

05. 협업- branch, fork, merge ..

1. VScode란

2. 기초셋팅

3. 버전관리

4. Vscode GIT협업하기

용어정리

fork	포크로 찍어서 가져오기
upstream	(상류)원본위치
pull	(차이만)땡겨옴
pull request(PR)	수정사항 보내기
conflict	충돌
merge	합치기

03. Visual Studio Code GIT협업

1. VScode란

2. 기초셋팅

3. 버전관리

4. Vscode GIT협업하기

기본
셋팅



반복
작업

03. Visual Studio Code GIT협업

1. VScode란

2. 기초셋팅

3. 버전관리

4. Vscode GIT협업하기

1. 협업의 기본셋팅

팀장

자신의 로컬에서
프로젝트 기본셋팅



GitHub 레포지토리 업로드

팀원

팀장 레포지토리를
fork해옴



fork해온것 **본인 로컬에 clone**하기



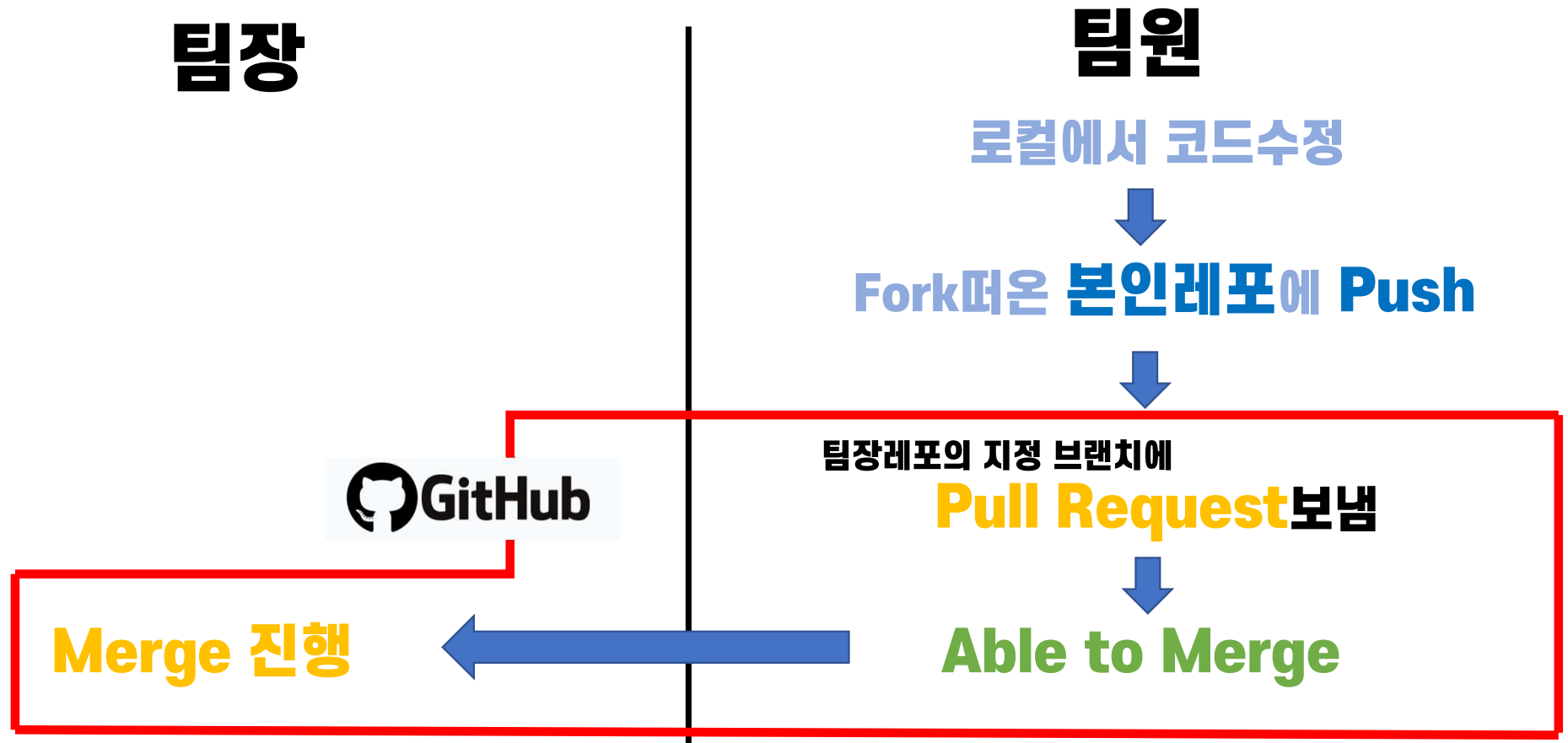
1. VScode란

2. 기초셋팅

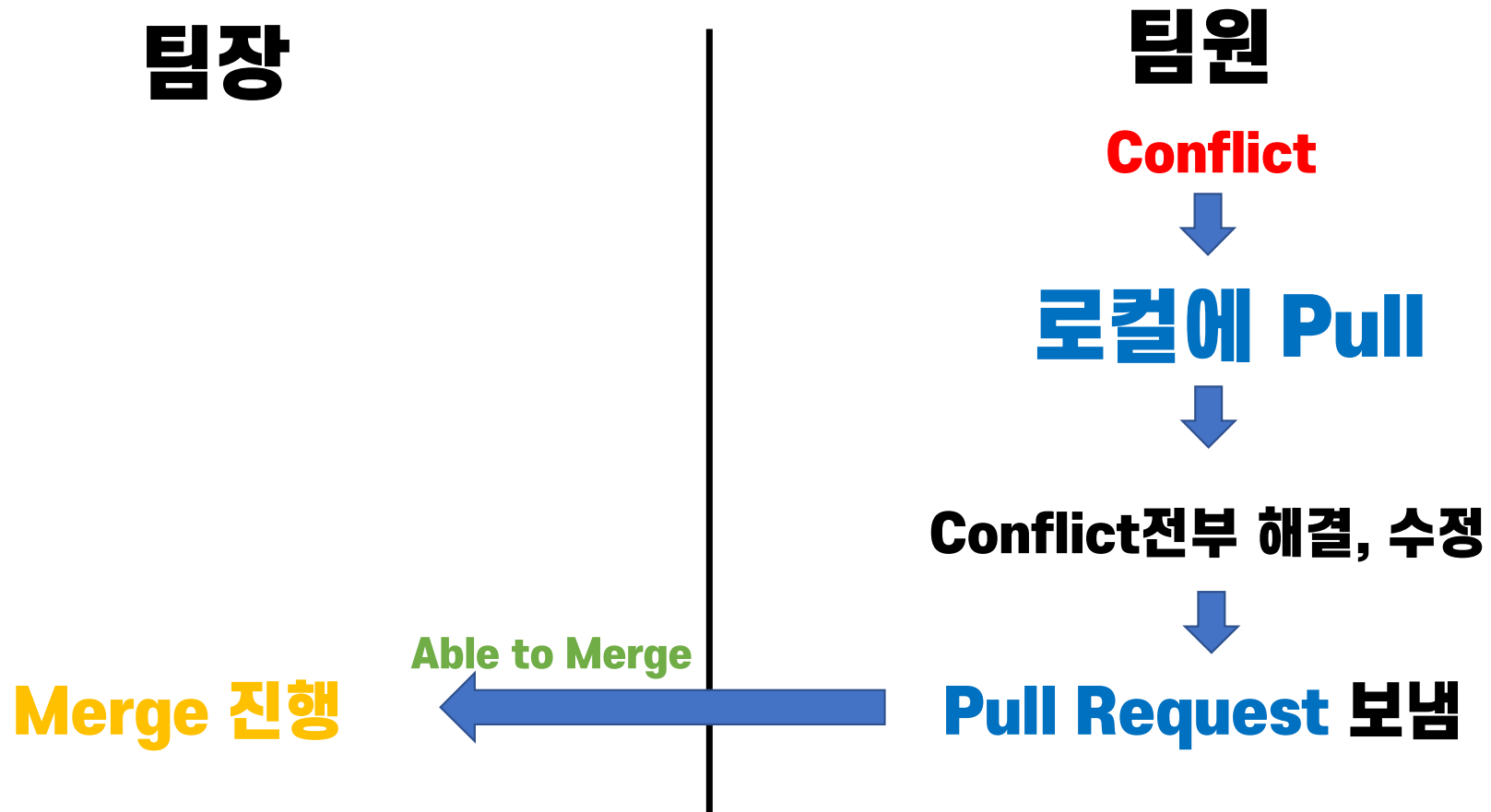
3. 버전관리

4. Vscode GIT협업하기

2. 팀원이 만든 코드 합치기



