

Open source Software

오픈소스 소프트웨어

22. 다양한 브랜치 병합



학습 개요

1. 브랜치 병합 merge
2. fast-forward, 3-way merge
3. squash merge



학습 목표

1. 브랜치의 병합(merge)을 이해하고 수행할 수 있다.
2. 빨리 감기(fast-forward) 병합을 수행할 수 있다.
3. 3-way 상태를 이해하고 3-way 병합을 수행할 수 있다.
4. 기본이 fast-forward 병합이 수행되는 일련 상태에서 3-way 병합을 수행할 수 있다.

LESSON 01

브랜치 병합 개요



1 브랜치 병합개요

🔧 브랜치 병합

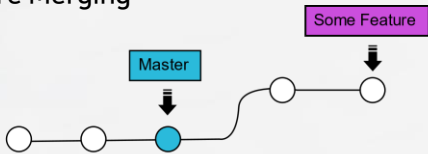
✅ 병합(merge)

- 두 개의 브랜치를 하나로 모으는 과정
 - fast-forward(빨리 감기) 병합
 - 3-awy 병합

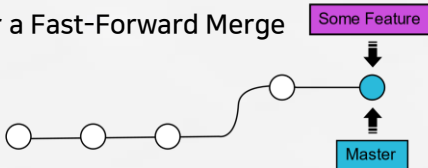
Fast-forward merge VS

3-way merge

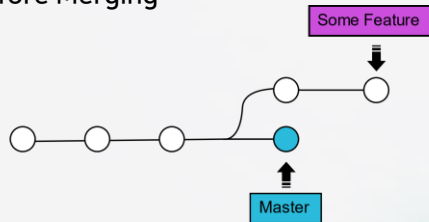
Before Merging



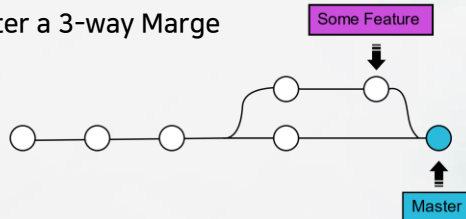
After a Fast-Forward Merge



Before Merging



After a 3-way Merge

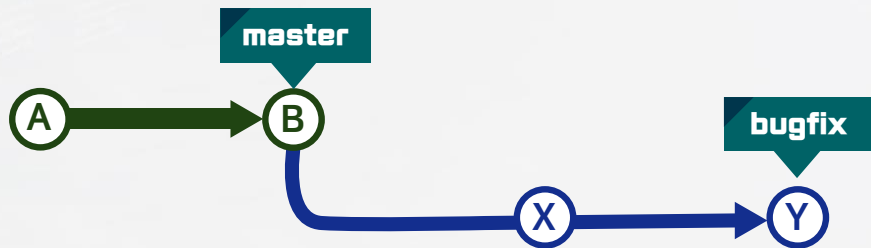


1 브랜치 병합개요

⚙️ 빨리 감기 fast-forward 병합의 이해

✓ fast-forward 병합 조건

- 현재 브랜치 master가 병합할 대상 커밋의 직접적인 뿌리(조상)가 되는 경우
 - 간단히 두 브랜치가 일렬 상태
 - bugfix 브랜치 이력이 master 브랜치 이력을 모두 포함하는 경우



✓ 브랜치 master에서 병합 명령

\$ git merge bugfix

1 브랜치 병합개요

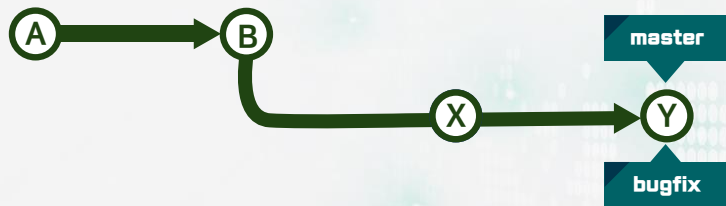
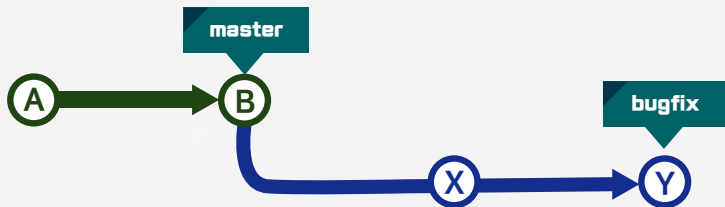
⚙️ 빨리 감기 fast-forward 병합 수행

