

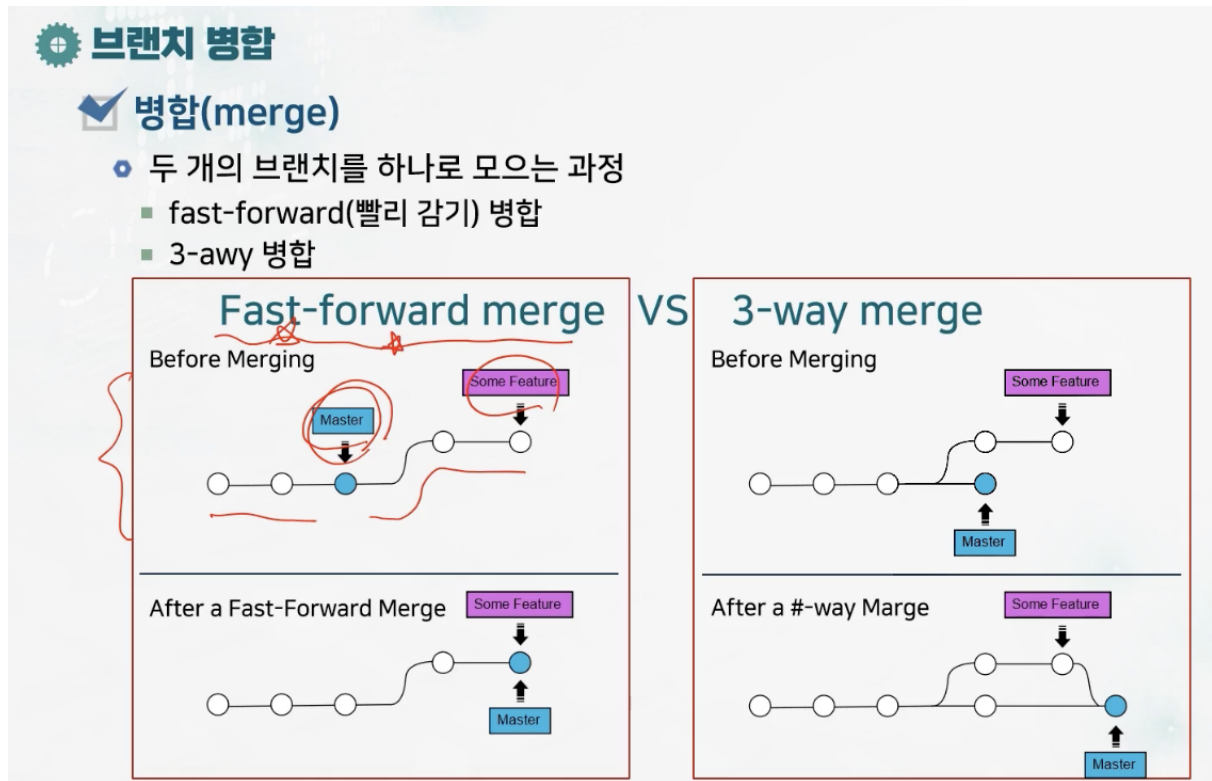
week11

브랜치 병합

브랜치 병합

병합(merge)

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



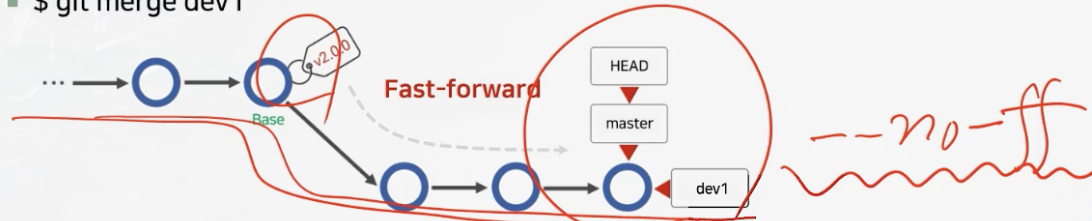
fast-forward : 단순히 브랜치가 이동 / 새로운 커밋 안 생김