3. Conflict 해결하기



03. Visual Studio Code GIT협업

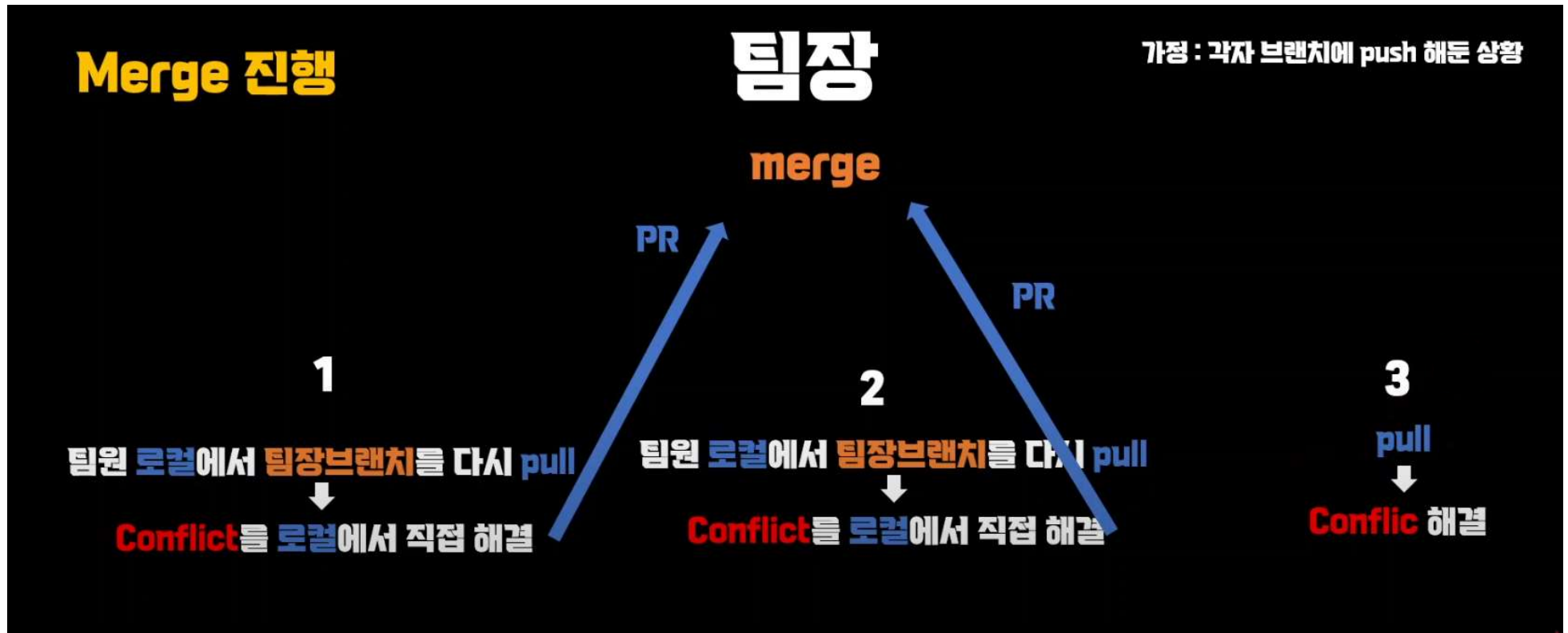
1. VScode란

2. 기초셋팅

3. 버전관리

4. Vscode GIT협업하기

4. 여러명과 협업하기



03. Visual Studio Code GIT협업

1. VScode란

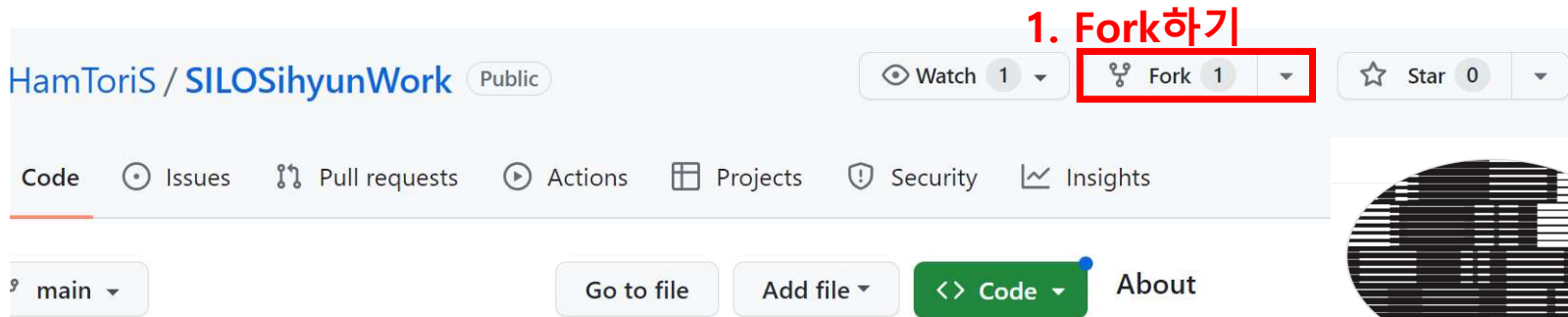
2. 기초셋팅

3. 버전관리

4. Vscode GIT협업하기

1. 외부저장소 fork 하기

1. Fork하기



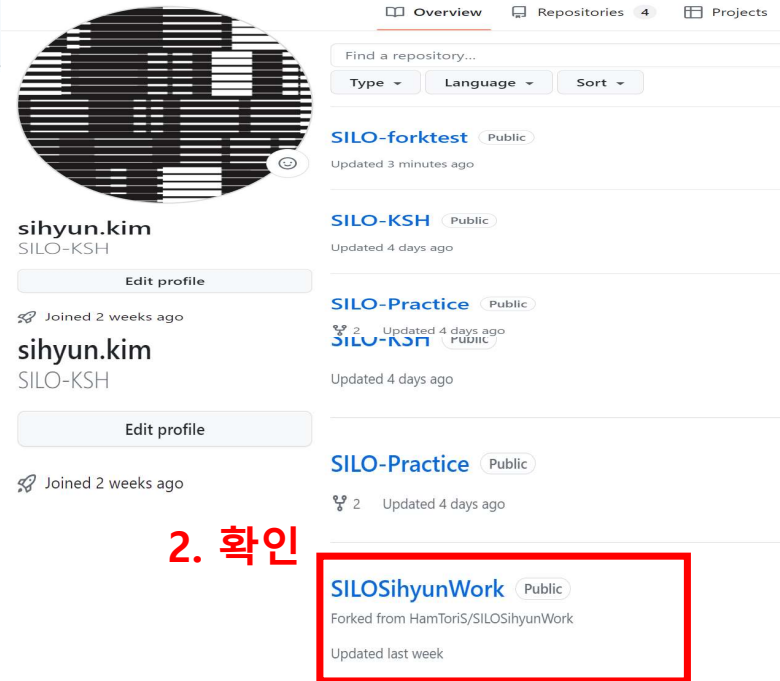
HamToriS / SILOsihyunWork Public

Watch 1 Fork 1 Star 0

Code Issues Pull requests Actions Projects Security Insights

main Go to file Add file Code About

2. 확인



Overview Repositories 4 Projects

Find a repository...

Type Language Sort

SILO-forktest Public Updated 3 minutes ago

