

# 글로벌 오픈소스 프로젝트 개발 참여 (기본)

## 3. Git/GitHub 고급 (2)

- 협업을 위한 Git/GitHub 기능과 실습
  - Commit History 조작
  - Rebase
  - VSCode를 이용하여 Git/GitHub 사용하기

강사 양석환



**협업을 위한 Git/GitHub 기능**

**Commit History 조작**

## • 커밋 이력 조작의 필요성

- 프로젝트를 진행하다 보면 다양하고 복잡한 환경으로 인해 커밋 이력을 조작할 필요가 생김
- 버전 관리 과정에서 문제가 발생했을 경우 현재의 커밋 상태가 해소되지 않으면 더이상 진행이 되지 않는 경우 등의 상황에서 커밋 이력을 조작함으로써 문제를 벗어날 수 있음

## • 커밋 이력 조작의 종류

- 다른 브랜치의 커밋을 작업 브랜치에 추가하기: **git cherry-pick**
- 이전 커밋으로 작업 브랜치의 최종 커밋 변경하기: **git reset**
- 변경 사항을 되돌리는 커밋 생성하기: **git revert**
- 브랜치 커밋 이력 재정렬하기: **git rebase**



- 다른 브랜치의 커밋을 작업 브랜치에 추가하기

- 두 브랜치를 병합하는 대신 다른 브랜치의 커밋을 선택적으로 작업 브랜치에 추가해야 하는 경우에 사용

- 명령어: `git cherry-pick` “추가하려는 커밋 체크섬”

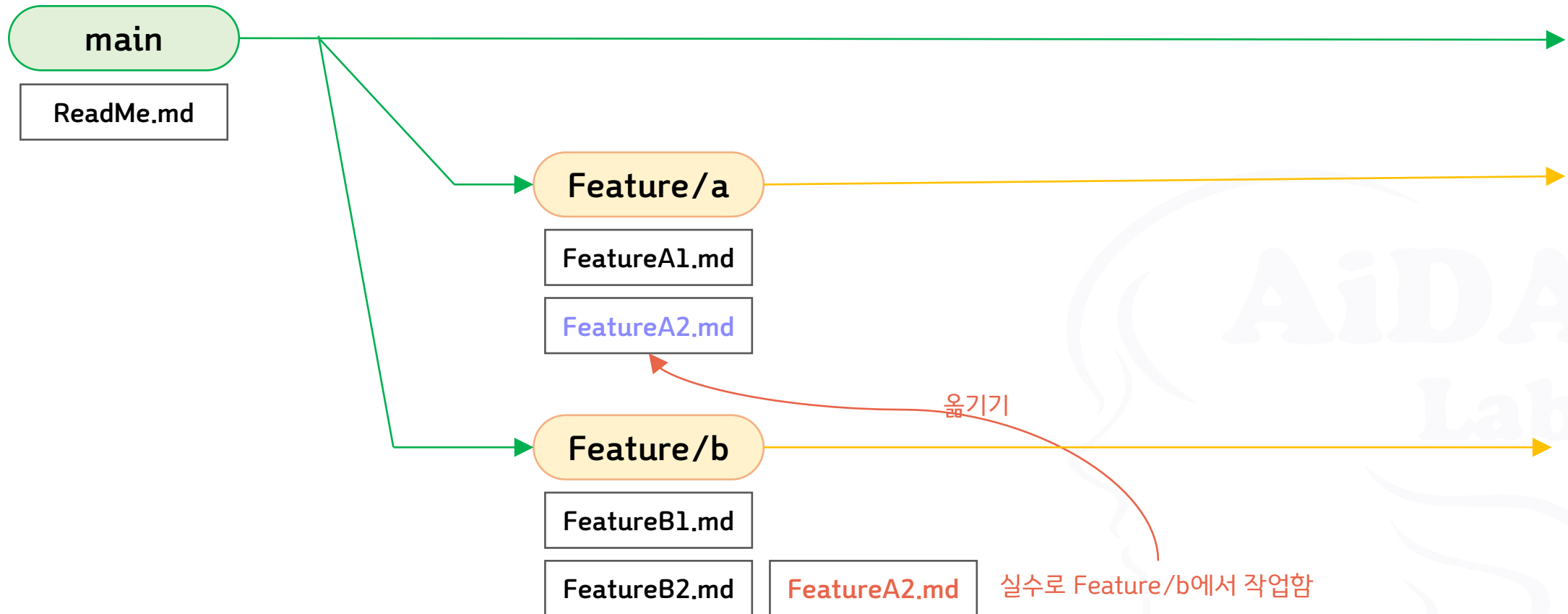
- 활용 예시 1

- 사용자에게 배포된 기능에서 치명적인 결함을 발견함 → 해당 결함으로 인해 사용자는 정상적인 서비스를 이용 불가
- 이미 개발이 완료되어 배포 전 테스트 중인 다른 커밋들과 상관없이
- 결함을 수정한 커밋 만을 서비스 운영에 사용되는 브랜치에 추가

- 활용 예시 2

- 새로운 기능을 개발하여 커밋을 생성하였음
- 커밋 생성 후, 현재 작업 브랜치가 잘못된 것을 발견함
- 의도한 브랜치로 작업 브랜치를 변경한 후, 잘못된 브랜치에서 생성한 커밋을 현재 작업 브랜치에 추가

# Commit History(커밋 이력) 조작하기



# Commit History(커밋 이력) 조작하기

```
터미널  GITLENS  디버그 콘솔  문제  출력
+ v powershell  [icon] [icon] ... ^ x

PS C:\AiDALab\Workspace\ComHist> git init
Initialized empty Git repository in C:/AiDALab/Workspace/ComHist/.git/
• PS C:\AiDALab\Workspace\ComHist> git status
On branch main

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
  •   README.md

nothing added to commit but untracked files present (use "git add" to track)
PS C:\AiDALab\Workspace\ComHist> git add .
• PS C:\AiDALab\Workspace\ComHist> git commit -m "README.md 파일 생성"
[main (root-commit) f288c6f] README.md 파일 생성
 1 file changed, 2 insertions(+)
 create mode 100644 README.md
• PS C:\AiDALab\Workspace\ComHist> git branch Feature/a
• PS C:\AiDALab\Workspace\ComHist> git branch Feature/b
• PS C:\AiDALab\Workspace\ComHist> git checkout Feature/a
Switched to branch 'Feature/a'
PS C:\AiDALab\Workspace\ComHist> git add .
• PS C:\AiDALab\Workspace\ComHist> git commit -m "FeatureA1.md 파일 생성"
[Feature/a 83bff35] FeatureA1.md 파일 생성
 1 file changed, 1 insertion(+)
 create mode 100644 FeatureA1.md
• PS C:\AiDALab\Workspace\ComHist> git checkout Feature/b
Switched to branch 'Feature/b'
PS C:\AiDALab\Workspace\ComHist> git add .
• PS C:\AiDALab\Workspace\ComHist> git commit -m "FeatureB1.md 파일 생성"
[Feature/b 1a66bd3] FeatureB1.md 파일 생성
 1 file changed, 1 insertion(+)
 create mode 100644 FeatureB1.md
• PS C:\AiDALab\Workspace\ComHist> git log --pretty=oneline --graph
* 1a66bd3acb24b64b61ad04e18dbe2237b0adb7ba (HEAD -> Feature/b) FeatureB1.md 파일 생성
* f288c6f8a20445680f3eec76ffca8404582d31a9 (main) README.md 파일 생성
• PS C:\AiDALab\Workspace\ComHist> git checkout Feature/a
Switched to branch 'Feature/a'
• PS C:\AiDALab\Workspace\ComHist> git log --pretty=oneline --graph
* 83bff356de2d88072a45c301373127228f407854 (HEAD -> Feature/a) FeatureA1.md 파일 생성
* f288c6f8a20445680f3eec76ffca8404582d31a9 (main) README.md 파일 생성
• PS C:\AiDALab\Workspace\ComHist>

줄 1, 열 10  공백: 4  UTF-8  CRLF  Markdown  [icon] [icon]
```

- PS C:\AiDALab\Workspace\ComHist> git checkout Feature/b  
Switched to branch 'Feature/b'
- PS C:\AiDALab\Workspace\ComHist> git add .
- PS C:\AiDALab\Workspace\ComHist> git commit -m "FeatureA2.md 파일 생성. 잘못 생성함"
- [Feature/b 09bc139] FeatureA2.md 파일 생성. 잘못 생성함
- 1 file changed, 0 insertions(+), 0 deletions(-)
- PS C:\AiDALab\Workspace\ComHist> git log --pretty=oneline --graph
- \* 09bc13952f4df152da0d57204c3f20d045cf425c (HEAD -> Feature/b) FeatureA2.md 파일 생성. 잘못 생성함
- \* 1a66bd3acb24b64b61ad04e18dbe2237b0adb7ba FeatureB1.md 파일 생성
- \* f288c6f8a20445680f3eec76ffca8404582d31a9 (main) README.md 파일 생성
- PS C:\AiDALab\Workspace\ComHist> git checkout Feature/a
- Switched to branch 'Feature/a'
- PS C:\AiDALab\Workspace\ComHist> git cherry-pick 09bc13952f4df152da0d57204c3f20d045cf425c
- [Feature/a ab62f6b] FeatureA2.md 파일 생성. 잘못 생성함
- Date: Thu May 25 13:57:18 2023 +0900
- 1 file changed, 0 insertions(+), 0 deletions(-)
- create mode 100644 FeatureA2.md
- PS C:\AiDALab\Workspace\ComHist> git log --pretty=oneline --graph
- \* ab62f6b60765484941a93e22dedf507fa5e13406 (HEAD -> Feature/a) FeatureA2.md 파일 생성. 잘못 생성함
- \* 83bff356de2d88072a45c301373127228f407854 FeatureA1.md 파일 생성
- \* f288c6f8a20445680f3eec76ffca8404582d31a9 (main) README.md 파일 생성
- PS C:\AiDALab\Workspace\ComHist> [icon]

- 이전 커밋으로 작업 브랜치의 최종 커밋 변경하기
  - 이미 생성된 커밋을 취소하고 이전 커밋으로 최종 커밋을 변경하고 싶을 때 사용
  - 명령어: `git reset` “이전 커밋 체크섬”

```
PS C:\AiDALab\Workspace\ComHist> git log --pretty=oneline --graph
* ab62f6b60765484941a93e22dedf507fa5e13406 (HEAD -> Feature/a) FeatureA2.md 파일 생성. 잘못 생성함
* 83bff356de2d88072a45c301373127228f407854 FeatureA1.md 파일 생성
* f288c6f8a20445680f3eec76ffca8404582d31a9 (main) ReadMe.md 파일 생성
PS C:\AiDALab\Workspace\ComHist> git reset 83bff356de2d88072a45c301373127228f407854
PS C:\AiDALab\Workspace\ComHist> git log --pretty=oneline --graph
* 83bff356de2d88072a45c301373127228f407854 (HEAD -> Feature/a) FeatureA1.md 파일 생성
* f288c6f8a20445680f3eec76ffca8404582d31a9 (main) ReadMe.md 파일 생성
PS C:\AiDALab\Workspace\ComHist>
```

