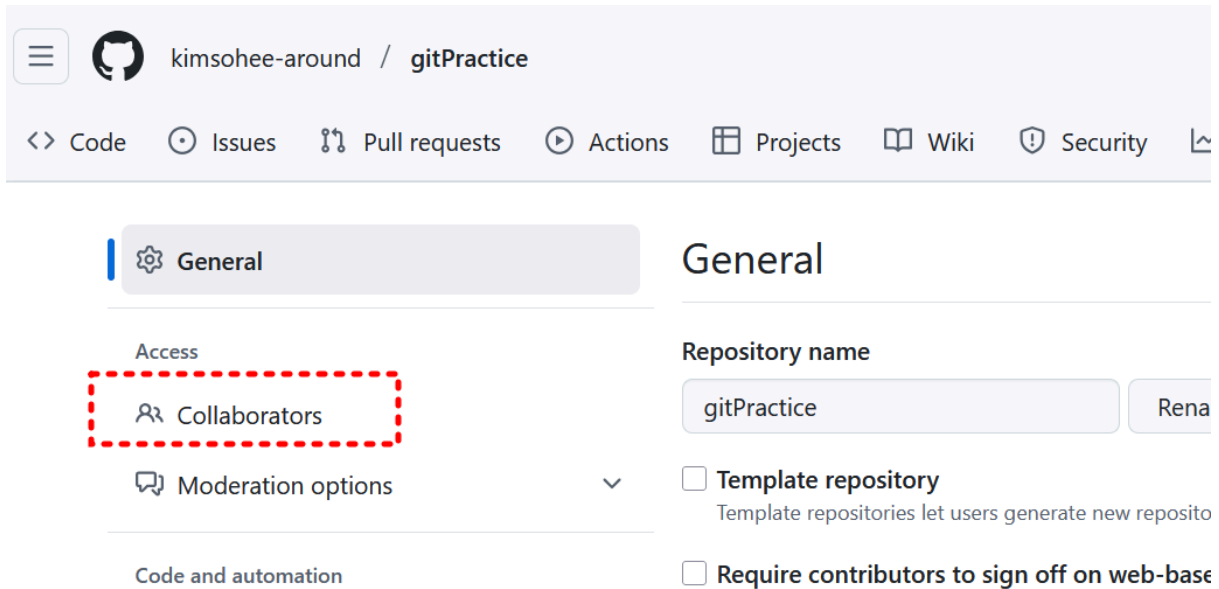


팀원 초대

- 팀 저장소 : Settings



GitHub repository settings for **gitPractice** by **kimsoshee-around**.

General

Access

- Collaborators** (highlighted with a red dashed box)
- Moderation options
- Code and automation

Repository name

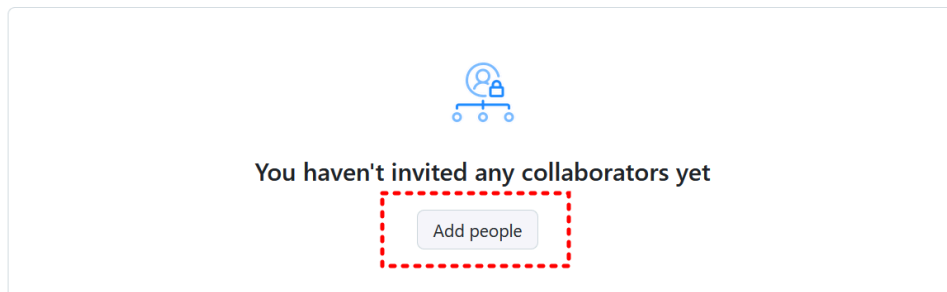
gitPractice

☐ **Template repository**
Template repositories let users generate new repositories from this template.