SILO-KSH Public Updated 4 days ago

SILO-Practice Public Updated 4 days ago

SILO-KSH Public Updated 4 days ago

SILO-Practice Public Updated 4 days ago

SILOsihyunWork Public Forked from HamToriS/SILOsihyunWork Updated last week

HamToriS 사람의 SILOsihyunWork라는 레퍼지토리를 Fork 하고난후, 확인해보니 내 레퍼지토리로 그대로 들어옴.

03. Visual Studio Code GIT협업

1. VScode란

2. 기초셋팅

3. 버전관리

4. Vscode GIT협업하기

2. Clone하기

앞에서 배운 Clone내용과 같다.

03. Visual Studio Code GIT협업

1. VScode란

2. 기초셋팅

3. 버전관리

4. Vscode GIT협업하기

3. 로컬 저장소에 원본 저장소 등록 (upstream)

```
$ git remote -v
```

위 명령어를 입력해 로컬 저장소와 연결된 원격저장소를 확인한다.

```
C:\Users\silolab_ksh\Desktop\CloneHamTori\SILOsihyunWork>git remote -v
origin https://github.com/SILO-KSH/SILOsihyunWork.git (fetch)
origin https://github.com/SILO-KSH/SILOsihyunWork.git (push)
```

```
$ git remote add upstream [원격 저장소 주소]
```

원본 원격 저장소를 등록하기 위해 위 명령어를 입력해준다.
보통 fork에서 원본저장소의 이름은 upstream을 사용한다.

```
C:\Users\silolab_ksh\Desktop\CloneHamTori\SILOsihyunWork>git remote -v
origin https://github.com/SILO-KSH/SILOsihyunWork.git (fetch)
origin https://github.com/SILO-KSH/SILOsihyunWork.git (push)
upstream https://github.com/HamToriS/SILOsihyunWork.git (fetch)
upstream https://github.com/HamToriS/SILOsihyunWork.git (push)
```