최종 커밋으로 만들려 하는 커밋 체크섬을 지정함

- 변경 사항을 되돌리는 커밋 생성하기
  - 이미 생성된 커밋을 취소하는 또 다른 방법
  - git reset과의 차이점: 취소하고자 하는 커밋의 변경 사항을 되돌리는 새로운 커밋이 생성됨
- 명령어: git revert “되돌리려는 커밋 체크섬”

git reset 명령어는 취소하려는 커밋 자체를 커밋 이력에서 제외하지만  
git revert는 커밋의 변경 사항을 되돌렸다는 새로운 커밋을 커밋 이력에 생성함

```
PS C:\AiDALab\Workspace\ComHist> git log --pretty=oneline --graph
* 83bff356de2d88072a45c301373127228f407854 (HEAD -> Feature/a) FeatureA1.md 파일 생성
* f288c6f8a20445680f3eec76ffca8404582d31a9 (main) ReadMe.md 파일 생성

PS C:\AiDALab\Workspace\ComHist> git status
On branch Feature/a
Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        deleted:    FeatureA1.md

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        FeatureA2.md
        ReadMe2.md

no changes added to commit (use "git add" and/or "git commit -a")
PS C:\AiDALab\Workspace\ComHist> git add .
PS C:\AiDALab\Workspace\ComHist> git commit -m "ReadMe2.md 파일 생성"
[Feature/a 328e692] ReadMe2.md 파일 생성
3 files changed, 1 insertion(+), 1 deletion(-)
delete mode 100644 FeatureA1.md
create mode 100644 FeatureA2.md
create mode 100644 ReadMe2.md

PS C:\AiDALab\Workspace\ComHist> git log --pretty=oneline --graph
* 328e69218c0f409abfde86147d39b2321a027bd1 (HEAD -> Feature/a) ReadMe2.md 파일 생성
* 83bff356de2d88072a45c301373127228f407854 FeatureA1.md 파일 생성
* f288c6f8a20445680f3eec76ffca8404582d31a9 (main) ReadMe.md 파일 생성

PS C:\AiDALab\Workspace\ComHist> git revert 328e69218c0f409abfde86147d39b2321a027bd1
hint: Waiting for your editor to close the file...
[3264;0525/141127.158;ERROR:service_worker_storage.cc(1933)] Failed to delete the database: Database IO error
[Feature/a 7a7dbe3] Revert "ReadMe2.md 파일 생성"
3 files changed, 1 insertion(+), 1 deletion(-)
create mode 100644 FeatureA1.md
delete mode 100644 FeatureA2.md
delete mode 100644 ReadMe2.md

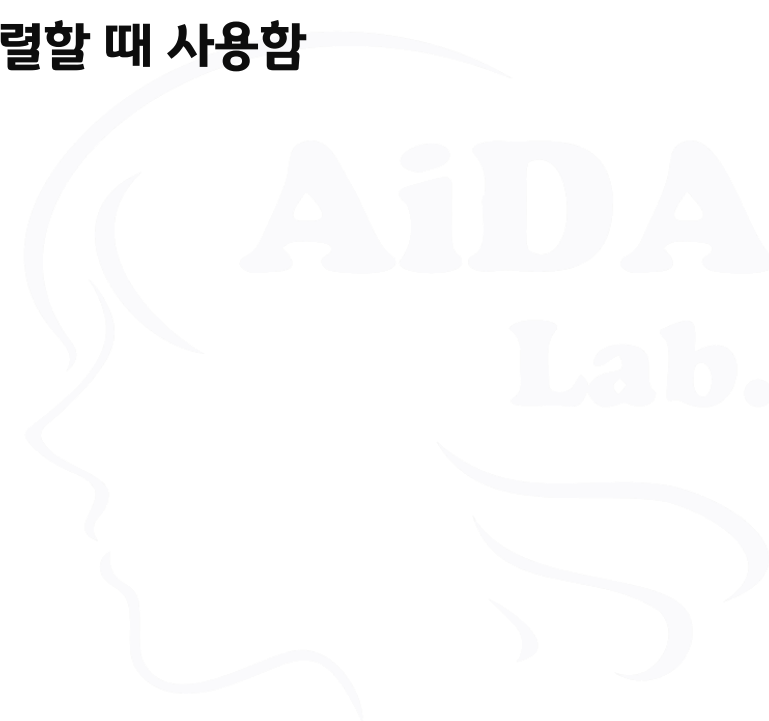
PS C:\AiDALab\Workspace\ComHist>
PS C:\AiDALab\Workspace\ComHist> git log --pretty=oneline --graph
* 7a7dbe33502e3ec2e65aabaef56dc8df67870d (HEAD -> Feature/a) Revert "ReadMe2.md 파일 생성"
* 328e69218c0f409abfde86147d39b2321a027bd1 ReadMe2.md 파일 생성
* 83bff356de2d88072a45c301373127228f407854 FeatureA1.md 파일 생성
* f288c6f8a20445680f3eec76ffca8404582d31a9 (main) ReadMe.md 파일 생성

PS C:\AiDALab\Workspace\ComHist>
```

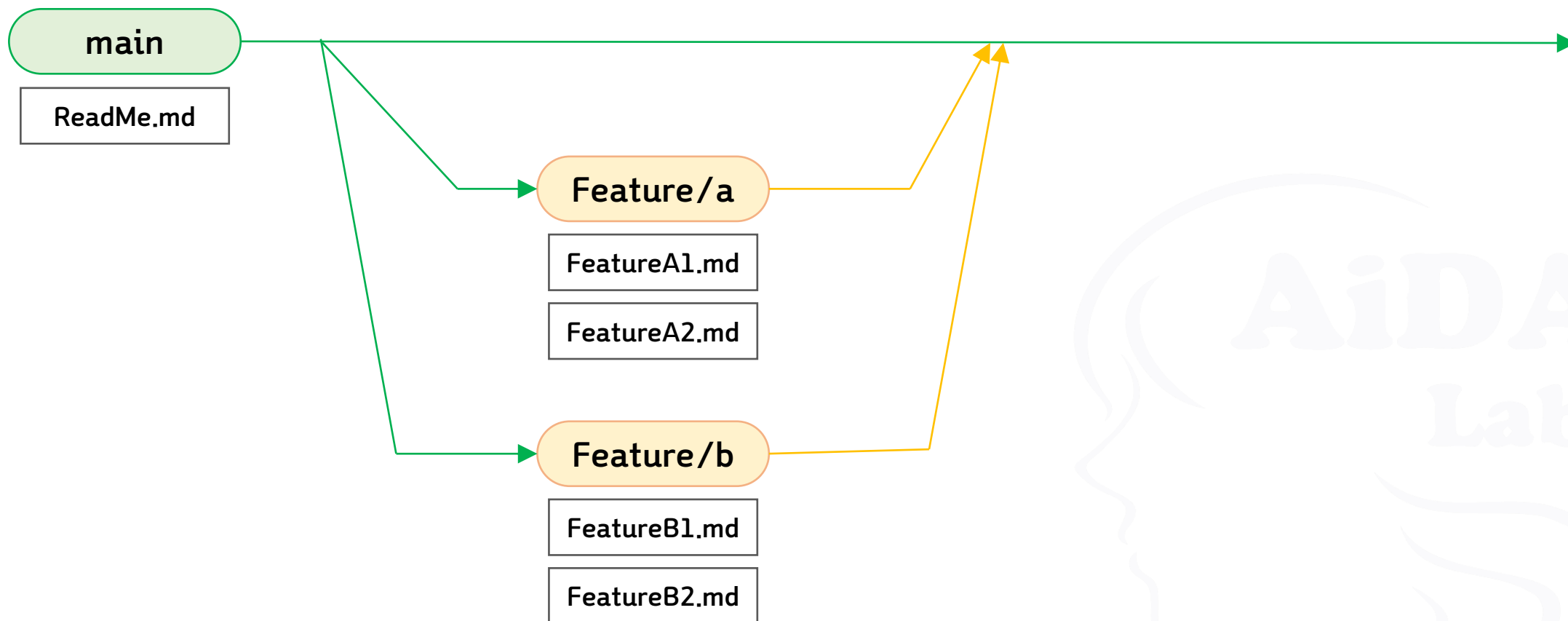


- 브랜치 커밋 이력 재정렬하기

- 기준 브랜치에서 여러 브랜치를 생성하여 작업 후, 계속해서 병합을 할 경우 커밋 이력을 한 눈에 알아보기 어려운 상태가 될 수 있음
- 특정 브랜치의 커밋 이력을 기준으로 작업 브랜치의 커밋 이력을 재정렬할 때 사용함
- 명령어: `git rebase` “재정렬을 위한 기준 브랜치”



- 브랜치 구조



## • 현재 상태

```
PS C:\AiDALab\Workspace\ComHist> git checkout Feature/b
Switched to branch 'Feature/b'
• PS C:\AiDALab\Workspace\ComHist> git status
• On branch Feature/b
Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        deleted:    FeatureA2.md

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        FeatureB2.md

no changes added to commit (use "git add" and/or "git commit -a")
PS C:\AiDALab\Workspace\ComHist> git add .
PS C:\AiDALab\Workspace\ComHist> git commit -m "FeatureA2.md 파일 삭제, FeatureB2.md 파일 생성"
[Feature/b ecfc4d4] FeatureA2.md 파일 삭제, FeatureB2.md 파일 생성
• 2 files changed, 1 insertion(+)
  delete mode 100644 FeatureA2.md
  create mode 100644 FeatureB2.md
PS C:\AiDALab\Workspace\ComHist> git log --pretty=oneline --graph
* ecfc4d47c2e1d5075d253911c047af37e4f44461 (HEAD -> Feature/b) FeatureA2.md 파일 삭제, FeatureB2.md 파일 생성
• * 09bc13952f4df152da0d57204c3f20d045cf425c FeatureA2.md 파일 생성. 잘못 생성함
  * 1a66bd3acb24b64b61ad04e18dbe2237b0adb7ba FeatureB1.md 파일 생성
  * f288c6f8a20445680f3eec76ffca8404582d31a9 (main) ReadMe.md 파일 생성
PS C:\AiDALab\Workspace\ComHist> git checkout Feature/a
Switched to branch 'Feature/a'
• PS C:\AiDALab\Workspace\ComHist> git log --pretty=oneline --graph
• * 66e3057c6cbe2e7ca360e78963ccc6591c4996b3 (HEAD -> Feature/a) FeatureA2.md 파일 생성
  * 7a7dbe33502e3ec2e65aabaee1f56dc8df67870d Revert "ReadMe2.md 파일 생성"
  * 328e69218c0f409abfde86147d39b2321a027bd1 ReadMe2.md 파일 생성
  * 83bff356de2d88072a45c301373127228f407854 FeatureA1.md 파일 생성
  * f288c6f8a20445680f3eec76ffca8404582d31a9 (main) ReadMe.md 파일 생성
```

```
PS C:\AiDALab\Workspace\ComHist> git checkout main
Switched to branch 'main'
• PS C:\AiDALab\Workspace\ComHist> git log --pretty=oneline --graph
  * f288c6f8a20445680f3eec76ffca8404582d31a9 (HEAD -> main) ReadMe.md 파일 생성
• PS C:\AiDALab\Workspace\ComHist> git status
On branch main
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        ReadMe2.md

nothing added to commit but untracked files present (use "git add" to track)
PS C:\AiDALab\Workspace\ComHist> git add .
PS C:\AiDALab\Workspace\ComHist> git commit -m "ReadMe2.md 파일 생성"
[main 9e597ed] ReadMe2.md 파일 생성
• 1 file changed, 1 insertion(+)
  create mode 100644 ReadMe2.md
PS C:\AiDALab\Workspace\ComHist> git log --pretty=oneline --graph
• * 9e597edb5c1b14a43d6e2050b48b303c0d4a58fe (HEAD -> main) ReadMe2.md 파일 생성
  * f288c6f8a20445680f3eec76ffca8404582d31a9 ReadMe.md 파일 생성
○ PS C:\AiDALab\Workspace\ComHist> █
```