✓ master 브랜치는 단순히 이동

- 이 때는 합칠 내용이 없음
 - 간단히 가리키는 지점이 대상 커밋이 되고 master가 bugfix로 이동
 - 작업공간과 스테이징 영역이 이동되는 Y 상태로 됨

✓ 기준 브랜치 master

- \$ git merge bugfix
 - 병합 전 **\$ git merge bugfix** ➡ 병합 후

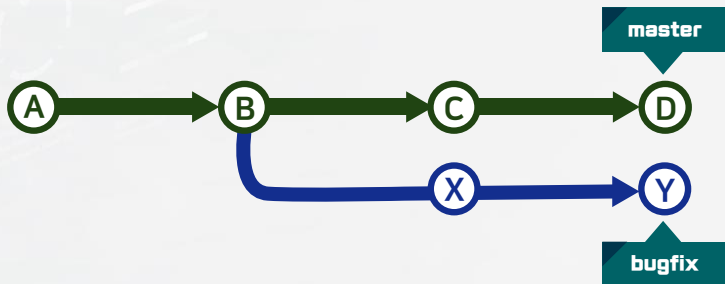


1 브랜치 병합개요

⚙️ 3-way 상태: 두 분기가 갈라진 상태

✓ 두 브랜치의 조상이 같은 경우

- master 브랜치 내의 변경 내용과 bugfix 브랜치 내의 변경 내용을 하나로 통합할 필요



✓ 3-way 병합

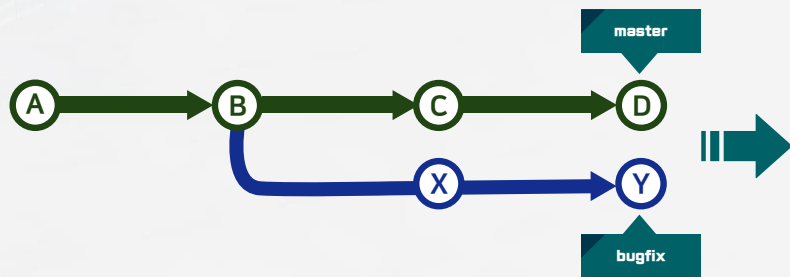
- 새로운 커밋을 사용하여 두 기록을 합침

1 브랜치 병합개요

3-way 병합 수행

✓ 새로운 커밋 E 생성

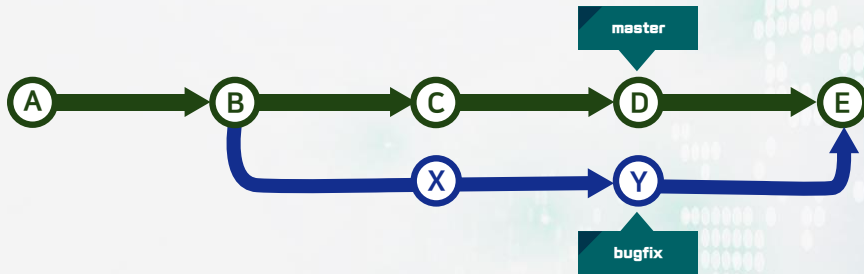
- 두 브랜치의 변경을 가져온 'merge commit(병합 커밋)' E를 생성
 - 병합 완료 후, 통합 브랜치인 'master' 브랜치로 통합된 이력이 생성
 - -m이 없으면 메시지 입력할 기본 편집기 실행
 - ➡ 'Merge branch bugfix'이 기본 메시지 내용



```
$ git merge bugfix
```

```
$ git merge bugfix -m 'merge msg'
```