실습(4) –fork ,clone

1. <https://github.com/SILO-KSH/SILO-forktest> 들어가서 fork하기
2. fork한 레포지토리를 자신 로컬에 Clone하기
3. git remote -v명령어를 통해 확인하기
4. 원본저장소 연결하기
5. git remote -v명령어를 통해 확인하기

03. Visual Studio Code GIT협업

1. VScode란

2. 기초셋팅

3. 버전관리

4. Vscode GIT협업하기

4. 로컬에서 작업하는 브랜치 만들기

branch명령어 참고

내용	명령어
브랜치 확인하기	git branch
새 브랜치 만들기	git branch [브랜치이름]
다른브랜치로 이동하기	git checkout [이동할브랜치이름]
브랜치삭제하기	git branch -d [삭제할브랜치이름]
브랜치에 코드 push하기	git push origin [브랜치이름]
merge하기(합치기)	git merge [합칠브랜치이름]

브랜치이름예시 :develop

```
#브랜치 생성
git branch develop

#다른브랜치로 이동
git checkout develop

#현재연결된브랜치확인
git branch
*develop
main
```

```
C:\Users\silolab_ksh\Desktop\Clone\Boss>git branch
* develop
main
```

03. Visual Studio Code GIT협업

1. VScode란

2. 기초셋팅

3. 버전관리

4. Vscode GIT협업하기

5. Commit & Push

로컬 저장소에서 수정한 내용을 내 브랜치로 commit & push 해준다.

1. 현재 연결된 브랜치 확인

2. 원하는 브랜치로 이동

3. `git push origin 브랜치이름` 명령어를 통해 push하기

ex) `git push origin develop`

03. Visual Studio Code GIT협업

1. VScode란

2. 기초셋팅

3. 버전관리

4. Vscode GIT협업하기

5. Pull request생성

SILO-KSH / Boss Public

forked from HamToriS/Boss

Code Pull requests Actions Projects Wiki Security Insights Settings

develop had recent pushes less than a minute ago [Compare & pull request](#)

develop 2 branches 0 tags [Go to file](#) [Add file](#) [Code](#)

This branch is 2 commits ahead of HamToriS:main. [Contribute](#) [Sync fork](#)

HamToriS t1

File	Commit
README.md	Initial commit
Test1.txt	t1

예시파일

This branch is 2 commits ahead of HamToriS:main.
Open a pull request to contribute your changes upstream.

[Open pull request](#)

03. Visual Studio Code GIT협업

1. VScode란

2. 기초셋팅

3. 버전관리

4. Vscode GIT협업하기

5. Pull request생성

Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#).

base repository: HamToriS/Boss

base: main

head repository: SILO-KSH/Boss

compare: develop

✓ Able to merge. These branches

연습 PR

Write

Preview

H B I ≡ <> 🔗 ≡ ≡ ≡ @ ↗ ↶

Pull Request 연습

commet 남기기

Attach files by dragging & dropping, selecting or pasting them.

☒ Allow edits by maintainers ?

Create pull request

실습(5) –Pull Request

1. Clone한 레퍼지토리에 로컬에서 파일 만들기
2. 브랜치 새로 만들고 이동하기
3. add & commit하기
4. 새로만든 브랜치로 push하기
5. 깃허브 들어가서 새로운 브랜치로 이동한후 Pull Request보내기

03. Visual Studio Code GIT협업

1. VScode란

2. 기초셋팅

3. 버전관리

4. Vscode GIT협업하기

6. 원본저장소 주인은 PR받고 merge여부 결정하기

팀장역할

Search or jump to...

Pull requests Issues Codespaces Marketplace Explore

HamToriS / Boss Public

Pin Unwatch 1 Fork 1 Star 0

Code Issues **Pull requests 1** Actions Projects Wiki Security Insights Settings

Label issues and pull requests for new contributors

Now, GitHub will help potential first-time contributors [discover issues](#) labeled with [good first issue](#)

Dismiss

Filters is:pr is:open

Labels 9 Milestones 0 New pull request

1 Open 0 Closed

Author Label Projects Milestones Reviews Assignee Sort

연습 PR

#1 opened 1 minute ago by SILO-KSH

ProTip! Ears burning? Get @HamToriS mentions with [mentions:HamToriS](#).