## • [main]-[Feature/a] Merge 후 rebase

```
PS C:\AiDALab\Workspace\ComHist> git status
On branch main
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    ReadMe2.md

nothing added to commit but untracked files present (use "git add" to track)
PS C:\AiDALab\Workspace\ComHist> git add .
PS C:\AiDALab\Workspace\ComHist> git commit -m "ReadMe2.md 파일 생성"
[main 9e597ed] ReadMe2.md 파일 생성
1 file changed, 1 insertion(+)
create mode 100644 ReadMe2.md
PS C:\AiDALab\Workspace\ComHist> git log --pretty=oneline --graph
* 9e597edb5c1b14a43d6e2050b48b303c0d4a58fe (HEAD -> main) ReadMe2.md 파일 생성
* f288c6f8a20445680f3eec76ffca8404582d31a9 ReadMe.md 파일 생성
PS C:\AiDALab\Workspace\ComHist>
PS C:\AiDALab\Workspace\ComHist>
PS C:\AiDALab\Workspace\ComHist> git checkout main
Already on 'main'
PS C:\AiDALab\Workspace\ComHist> git merge feature/a
Merge made by the 'ort' strategy.
 FeatureA1.md | 1 +
 FeatureA2.md | 1 +
 2 files changed, 2 insertions(+)
 create mode 100644 FeatureA1.md
 create mode 100644 FeatureA2.md
```

```
PS C:\AiDALab\Workspace\ComHist> git log --pretty=oneline --graph
* a12fca7def2075866998573950314ee6c5dce6e5 (HEAD -> main) Merge branch 'feature/a'
|
| * 66e3057c6cbe2e7ca360e78963ccc6591c4996b3 (Feature/a) FeatureA2.md 파일 생성
| * 7a7dbe33502e3ec2e65aabaee1f56dc8df67870d Revert "ReadMe2.md 파일 생성"
| * 328e69218c0f409abfde86147d39b2321a027bd1 ReadMe2.md 파일 생성
| * 83bff356de2d88072a45c301373127228f407854 FeatureA1.md 파일 생성
| * 9e597edb5c1b14a43d6e2050b48b303c0d4a58fe ReadMe2.md 파일 생성
|/
* f288c6f8a20445680f3eec76ffca8404582d31a9 ReadMe.md 파일 생성
PS C:\AiDALab\Workspace\ComHist> git checkout Feature/a
Switched to branch 'Feature/a'
PS C:\AiDALab\Workspace\ComHist> git rebase main
Successfully rebased and updated refs/heads/Feature/a.
PS C:\AiDALab\Workspace\ComHist> git log --pretty=oneline --graph
* a12fca7def2075866998573950314ee6c5dce6e5 (HEAD -> Feature/a, main) Merge branch 'feature/a'
|
| * 66e3057c6cbe2e7ca360e78963ccc6591c4996b3 FeatureA2.md 파일 생성
| * 7a7dbe33502e3ec2e65aabaee1f56dc8df67870d Revert "ReadMe2.md 파일 생성"
| * 328e69218c0f409abfde86147d39b2321a027bd1 ReadMe2.md 파일 생성
| * 83bff356de2d88072a45c301373127228f407854 FeatureA1.md 파일 생성
| * 9e597edb5c1b14a43d6e2050b48b303c0d4a58fe ReadMe2.md 파일 생성
|/
* f288c6f8a20445680f3eec76ffca8404582d31a9 ReadMe.md 파일 생성
PS C:\AiDALab\Workspace\ComHist> git checkout main
Switched to branch 'main'
PS C:\AiDALab\Workspace\ComHist> git merge Feature/a
Already up to date.
```

- [main]-[Feature/b] Merge 후 rebase

```
PS C:\AiDALab\Workspace\ComHist> git merge Feature/b
Merge made by the 'ort' strategy.
  FeatureB1.md | 1 +
  FeatureB2.md | 1 +
  2 files changed, 2 insertions(+)
● create mode 100644 FeatureB1.md
● create mode 100644 FeatureB2.md
PS C:\AiDALab\Workspace\ComHist> git checkout Feature/b
Switched to branch 'Feature/b'
PS C:\AiDALab\Workspace\ComHist> git rebase main
Successfully rebased and updated refs/heads/Feature/b.
PS C:\AiDALab\Workspace\ComHist> git log --pretty=oneline --graph
● * 40f65c379b58d0d83c183d144ae4727ae24f0373 (HEAD -> Feature/b, main) Merge branch 'Feature/b'
● /
● * ecfc4d47c2e1d5075d253911c047af37e4f44461 FeatureA2.md 파일 삭제, FeatureB2.md 파일 생성
● * 09bc13952f4df152da0d57204c3f20d045cf425c FeatureA2.md 파일 생성. 잘못 생성함
● * 1a66bd3acb24b64b61ad04e18dbe2237b0adb7ba FeatureB1.md 파일 생성
● * | a12fca7def2075866998573950314ee6c5dce6e5 (Feature/a) Merge branch 'feature/a'
● /
● * | 66e3057c6cbe2e7ca360e78963ccc6591c4996b3 FeatureA2.md 파일 생성
● * | 7a7dbe33502e3ec2e65aabaee1f56dc8df67870d Revert "ReadMe2.md 파일 생성"
● * | 328e69218c0f409abfde86147d39b2321a027bd1 ReadMe2.md 파일 생성
● * | 83bff356de2d88072a45c301373127228f407854 FeatureA1.md 파일 생성
● * / 9e597edb5c1b14a43d6e2050b48b303c0d4a58fe ReadMe2.md 파일 생성
● /
● * f288c6f8a20445680f3eec76ffca8404582d31a9 ReadMe.md 파일 생성
PS C:\AiDALab\Workspace\ComHist> git checkout main
Switched to branch 'main'
PS C:\AiDALab\Workspace\ComHist> git merge Feature/b
● Already up to date.
```

- main 브랜치의 최종 커밋 이력

```
PS C:\AiDALab\Workspace\ComHist> git log --pretty=oneline --graph
* 40f65c379b58d0d83c183d144ae4727ae24f0373 (HEAD -> main, Feature/b) Merge branch 'Feature/b'
|
| * ecfc4d47c2e1d5075d253911c047af37e4f44461 FeatureA2.md 파일 삭제, FeatureB2.md 파일 생성
| * 09bc13952f4df152da0d57204c3f20d045cf425c FeatureA2.md 파일 생성. 잘못 생성함
| * 1a66bd3acb24b64b61ad04e18dbe2237b0adb7ba FeatureB1.md 파일 생성
| * a12fca7def2075866998573950314ee6c5dce6e5 (Feature/a) Merge branch 'feature/a'
| |
| | * 66e3057c6cbe2e7ca360e78963ccc6591c4996b3 FeatureA2.md 파일 생성
| | * 7a7dbe33502e3ec2e65aaba1f56dc8df67870d Revert "ReadMe2.md 파일 생성"
| | * 328e69218c0f409abfde86147d39b2321a027bd1 ReadMe2.md 파일 생성
| | * 83bff356de2d88072a45c301373127228f407854 FeatureA1.md 파일 생성
| |
| | /
| / 9e597edb5c1b14a43d6e2050b48b303c0d4a58fe ReadMe2.md 파일 생성
| /
| * f288c6f8a20445680f3eec76ffca8404582d31a9 ReadMe.md 파일 생성
PS C:\AiDALab\Workspace\ComHist>
```

**협업을 위한 Git/GitHub 기능**

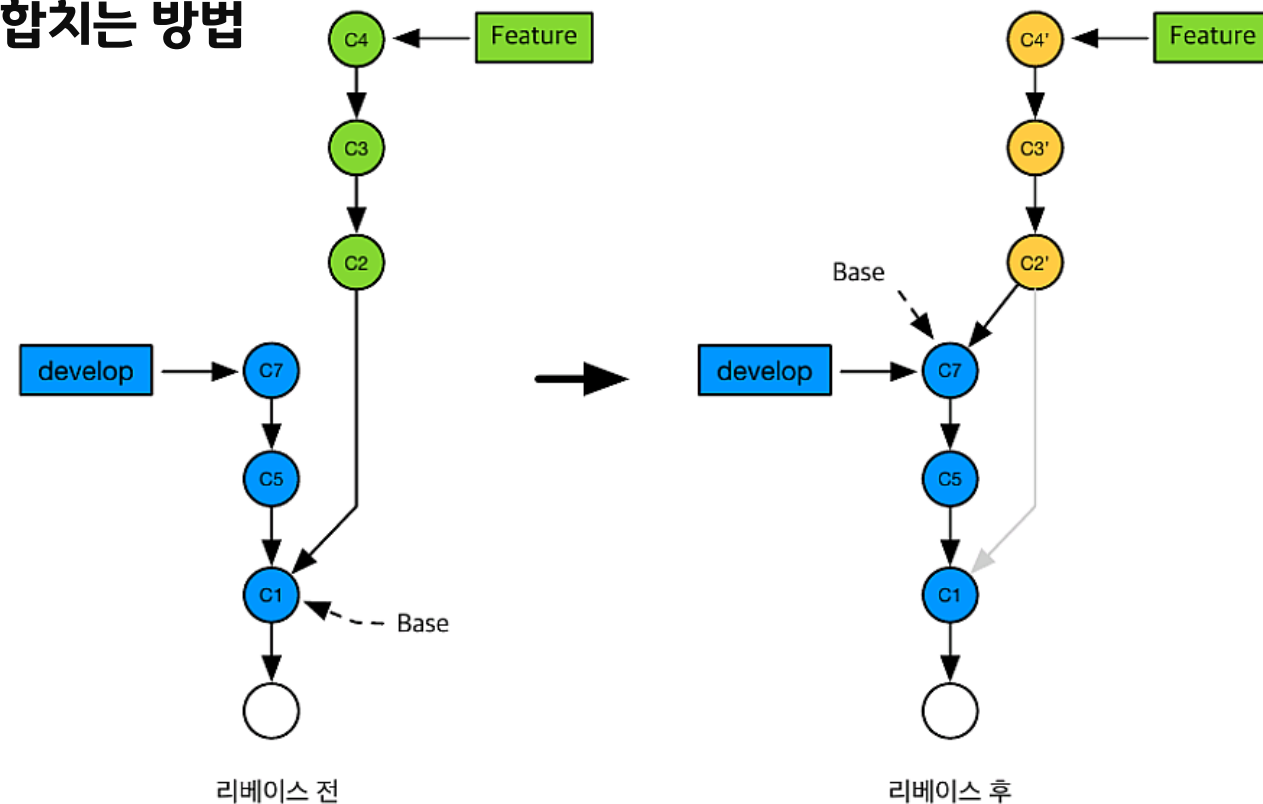
**Rebase**

## • Rebase란?

- Base: 브랜치의 시작점이 되는 브랜치(주로 main 또는 master)
- Rebase: 브랜치의 Base를 변경하여 브랜치를 합치는 방법