-m이 없으면 메시지 입력할 기본
편집기 실행

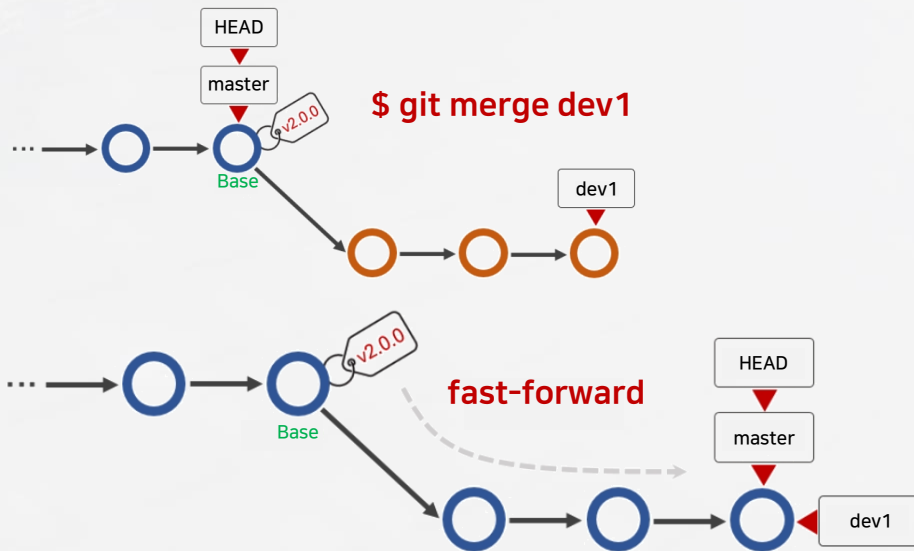


1 브랜치 병합개요

일렬 상태에서 기본이 fast forward 병합

병합할 브랜치의 조상이 기준 브랜치인 경우, 즉 일렬 상태에서

- 병합 기본은 fast forward 병합
- 마스터 브랜치에서 브랜치 dev1을 병합
 - 기본이 fast-forward 병합

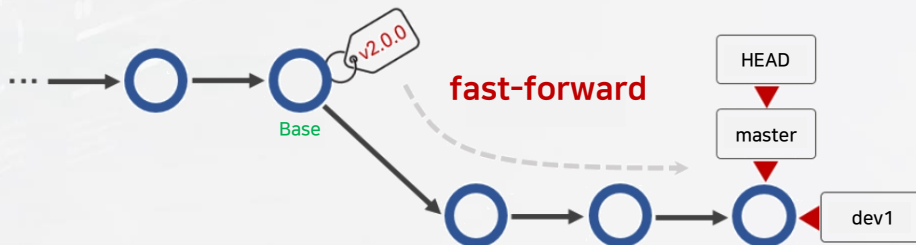


1 브랜치 병합개요

기본이 fast forward 상태에서 non fast forward 병합

✓ 마스터에서 dev1을 fast forward로 병합

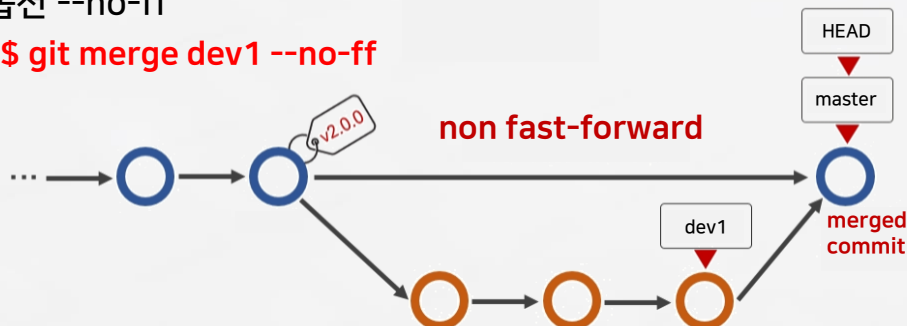
- \$ git merge dev1



✓ 마스터에서 dev1을 non fast forward로 병합(3-way 병합)