03. Visual Studio Code GIT협업

1. VScode란

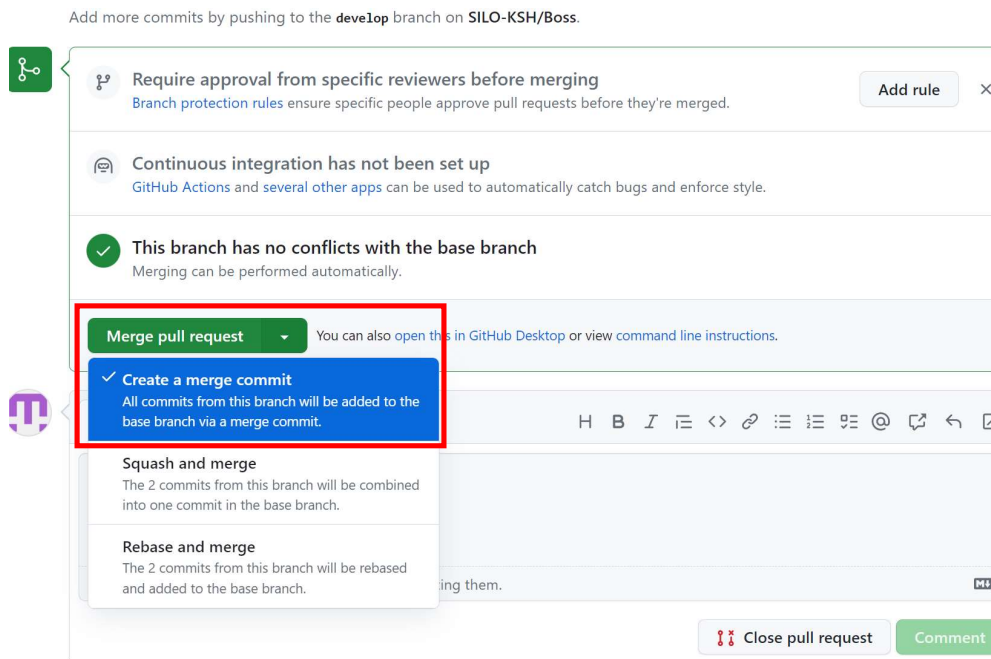
2. 기초셋팅

3. 버전관리

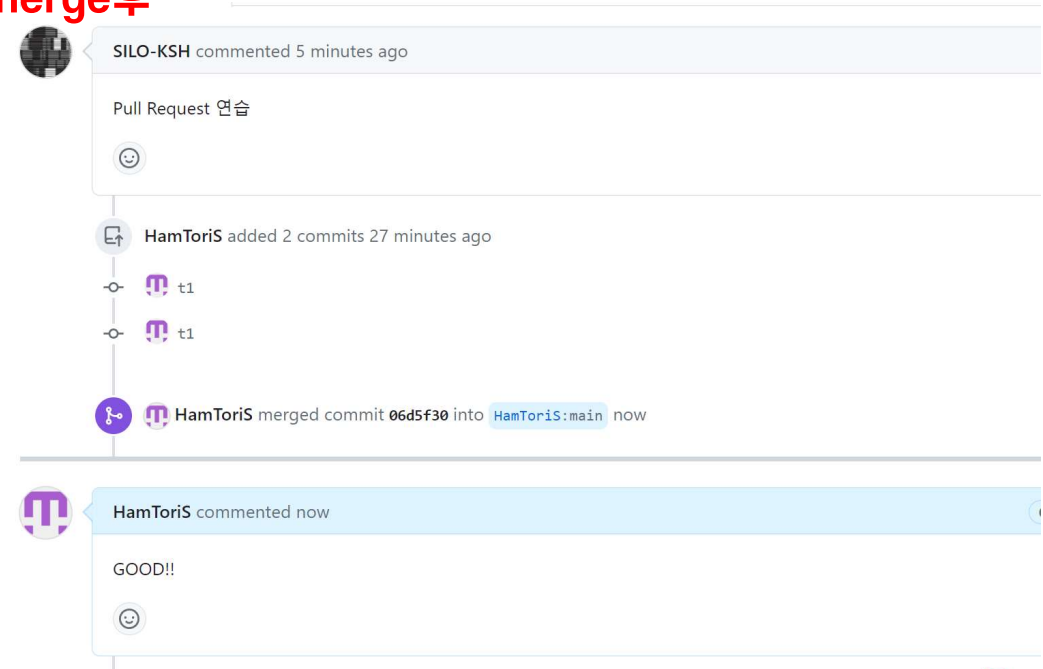
4. Vscode GIT협업하기

6. 원본저장소 주인은 PR받고 merge여부 결정하기

팀장역할



merge후



03. Visual Studio Code GIT협업

1. VScode란

2. 기초셋팅

3. 버전관리

4. Vscode GIT협업하기

6. 원본저장소 레퍼지토리 확인

팀장역할

HamToriS / Boss Public 원본저장소

<> Code Issues Pull requests Actions Projects Wiki Security Insights

main 1 branch 0 tags Go to file Add file

HamToriS Merge pull request #1 from SILO-KSH/develop 06d5f30 3 minutes ago

README.md	Initial commit
Test1.txt	t1

파일 들어왔다. merge완료!

README.md

Boss

03. Visual Studio Code GIT협업

1. VScode란

2. 기초셋팅

3. 버전관리

4. Vscode GIT협업하기

7. Merge 이후 동기화 및 branch삭제

master로 이동
git checkout main

upstream 동기화
git pull upstream main

origin에 반영
git push origin main

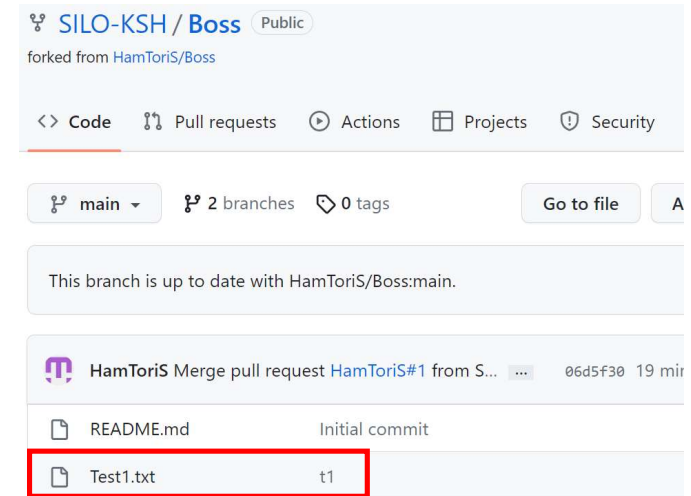
```
C:\Users\silolab_ksh\Desktop\Clone\Boss>git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

C:\Users\silolab_ksh\Desktop\Clone\Boss>git branch
develop
* main