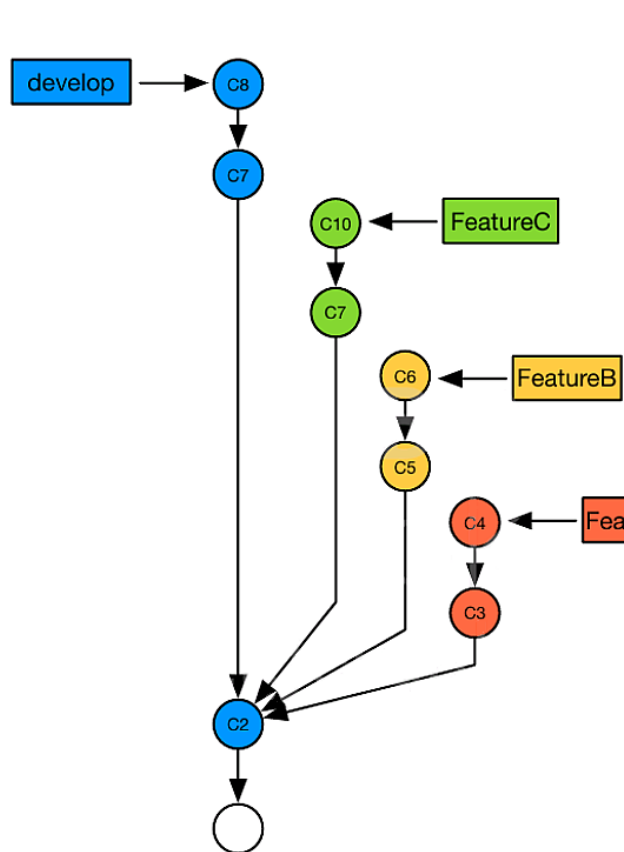
- 오른쪽의 그림에서

- Feature 브랜치의 Base는 C1 커밋
- develop 브랜치는 C7까지 작업됨
- Feature 브랜치는 C1 이후로 C5, C7과는 독립적으로(중복되지 않고) 작업이 진행되었다면
- Feature와 develop 브랜치를 합치기 위하여 Feature의 Base를 C1 → C7로 변경
- 두 브랜치는 일렬로 된 형태로 병합됨

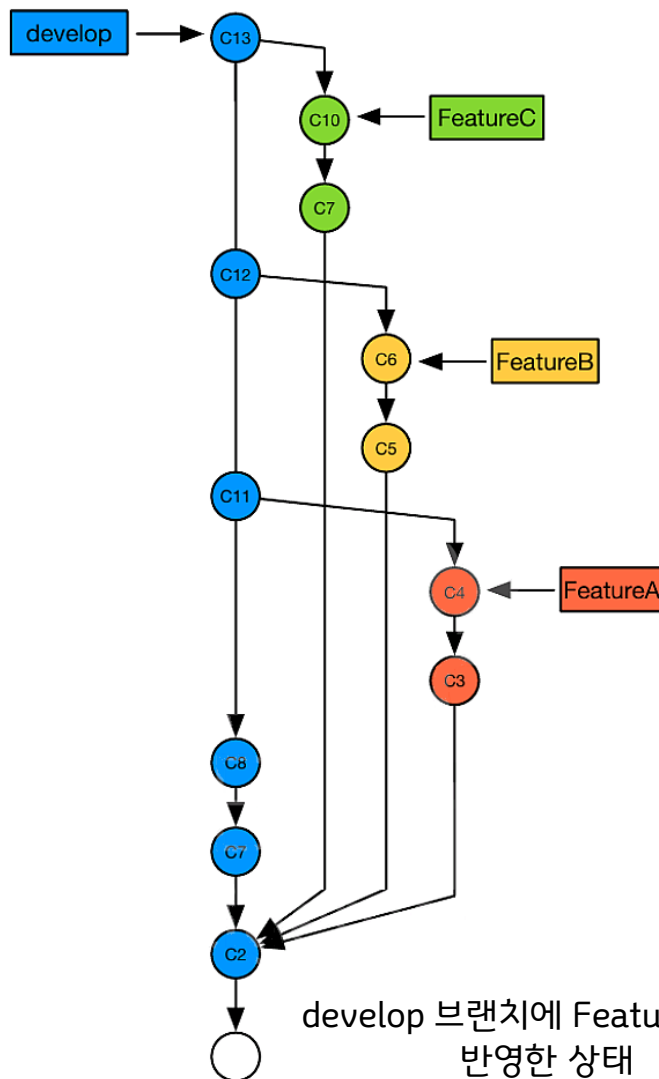




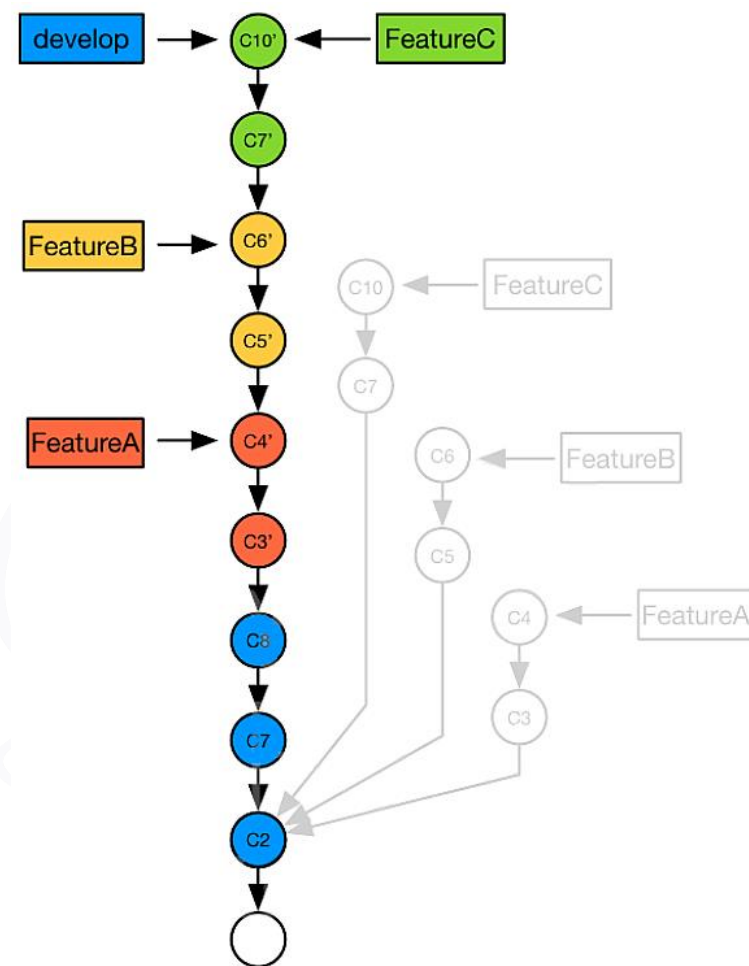
# Rebase (Base 재지정)



순서대로 Merge 시키면 이런 모양



develop 브랜치에 FeatureA~C를 반영한 상태



Rebase를 통해서 각 브랜치를 Merge-Commit 생성 없이 일렬로 정리한 모습

## • Rebase의 동작

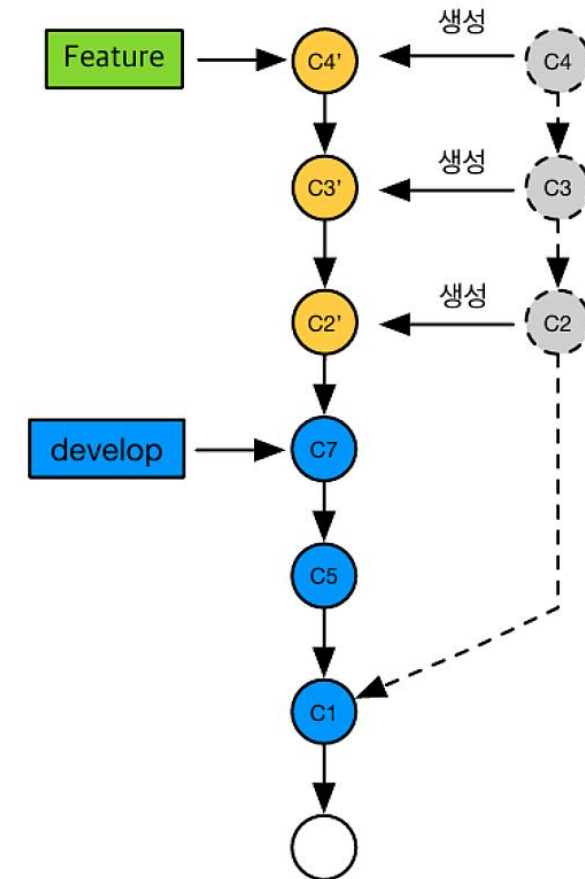
### • Rebase를 수행하면

- 내부적으로는 각 커밋별로 패치를 생성
- 순서대로 다른 브랜치에 적용

### • Rebase 후

- 커밋이 원본 브랜치에서 다른 브랜치로 base만 바뀌어서 이동한 것으로 보이지만
- 실제로는 같은 메시지와 변경 사항을 포함한 새로운 커밋이 생성되어 적용된 것

동일한 파일과 동일한 메시지를 가진 커밋을 새로 생성하면 전혀 다른 커밋이 되는 이유는?  
커밋 생성 시 볼 수 있는 커밋 ID는 SHA-1 알고리즘에 의해 생성되므로 생성 시간과 base가 변경  
되면서 커밋 ID 자체가 변경되기 때문에 전혀 다른 커밋으로 인식됨



Feature 브랜치를 develop에  
Rebase하는 과정

## • Rebase 시 주의 사항

- Rebase는 원본 커밋이 버려지고 새로운 커밋이 생성되기 때문에 개인(지역) 저장소에서만 수행해야 함
  - 공용(원격) 저장소에서 수행하면 커밋 히스토리가 꼬이게 됨
  - 예: A는 Rebase 이전의 커밋에 이어서 커밋을 생성하고, B는 Rebase 후의 커밋에 이어서 작업을 했다만 두 커밋 히스토리는 일반적인 방법으로는 병합 불가 상태에 빠지게 됨
- 꼭 공용(원격) 저장소에서 Rebase를 수행해야만 하는 경우에는
  - 모든 협업자에게 전달하여 현재까지의 작업내용을 모두 커밋하고 공용(원격) 저장소에 Push 할 것을 요청
  - Rebase 작업 중에는 다른 작업을 Push 하지 않도록 요청
  - Rebase가 완료되면 모든 협업자에게 전달하여 전원이 새로운 저장소의 내용을 Pull(Checkout) 하도록 요청

## • Merge와 Rebase의 차이점

### • 커밋 히스토리의 변경

- Fast Forward Merge: 커밋의 변경 또는 추가가 없음
- Three Way Merge: 새로운 커밋은 추가하지만 이전 커밋의 변경은 없음 → 언제든지 예전 커밋 상태로 복귀 가능
- Rebase: 옛날 커밋을 버리고 새로운 커밋을 생성 → 일반적인 방법으로는 예전 커밋 상태로 복귀 불가

### • 명령 수행 횟수

- Merge: 단 한 번으로 완료함 → 충돌이 발생한 경우에도 단 한 번의 처리로 해결 가능
- Rebase: Rebase되는 커밋을 각각 재생성하여 적용 → 충돌 발생 시, 여러 번에 걸쳐 나누어서 처리해야 함