3-way merge : 두 브랜치가 하나의 브랜치로 합쳐짐 / 새로운 커밋 생김

기본이 fast forward 상태에서 non fast forward 병합

마스터에서 dev1을 Fast forward로 병합

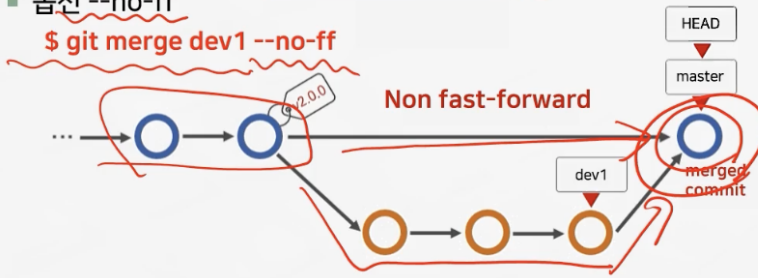
- \$ git merge dev1



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

- 옵션 --no-ff

\$ git merge dev1 --no-ff



—no-ff를 하면 fast forward 상태에서도 3-way 처럼 병합을 할 수 있다~

—no-ff : 무조건 3-way 병합

—ff-only : 무조건 fast-forward 하겠다지만 3-way에선 불가능 해서 일렬 상태에서만 가능

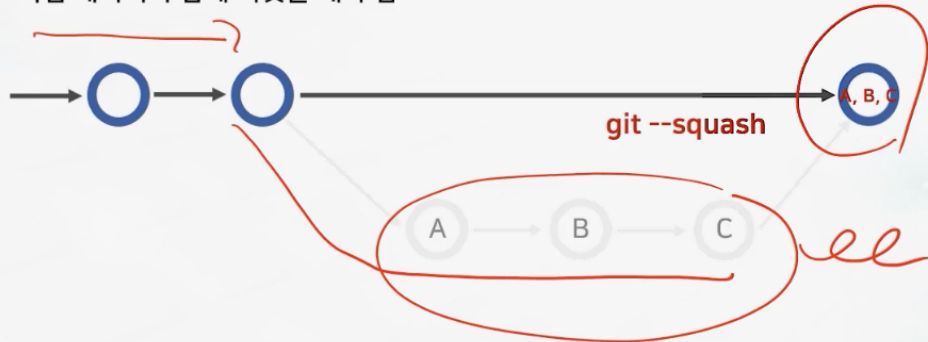
—squash : 현재 브랜치에 병합 대상과의 합치는 커밋을 하나 생성해 병합

병합되는 브랜치는 사용하지 않고 그대로 남음

⚙ 옵션 --squash

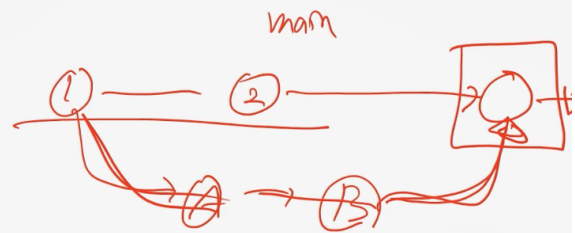
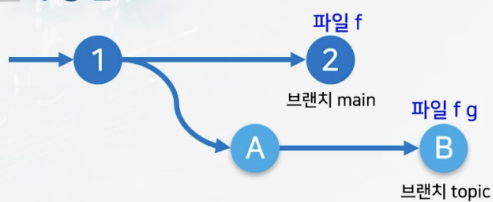
✓ 강압적인(?) 병합

- `$ 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

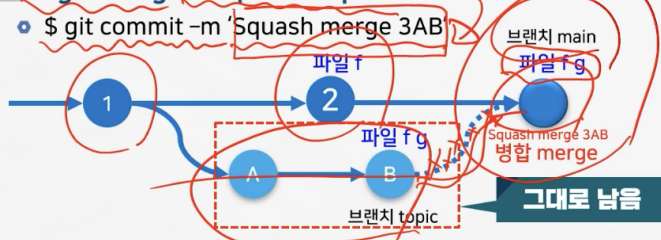


⚙ 옵션 --squash 병합 실행

✓ 수행 전



✓ `$ git merge --squash topic`



병합 충돌과 해결

병합에서 충돌 발생 표기

```
PC@DESKTOP-482NOAB MINGW64 /c/[git tutorial]/branch-lab (main)
```

```
$ git merge feat/list
```

```
Auto-merging basic.py
```

```
CONFLICT (content): Merge conflict in basic.py
```

```
Automatic merge failed; fix conflicts and then commit the result.
```

```
PC@DESKTOP-482NOAB MINGW64 /c/[git tutorial]/branch-lab (main|MERGING)
```

```
$ git log
```

```
2ecae57 (HEAD -> main) add divmod, main
```

```
56361e0 Add print list of range, feat/list
```

```
5be70c1 Add print list, feat/list
```

```
834e72e Add print literals, main
```

```
fab1006 Create basic.py, main
```

충돌 발생 의미

```
PC@DESKTOP-482NOAB MINGW64 /c/[git tutorial]/branch-lab (main|MERGING)
```

```
$ git status
```

```
On branch main
```

```
You have unmerged paths.
```

```
(fix conflicts and run "git commit")
```

```
(use "git merge --abort" to abort the merge)
```

```
Unmerged paths:
```

```
(use "git add <file>..." to mark resolution)
```

```
both modified: basic.py
```

```
no changes added to commit (use "git add" and/or "git commit -a")
```

» 3-way 충돌 발생

- ◆ \$ git merge hotfix

» 충돌한 파일을 인지하고 파일 수정

- ◆ \$ code file

» 수정 후 다시 add, commit

- ◆ \$ git commit -am 'msg'

» 충돌 이후 병합 취소

- ◆ \$ git merge --abort