- 옵션 --no-ff

\$ git merge dev1 --no-ff



1 브랜치 병합개요

⚙️ 두 브랜치의 일렬 상태에서 2가지 병합

☑️ 일렬 상태



☑️ 두 가지 병합 방법

• 옵션 없는 경우

■ fast-forward merge

\$ git merge topic

• 옵션 --no-ff

■ 3-way merge

\$ git merge --no-ff topic



LESSON 02

병합의 다양한 옵션

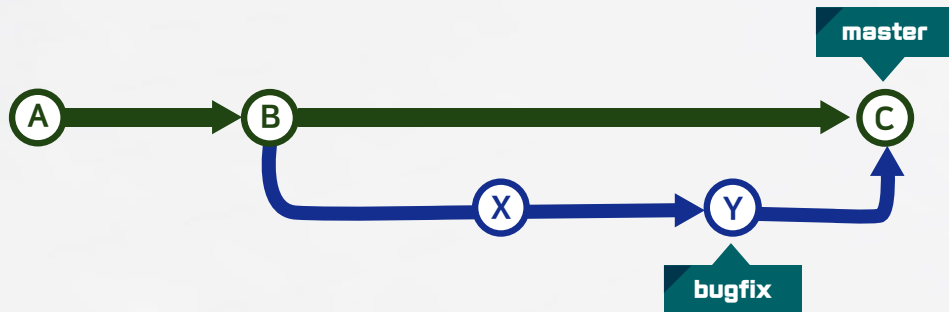


2 병합의 다양한 옵션

⚙️ 'non fast-forward 병합' 옵션을 지정

✓ non fast-forward 병합

- \$ git merge --no-ff {병합할 브랜치 명}
- 병합 실행 시에 'fast-forward 병합'이 가능한 경우라도 3-way 병합을 수행
 - 병합된 브랜치가 그대로 남기 때문에
 - ➡ 그 브랜치로 실행한 작업 확인 및 브랜치 관리 면에서 더 유용



2 병합의 다양한 옵션

병합의 다양한 옵션 종류

\$ git merge {병합할 브랜치 명}

- 보통의 병합, 융통성 있는 병합
- 현 브랜치와 병합할 브랜치가 일렬 상태
 - fast-forward 병합
 - ➡ 새로운 병합 커밋 없이 병합할 브랜치 커밋으로 이동
- 현 브랜치와 병합할 브랜치가 갈라져 있는 상태
 - 3-way 병합
 - ➡ 새로운 병합 커밋을 생성해 두 브랜치를 합침

\$ git merge --no-ff {병합할_브랜치_명}

- 무조건 3-way 병합되는 옵션

2 병합의 다양한 옵션

병합의 다양한 옵션 종류

✓ \$ git merge --ff-only {병합할_브랜치_명}

- 상태가 fast-Forward인 일렬 상태에서만 병합 진행

✓ \$ git merge --squash {병합할_브랜치_명}

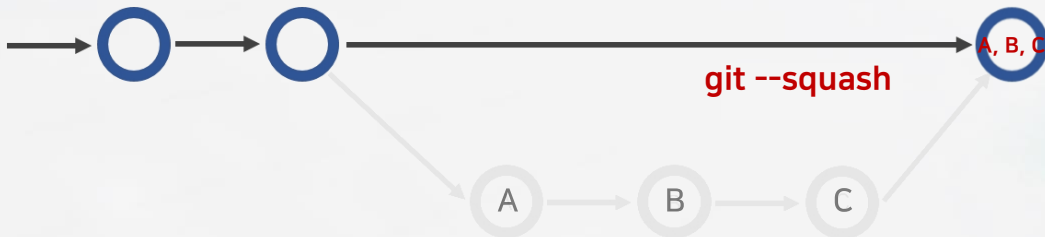
- 현재 브랜치에 병합 대상과의 합치는 커밋을 하나 생성해 병합
 - 병합되는 브랜치는 사용하지 않고 그대로 남음

2 병합의 다양한 옵션

옵션 --squash

☑ 강압적인(?) 병합

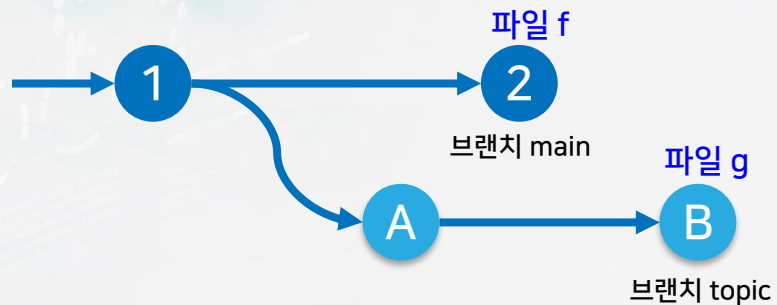
- \$ git merge --squash hotfix
 - 사전적 의미(짓 뭉개다)에서 알 수 있듯이
 - 커밋 이력과 병합되는 브랜치 이력도 남기지 않음
 - 새로운 커밋에 상대 브랜치의 내용을 모두 합해 새로운 커밋으로 병합
- ➡ 직접 메시지와 함께 커밋을 해야 함