### • 커밋 히스토리의 복잡도

- Three Way Merge가 Rebase보다 훨씬 복잡함 → Rebase를 사용하는 이유

Merge와 Rebase는 모두 브랜치를 병합한다는 공통된 작업을 수행함

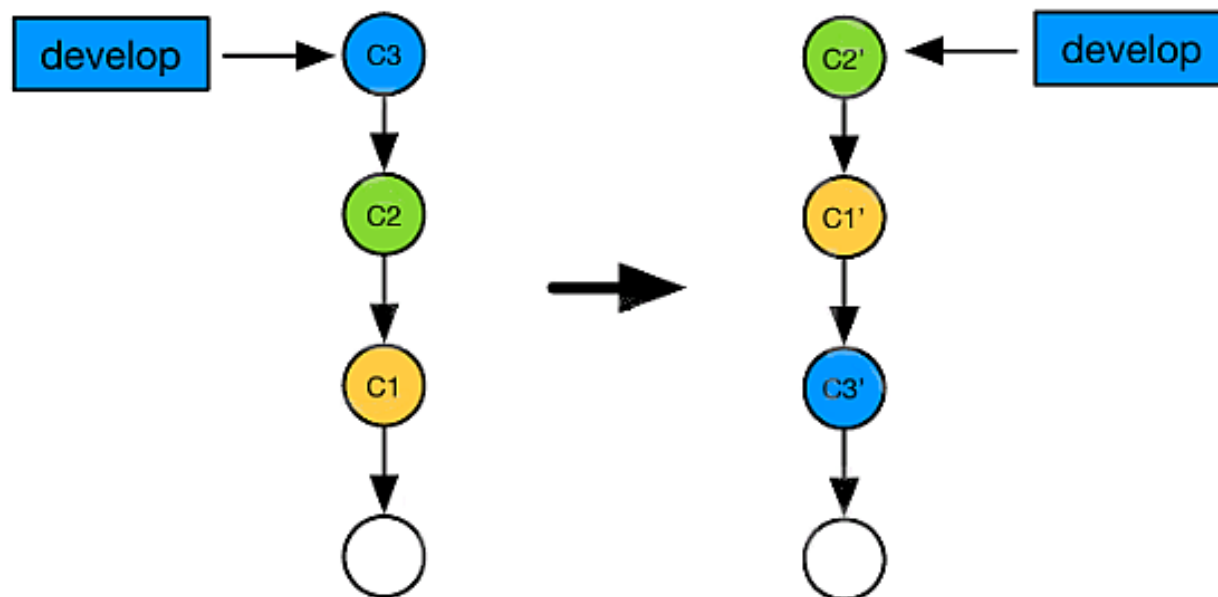
Three Way Merge

두 개의 브랜치를 병합할 때 공통 조상을 기준으로 하는 방식

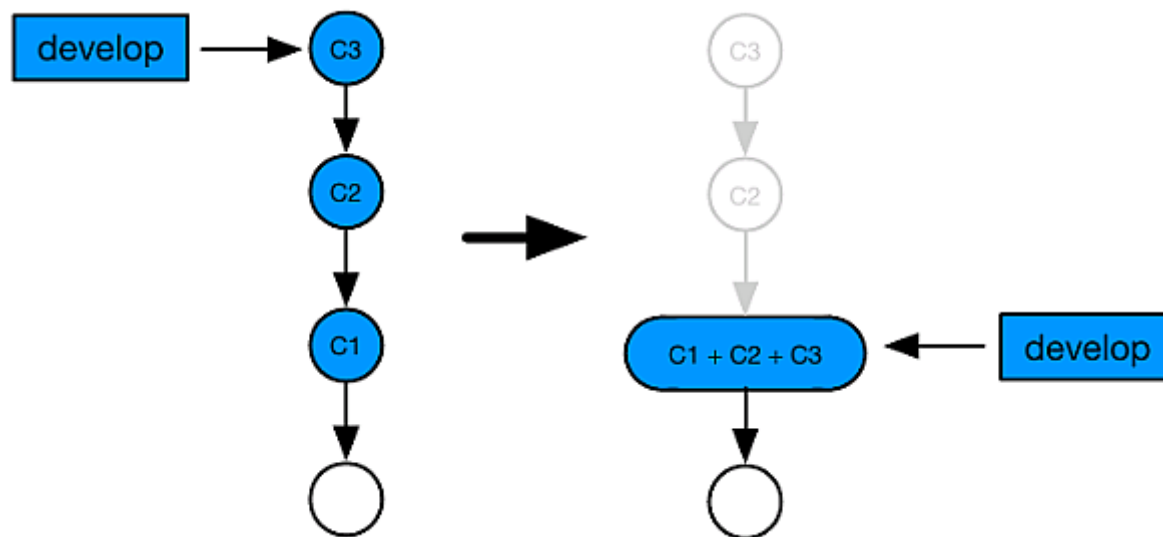
- Rebase의 활용 방안
  - 커밋 메시지 변경하기



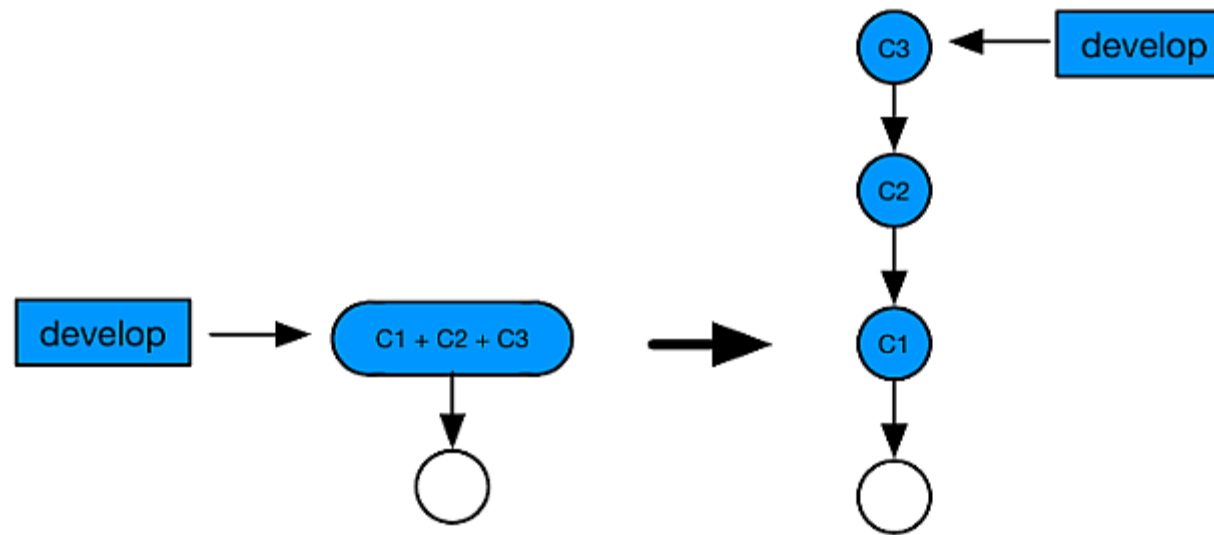
- Rebase의 활용 방안
  - 커밋 순서 변경하기



- Rebase의 활용 방안
  - 커밋 합치기

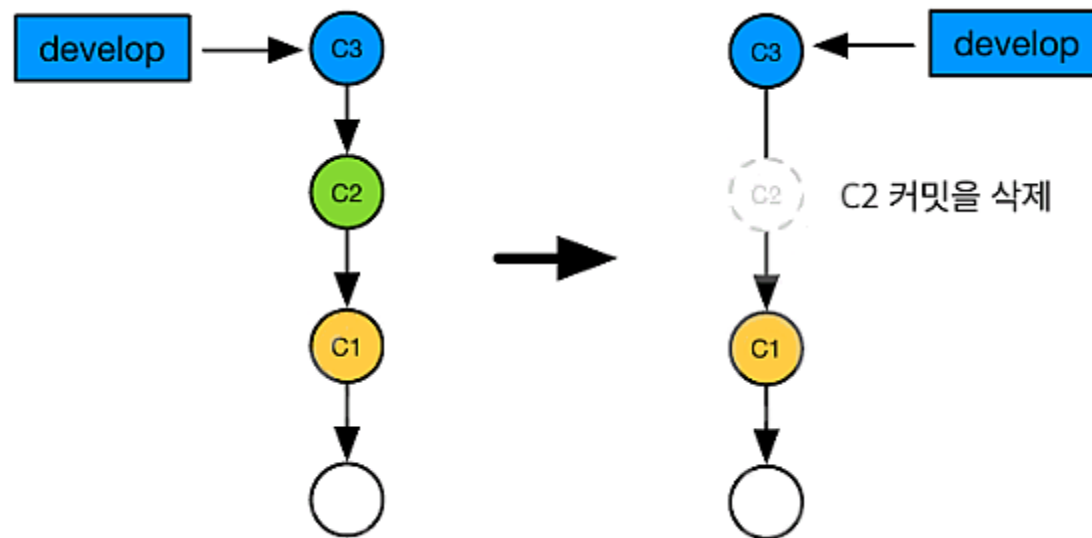


- Rebase의 활용 방안
  - 커밋 분리하기





- Rebase의 활용 방안
  - 커밋 삭제하기

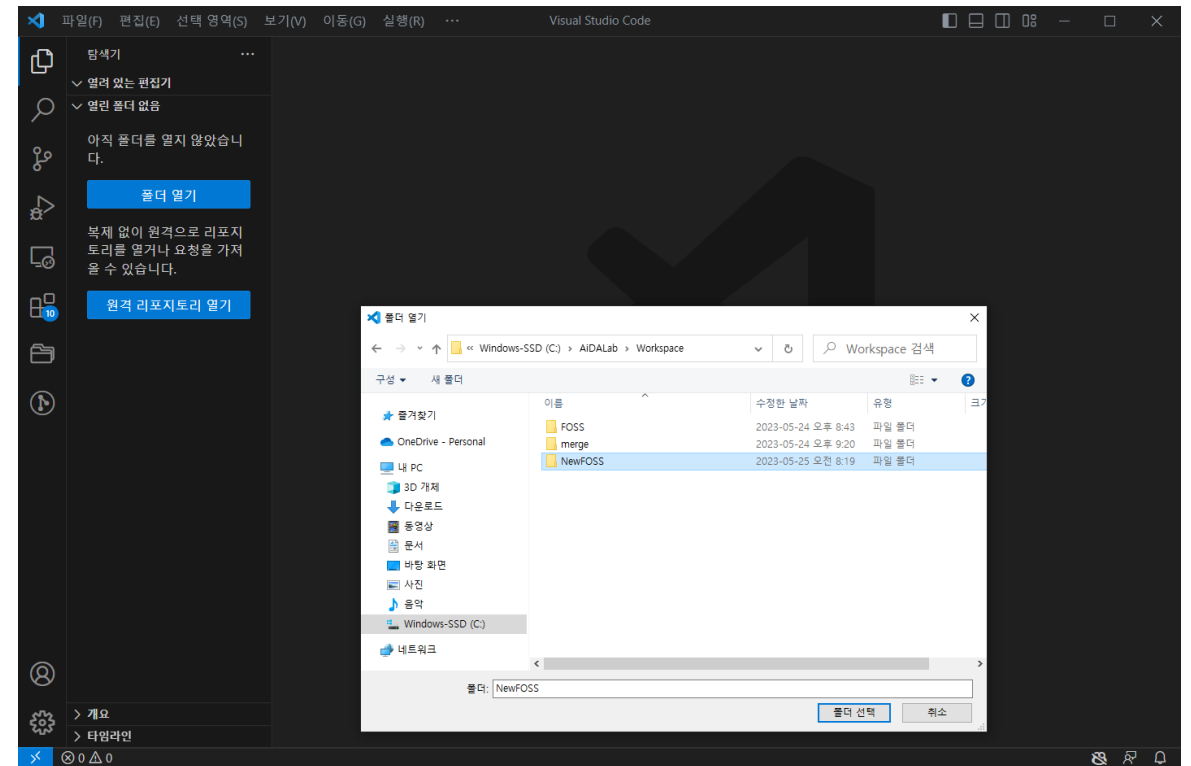
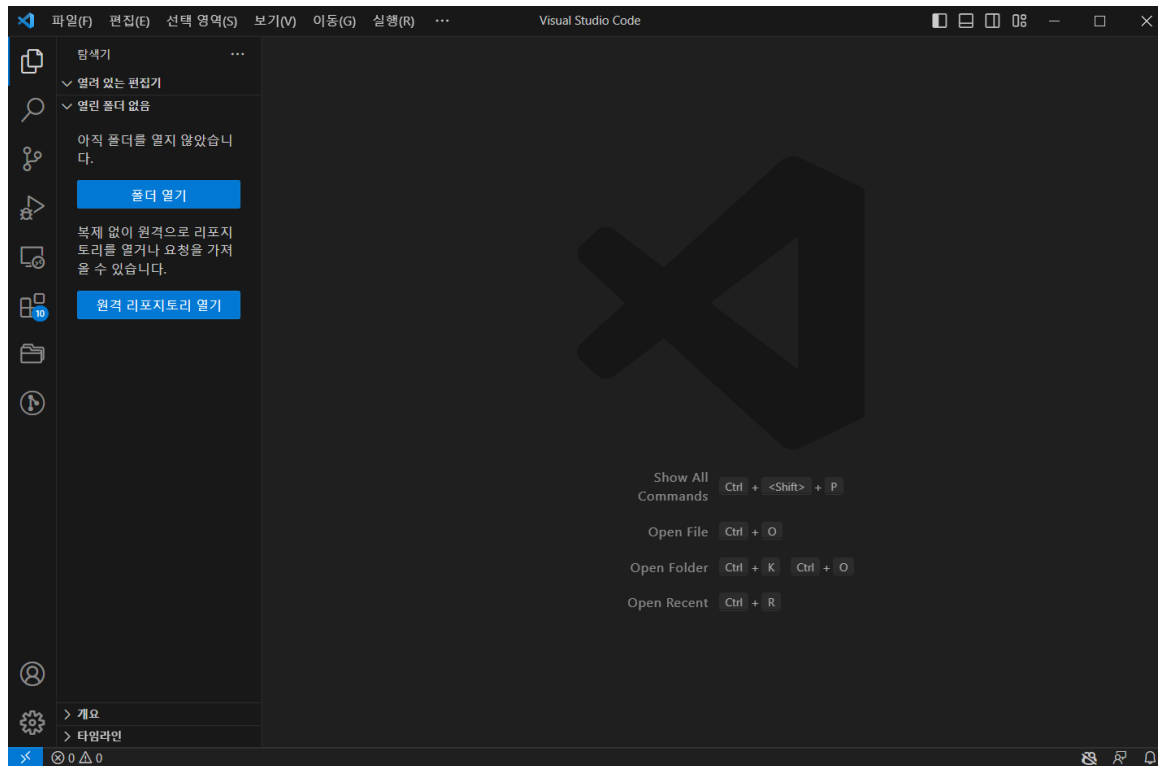