☐ **Require contributors to sign off on web-based commits**

- 팀원 추가

Manage access



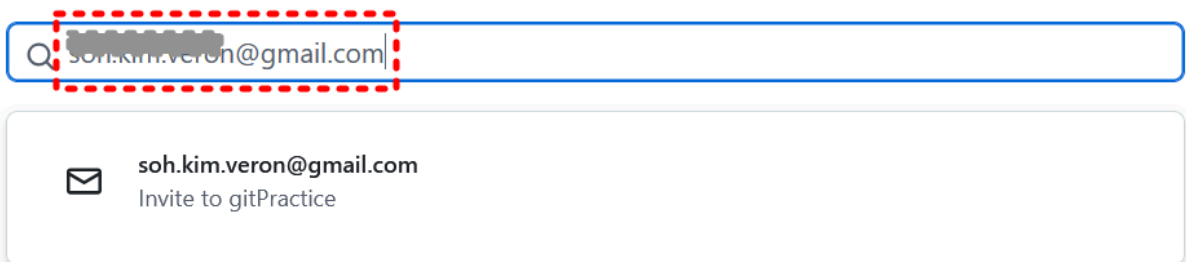
You haven't invited any collaborators yet

Add people (highlighted with a red dashed box)

Add people to gitPractice

×

Search by username, full name, or email



soh.kim.veron@gmail.com (highlighted with a red dashed box)

soh.kim.veron@gmail.com
Invite to gitPractice

Cancel

Add to repository

Add people to gitPractice



Cancel

Add sch.kim.veron@gmail.com

- 팀원 수신 이메일



@kimssohee-around has invited you to collaborate
on the
[kimssohee-around/gitPractice](#) repository

You can [accept](#) or [decline](#) this invitation. You can also head over to <https://github.com/kimssohee-around/gitPractice> to check out the repository or visit [@kimssohee-around](#) to learn a bit more about them.

This invitation will expire in 7 days.

[View invitation](#)

- 팀원 수락



[kimssohee-around](#) invited you to collaborate on
[kimssohee-around/gitPractice](#)

[Accept invitation](#)

[Decline invitation](#)

 Owners of gitPractice will be able to see:

- Your public profile information
- [Certain activity](#) within this repository
- Country of request origin
- Your access level for this repository
- Your IP address

Is this user sending spam or malicious content?
[Block kimssohee-around](#)

- 팀 저장소 : 팀원 등록 결과


Manage access


Add people

☐ Select all

Type ▾

☐

 **sohi-kim**
Collaborator



원격 저장소 보호

팀 저장소 Settings

⚙️ General

Access

👤 Collaborators

🔒 Moderation options ▾

Code and automation

🌿 Branches

Branch protection rules

🔗

Classic branch protections have not been configured
Define branch rules to disable force pushing, prevent branches from being deleted, or require pull request reviews.
[Learn more about repository rules](#) and [protected branches](#).

Add branch ruleset

Add classic branch protection rule

- ruleset 추가

Ruleset Name *

main

Enforcement status

 Active ▾

Target branches

Which branches should be matched?

Branch targeting criteria

Add target ▾

+

 Include default branch

+

 Include all branches

Branch targeting has not been configured

설정 추가

Rules

Which rules should be applied?

Branch rules

다른 기존 체크 항목은 그대로 유지함.
아래 사항만 체크하고 값 설정

- ☒ **Require a pull request before merging**
Require all commits be made to a non-target branch and submitted via a pull request

Hide additional settings ^

Required approvals

1 ▾

The number of approving reviews that are required before a pull request can be merged

→ 맨 아래에 **create** 버튼 클릭하여 완료.!

협업: 작업 → PR

1. 팀원 로컬 저장소로 clone

```
git clone https://github.com/kimsohee-around/gitPractice.git
```

