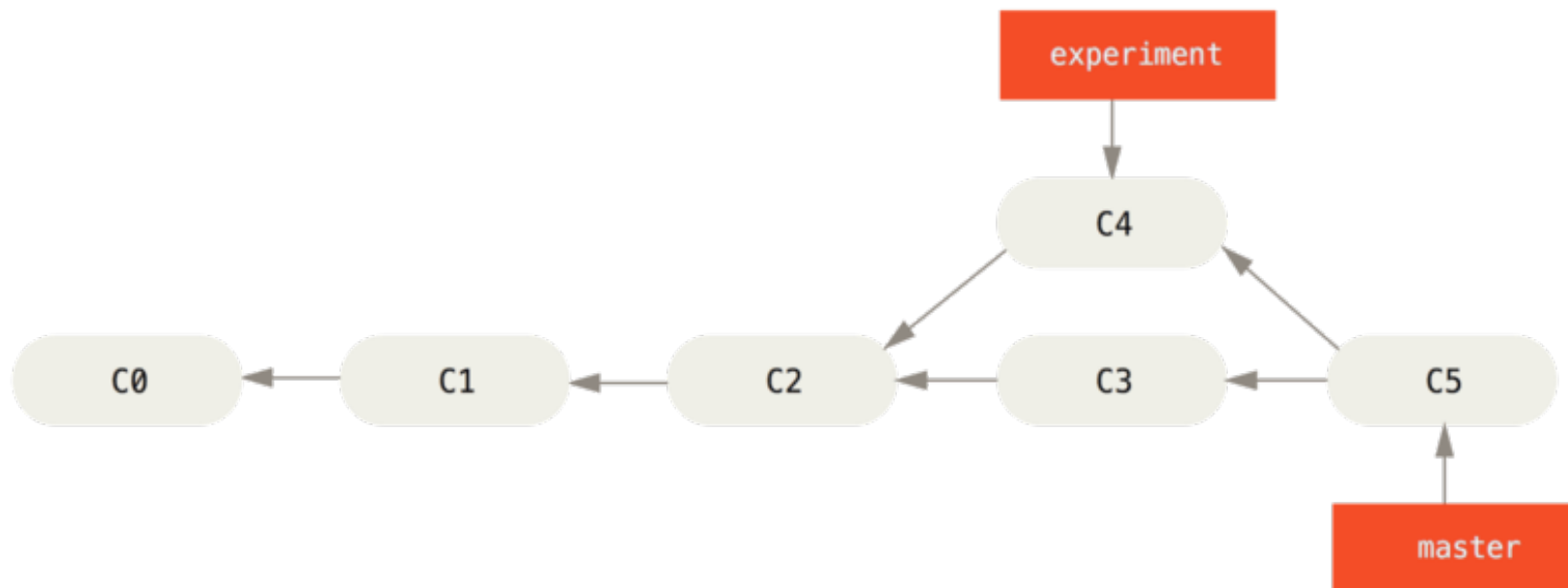
4. 브랜치 가르기

\$ git commit -a -m 'made a change in master'



Git 브랜치 합치기 (Merge)

기존 브랜치가 유지 됨



Git 브랜치 합치기 (Rebase)

원본 브랜치에서 각 commit에 대해 새로운 commit을 만들어 기록을 다시 작성