**협업을 위한 Git/GitHub 기능**

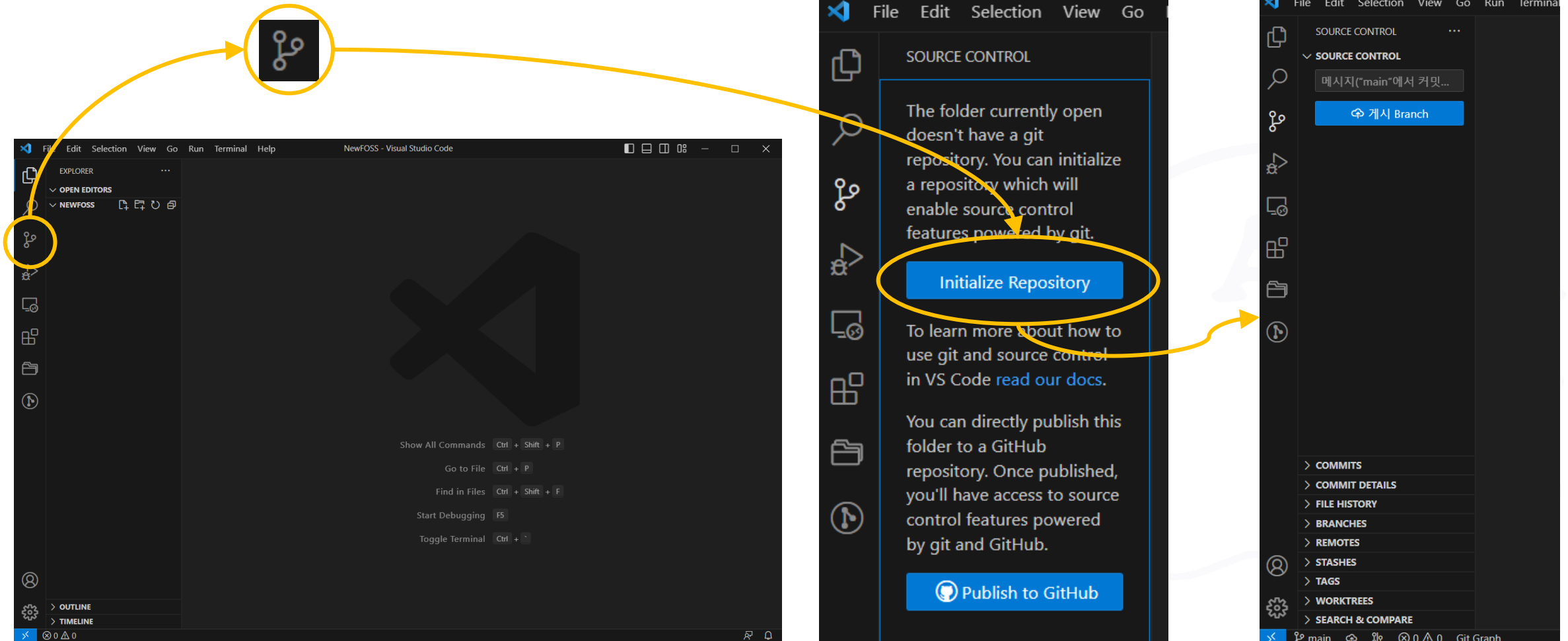
**VSCode를 이용하여 Git/GitHub 사용하기**

## • 폴더 열기

- 없으면 새 폴더 만들어서 열기



- 소스 코드 버전 관리 버튼 > Initialize Repository (최초 사용 시)



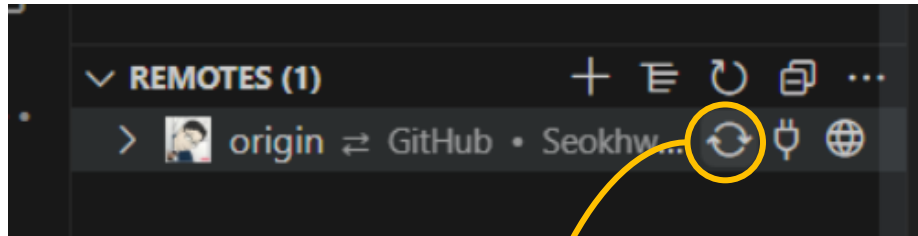
## • 소스 코드 버전 관리 버튼 > REMOTES > + 아이콘 >

저장소 명: origin  
저장소 주소 : <https://github.com/SeokhwanYang/FOSS.git>

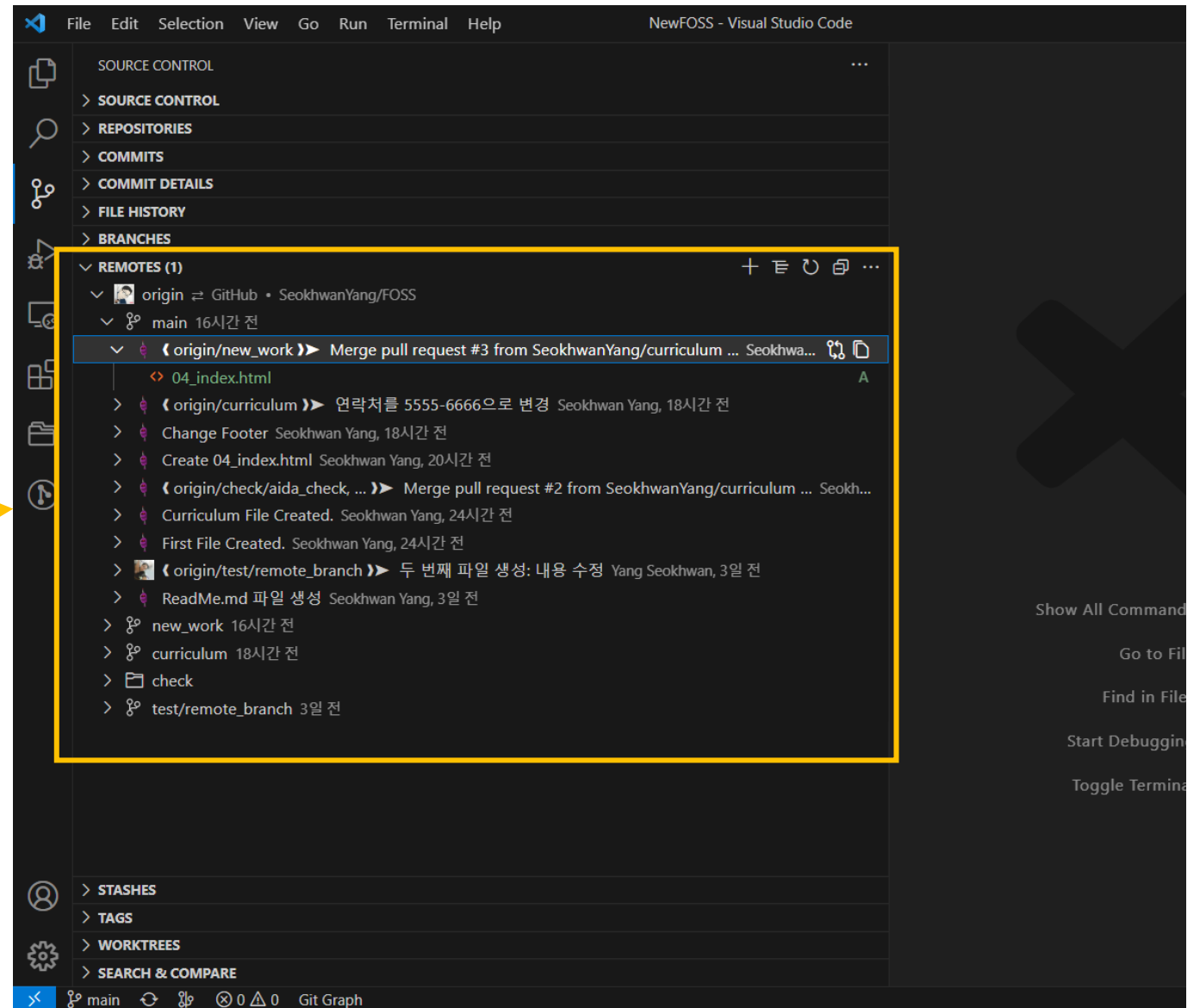
의 순서대로 입력 후  
Add Remote 또는 Add Remote and Fetch -f 를 선택함