2. 원격 저장소 확인

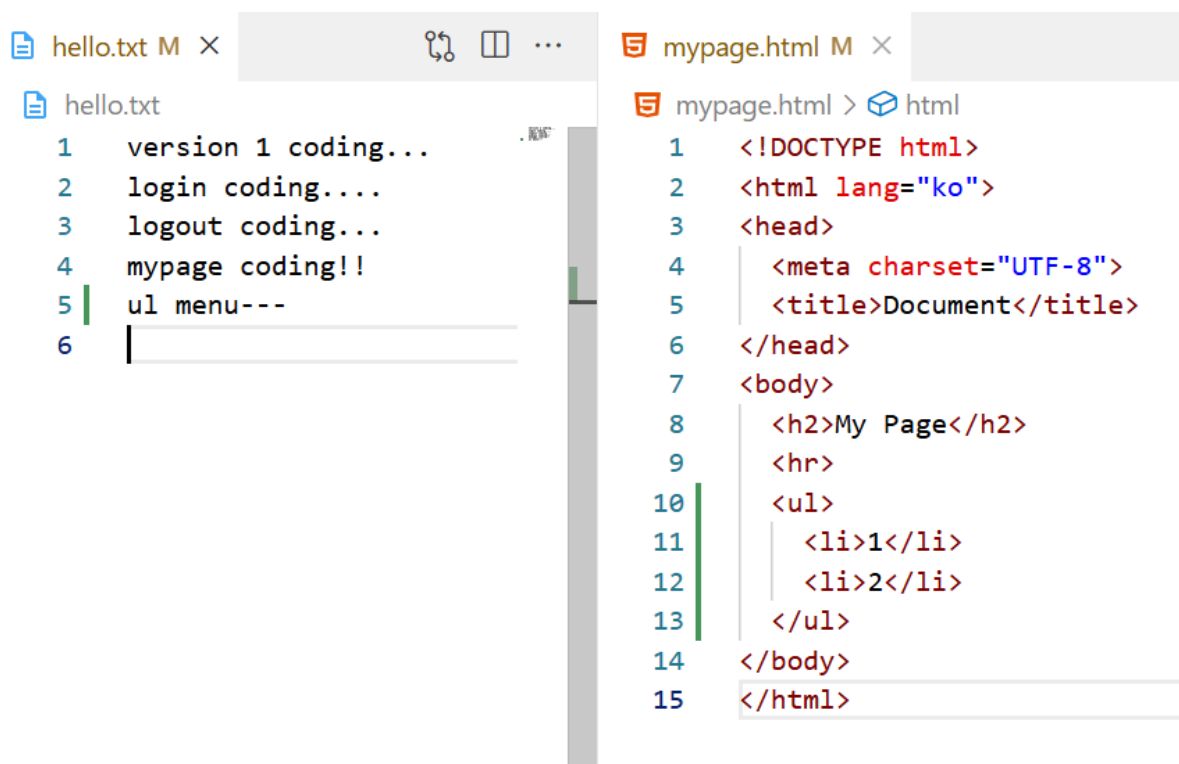
```
cd gitPractice
```

```
git remote -v
```

3. 개인 저장소에서 브랜치 생성 및 작업 → 예: **feature/mypage** 같은 브랜치를 만들어 코드 작업 수행.

```
git switch -c feature/mypage
```

- 코드 작업 예시



- 스테이징 후 커밋

```
git add .
```

```
git commit -m "mypage menu"
```

4. main 브랜치 변경 확인 후 변경 내용 가져오기

```
git fetch origin main
```

```
git pull origin main
```

6. 작업 완료 후 로컬 저장소의 feature/mypage 브랜치에서 main 브랜치 병합

(main 의 변경 내용과 충돌 있는지 먼저 확인)