C:\Users\silolab_ksh\Desktop\Clone\Boss>git pull upstream main
remote: Enumerating objects: 1, done.
remote: Counting objects: 100% (1/1), done.
remote: Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (1/1), 641 bytes | 320.00 KiB/s, done.
From https://github.com/HamToriS/Boss
* branch          main      -> FETCH_HEAD
* [new branch]     main      -> upstream/main
Updating 4827874..06d5f30
Fast-forward
 Test1.txt | 3 +++
1 file changed, 3 insertions(+)
create mode 100644 Test1.txt

C:\Users\silolab_ksh\Desktop\Clone\Boss>git push origin main
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 661 bytes | 661.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/SILO-KSH/Boss.git
4827874..06d5f30  main -> main
```

다시 main branch로 이동함.



fork한 저장소에도 파일동기화

03. Visual Studio Code GIT협업

1. VScode란

2. 기초셋팅

3. 버전관리

4. Vscode GIT협업하기

7. Merge 이후 동기화 및 branch삭제

로컬 브랜치 삭제

git branch -d develop

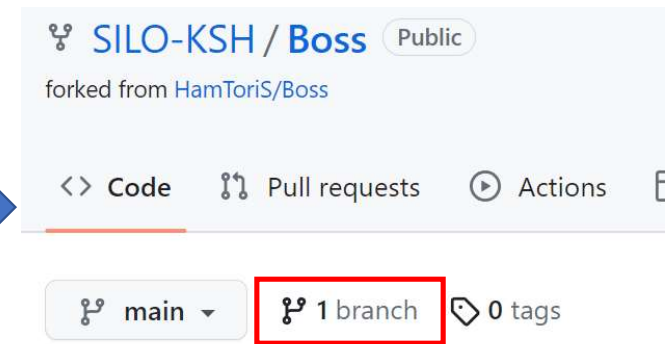
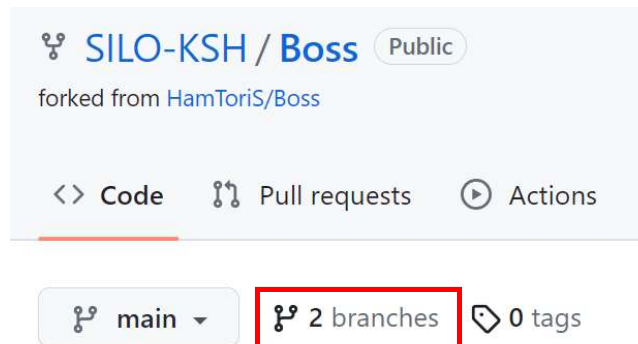
원격 브랜치 저장소도 삭제하고 싶은 경우