REMOTES (1)  
> origin ⇄ GitHub • SeokhwanYang/FOSS

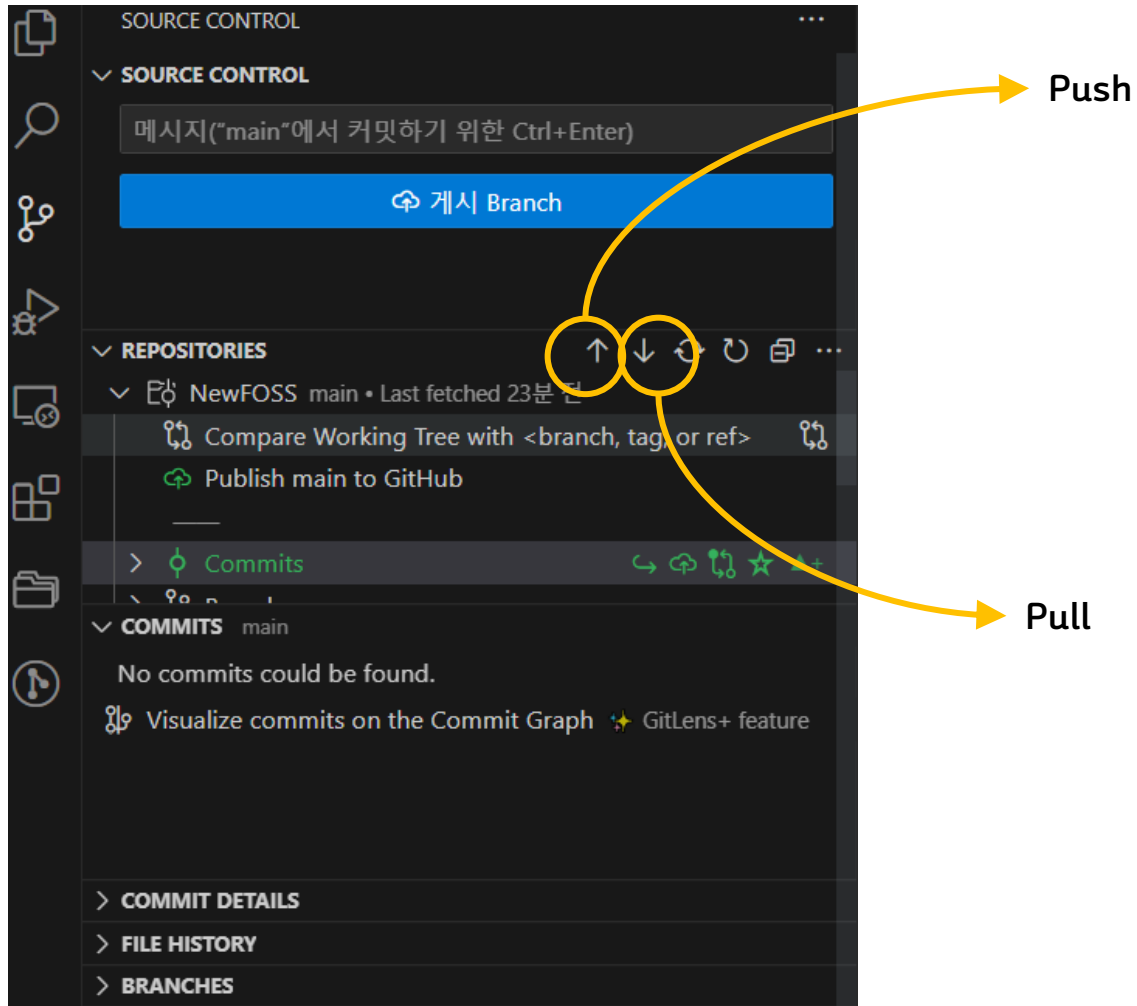
# VSCode & Git/GitHub



“Fetch”를 클릭해서 정보 수신



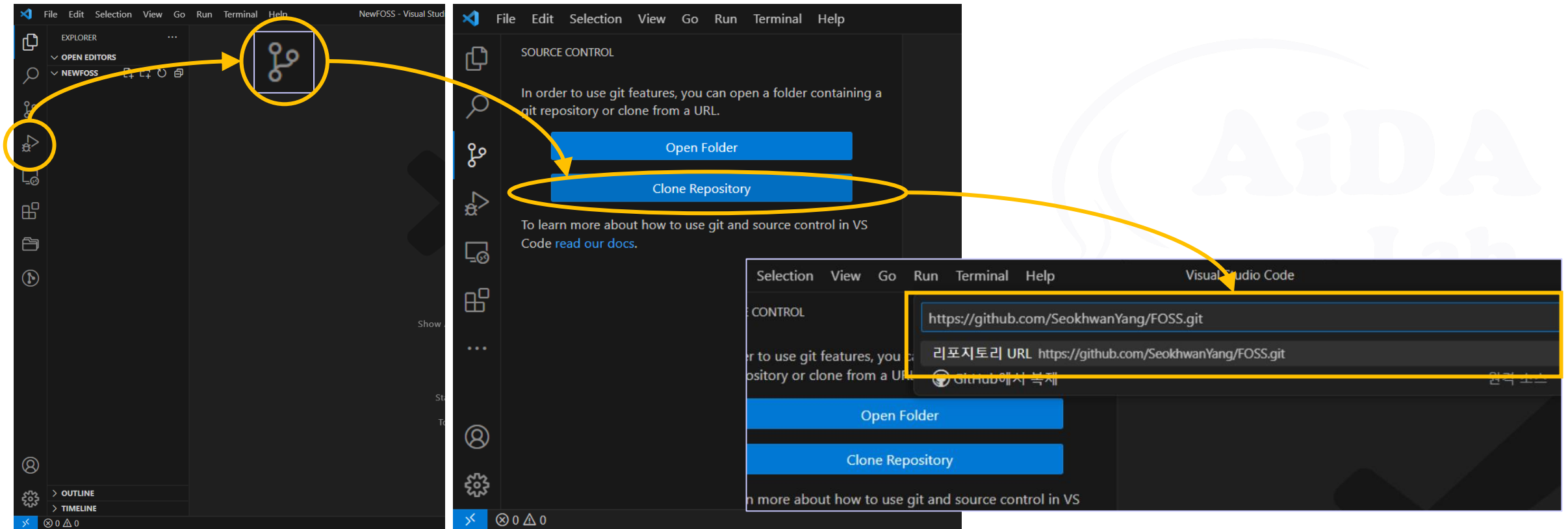
## • Push / Pull



이 내용들은 VSCode의 터미널을 이용하여  
CLI로 처리해도 동일한 결과를 얻을 수 있음

## • Clone Repogitory

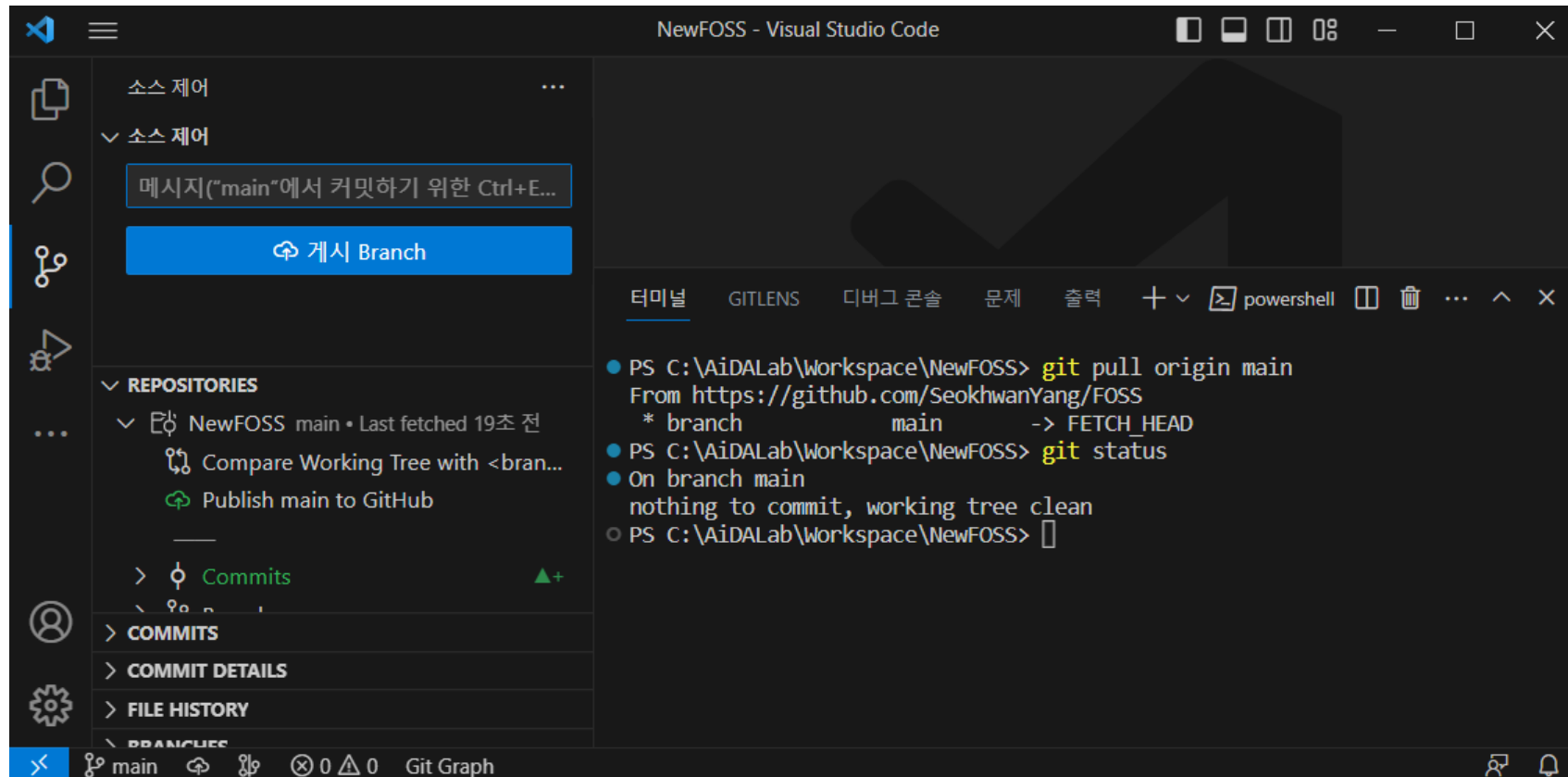
- 모든 폴더를 닫은 상태에서 수행함





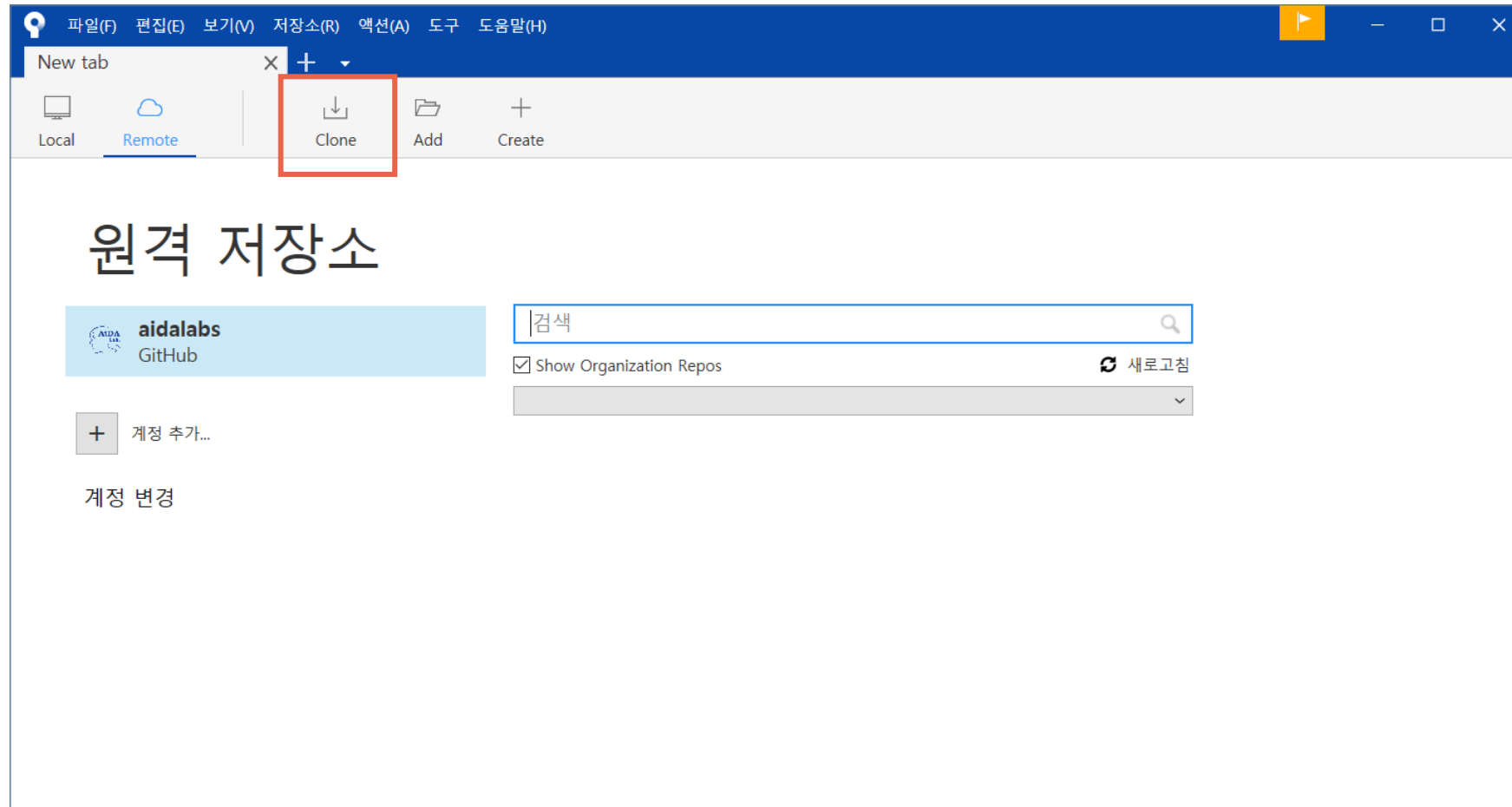
## • VSCode의 터미널 이용하기

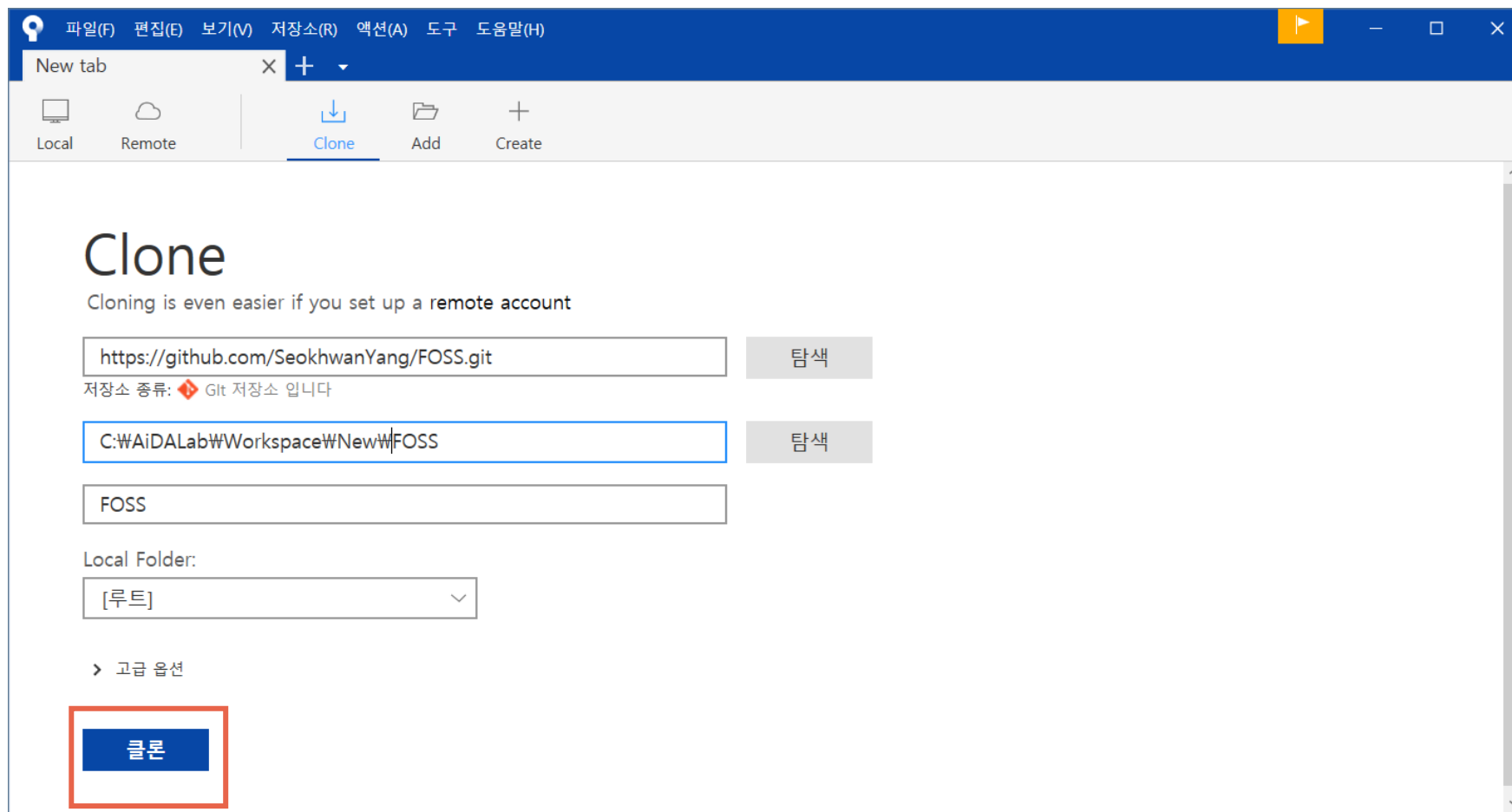
- Ctrl + ` (ESC 키 아래에 있는 키를 말함) 를 눌러서 터미널을 활성화 시킴



**협업을 위한 Git/GitHub 기능**

**SourceTree를 이용하여 Git/GitHub 사용하기**





파일(F) 편집(E) 보기(V) 저장소(R) 액션(A) 도구 도움말(H)

FOSS

커밋 Pull Push 패치 브랜치 병합 스테시 폐기 태그

깃 플로우... 원격 터미널 탐색기 설정

WORKSPACE

파일 상태

History

Search

브랜치

main

태그

원격

스테시

모든 브랜치 원격 브랜치 날짜순으로 정렬

그래프

설명

날짜

작성자

커밋

건너뛰기:

main origin/new\_work origin/main origin/HEAD Merge pull request #3 from 24 5 2023 17:20 Seokhwan Yang 53f4872

origin/curriculum 연락처를 5555-6666으로 변경 24 5 2023 15:48 Seokhwan Yang 820e5d7

Change Footer 24 5 2023 15:02 Seokhwan Yang 1908cb4

Create 04\_index.html 24 5 2023 13:48 Seokhwan Yang baa4009

origin/check/pr\_check\_branch origin/check/aida\_check Merge pull request #2 from Seokhwan 24 5 2023 11:11 Seokhwan Yang f0daa46

Curriculum File Created. 24 5 2023 9:54 Seokhwan Yang ef89f47

First File Created. 24 5 2023 9:53 Seokhwan Yang 52e86f6

origin/test/remote\_branch 두 번째 파일 생성: 내용 수정 22 5 2023 15:24 Yang Seokhwan 43a46a6

ReadMe.md 파일 생성 22 5 2023 14:29 Seokhwan Yang 1a7c90c

파일 상태순 정렬

04\_index.html

커밋: 53f48728af2042f2e598fd9fcd9eebe094230932 [53f4872]

상위 항목: f0daa46786, 820e5d7803

작성자: Seokhwan Yang <yang.seokhwan@gmail.com>

날짜: 2023년 5월 24일 수요일 오후 5:20:48

커밋한 사람: GitHub

Merge pull request #3 from SeokhwanYang/curriculum

04\_index.html

파일 내용

코드창치 되돌리기

```

1 + <html>
2 +
3 + <head>