2 병합의 다양한 옵션

⚙ 옵션 --squash 병합

☑ 합병 전 상태

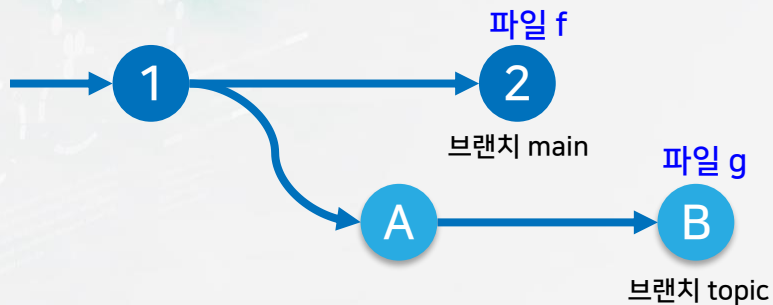


```
$ git log --oneline --graph --all
* 3c3cd0f (HEAD -> main) 2
| * 3d36e77 (topic) B
| * 8b48b8c A
|/
* afd4625 1
```

2 병합의 다양한 옵션

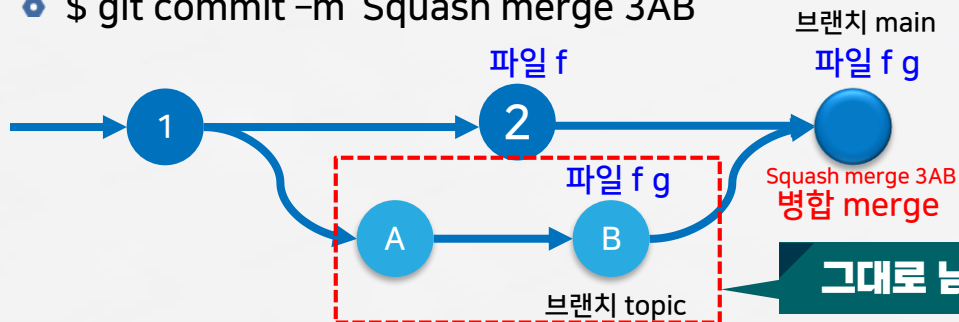
옵션 --squash 병합 실행

수행 전



\$ git merge --squash topic

- \$ git commit -m 'Squash merge 3AB'



그대로 남음

2 병합의 다양한 옵션

병합 명령

--squash

```
파이썬@DESKTOP-8TN3J1L MINGW64 /c/[smart git]/mysqsh (main)
$ git log --oneline --graph --all
* 3c3cd0f (HEAD -> main) 2
| * 3d36e77 (topic) B
| * 8b48b8c A
|/
* afd4625 1
```

```
파이썬@DESKTOP-8TN3J1L MINGW64 /c/[smart git]/mysqsh (main)
$ git merge --squash topic
Automatic merge went well; stopped before committing as requested
Squash commit -- not updating HEAD
```

```
파이썬@DESKTOP-8TN3J1L MINGW64 /c/[smart git]/mysqsh (main)
$ git commit -m 'Squash merge 3AB'
[main 07b1c30] Squash merge 3AB
1 file changed, 2 insertions(+)
create mode 100644 g
```

```
파이썬@DESKTOP-8TN3J1L MINGW64 /c/[smart git]/mysqsh (main)
$ git log --oneline --graph --all
* 07b1c30 (HEAD -> main) Squash merge 3AB
* 3c3cd0f 2
| * 3d36e77 (topic) B
| * 8b48b8c A
|/
* afd4625 1
```

```
파이썬@DESKTOP-8TN3J1L MINGW64 /c/[smart git]/mysqsh (main)
$ ls
f g
```

Summary

» 기존 브랜치에서 hotfix 브랜치 병합

- ◆ \$ git merge hotfix
 - fast-forward, 3-way merge

» 무조건 3-way 병합 수행

- ◆ \$ git merge --no-ff hotfix

» fast-forward인 경우에만 병합 진행

- ◆ \$ git merge --ff-only hotfix

» 현재 브랜치에서 커밋 하나만 생성해서 병합

- ◆ \$ git merge --squash hotfix