git push origin --delete develop

```
C:\Users\silolab_ksh\Desktop\Clone\Boss>git branch -d develop
Deleted branch develop (was ed65a8d).

C:\Users\silolab_ksh\Desktop\Clone\Boss>git branch
* main

C:\Users\silolab_ksh\Desktop\Clone\Boss>git push origin --delete develop
To https://github.com/SILO-KSH/Boss.git
- [deleted]      develop
```



원격저장소 브랜치 까지 삭제 완료

작업 수행마다 다시 브랜치를 만들어서 작업하고 반복하면 된다.

<순서정리>

1. 원본저장소를 fork한다.
2. 내 컴퓨터로 clone한다.
3. 나의 브랜치를 새로 만든다.
4. 작업한 파일들을 내 브랜치에 add, commit, push한다.
5. 깃허브에서 fork한 내 레퍼지토리에 들어간다.
6. 내가 올린 브랜치로 이동한다.
7. pull request요청을 한다.

팀원

8. 원본저장소(팀장)은 pull request요청을 보고 승낙한다.

9. merge한다.

팀장

10. vscode에서 원본저장소의 merge된 내용을 pull한다.

11. 내 원격저장소로 다시 push한다.

12. 작업이 끝난 브랜치들은 삭제한다.

팀원



반복

실습(6) – 내 원격저장소로 push

1. 생성했던 브랜치 삭제
2. 원본저장소에서 pull받기
3. 내 원격저장소로 push하기

03. Visual Studio Code GIT협업

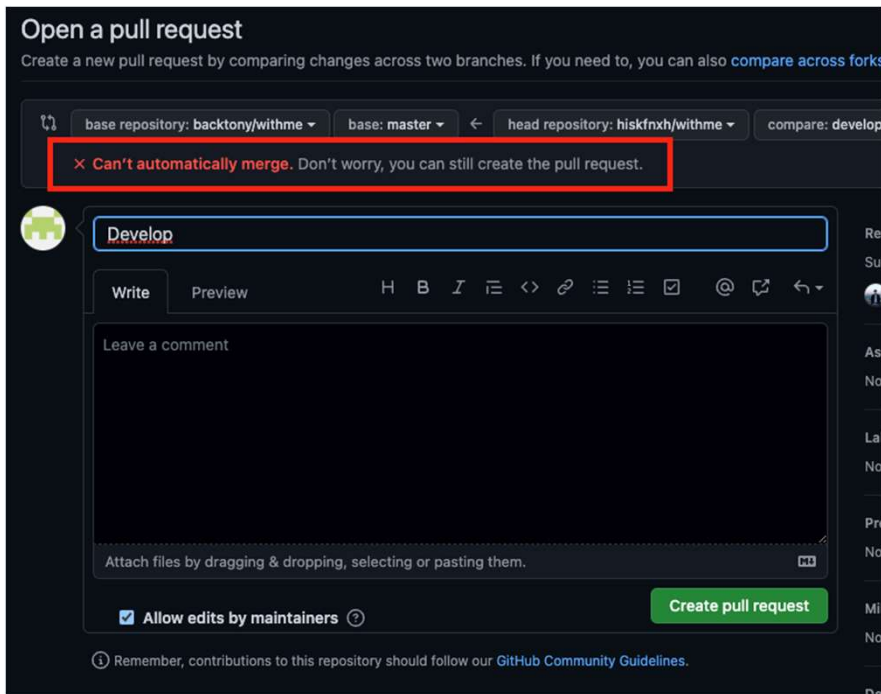
1. VScode란

2. 기초셋팅

3. 버전관리

4. Vscode GIT협업하기

8. Pull Request merge **conflict** 충돌발생경우



upstream 에서 코드를 가져와 작업하고 origin에
푸시하여 pull request를 날리려고 보면 아래와 같
은 문구가 나올때가 있다.

내가 로컬에서 작업하는 동안 누군가 upstream에
commit 했기 때문에 지금 pull request를 진행할
경우 **merge conflict**가 발생한다는 의미.