4 + <title>강의계획서 ver.1</title>
5 + <link rel="stylesheet" href="/stylesheets/style.css">
6 + </head>
7 +
8 + <body>
9 + <h1>강의계획서</h1>
10 + <p>글로벌 오픈소스 프로젝트 참여</p>
11 + </body>
12 +
13 + <footer>
14 + <p>연락처: 5555-6666</p>

```

파일(F) 편집(E) 보기(V) 저장소(R) 액션(A) 도구 도움말(H)

FOSS

커밋 Pull Push 패치 브랜치 병합 스테시 폐기 태그

깃 플로우... 원격 터미널 탐색기 설정

WORKSPACE

파일 상태

History

Search

브랜치

main

태그

원격

origin

curriculum

HEAD

check

main

new\_work

test

스테시

모든 브랜치 원격 브랜치 날짜순으로 정렬

그래프

설명

날짜

작성자

커밋

건너뛰기:

main origin/new\_work origin/main origin/HEAD Merge pull request #3 from 24 5 2023 17:20 Seokhwan Yang < 53f4872

origin/curriculum 연락처를 5555-6666으로 변경 24 5 2023 15:48 Seokhwan Yang < 820e5d7

Change Footer 24 5 2023 15:02 Seokhwan Yang < 1908cb4

Create 04\_index.html 24 5 2023 13:48 Seokhwan Yang < baa4009

origin/check/pr\_check\_branch origin/check/aida\_check Merge pull request #2 from Seokhwan 24 5 2023 11:11 Seokhwan Yang < f0daa46

Curriculum File Created. 24 5 2023 9:54 Seokhwan Yang < ef89f47

First File Created. 24 5 2023 9:53 Seokhwan Yang < 52e86f6

origin/test/remote\_branch 두 번째 파일 생성: 내용 수정 22 5 2023 15:24 Yang Seokhwan < 43a46a6

ReadMe.md 파일 생성 22 5 2023 14:29 Seokhwan Yang < 1a7c90c

파일 상태순 정렬

04\_index.html

커밋: 53f48728af2042f2e598fd9fcd9eebe094230932 [53f4872]

상위 항목: f0daa46786, 820e5d7803

작성자: Seokhwan Yang <yang.seokhwan@gmail.com>

날짜: 2023년 5월 24일 수요일 오후 5:20:48

커밋한 사람: GitHub

Merge pull request #3 from SeokhwanYang/curriculum

04\_index.html

파일 내용

코드루치 되돌리기

```

1 + <html>
2 +
3 + <head>
4 + .....<title>강의계획서 ver.1</title>
5 + .....<link rel="stylesheet" href="/stylesheets/style.css">
6 + </head>
7 +
8 + <body>
9 + .....<h1>강의계획서</h1>
10 + .....<p>글로벌 오픈소스 프로젝트 참여</p>
11 + </body>
12 +
13 + <footer>
14 + .....<p>연락처: 5555-6666</p>

```

THANK  
YOU