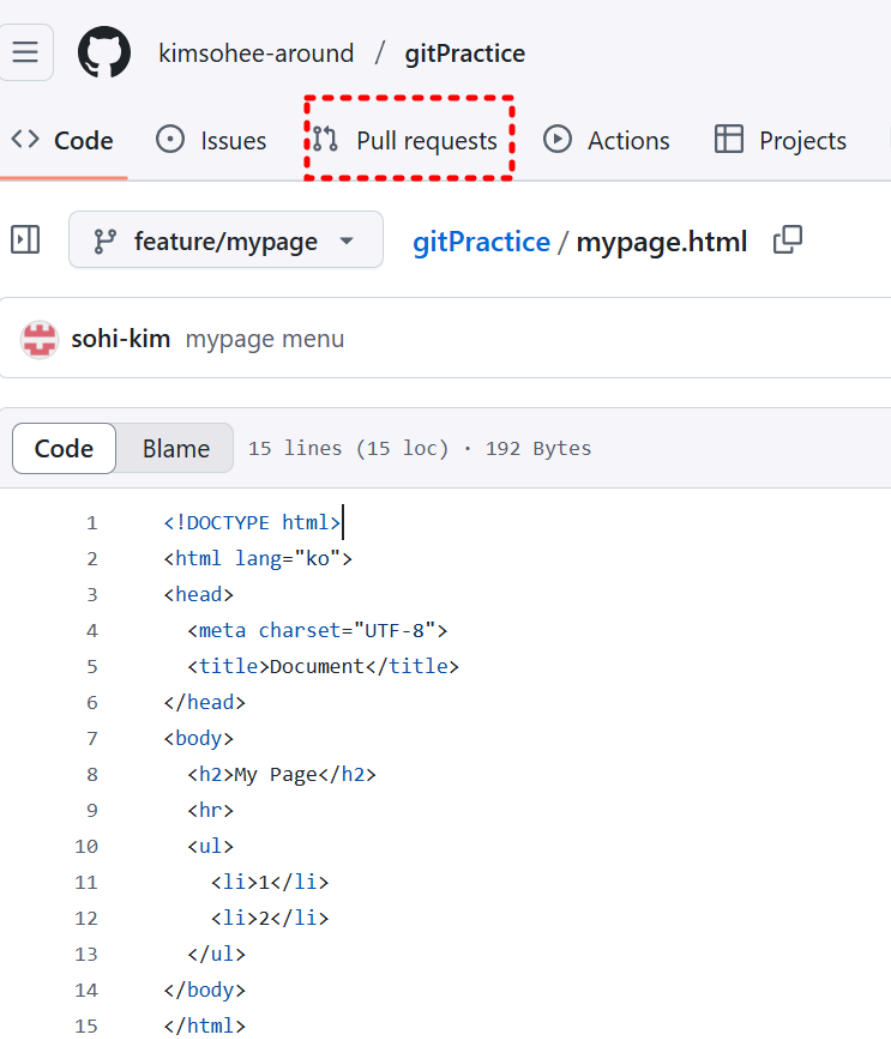
```
git merge main
```

충돌이 있다면 feature/mypage 브랜치의 소스파일을 수정하기

```
git push origin feature/mypage
```

```
myste@DESKTOP-CER1G3T MINGW64 /c/class250616/gitPractice (main)
$ git push origin feature/mypage
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 394 bytes | 394.00 KiB/s, done.
Total 4 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
remote:
remote: Create a pull request for 'feature/mypage' on GitHub by visiting:
remote:   https://github.com/kimsohee-around/gitPractice/pull/new/feature/mypage
remote:
To https://github.com/kimsohee-around/gitPractice.git
 * [new branch]      feature/mypage -> feature/mypage
```

8. 팀 저장소에서 **Pull Request(PR)** 생성 → GitHub에서 “Compare & pull request” 메시지가 뜨면 사용하거나, 수동으로 팀 저장소에 **PR** 요청.



GitHub repository page for **kimsoshee-around / gitPractice**. The **Pull requests** tab is highlighted with a red dashed box. Below the repository name, the file **feature/mypage** is selected, showing **gitPractice / mypage.html**. The code editor displays the following HTML code:

```
1 <!DOCTYPE html>|
2 <html lang="ko">
3 <head>
4   <meta charset="UTF-8">
5   <title>Document</title>
6 </head>
7 <body>
8   <h2>My Page</h2>
9   <hr>
10  <ul>
11    <li>1</li>
12    <li>2</li>
13  </ul>
14 </body>
15 </html>
```

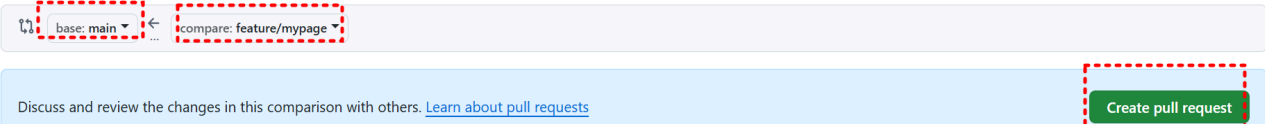
↓ 코드 작성자가 **PR** 생성하기

New pull request

↓ 작업 브랜치를 **main** 으로 병합 **PR**

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](#) or [learn more about diff comparisons](#).



base: main ← compare: feature/mypage

Discuss and review the changes in this comparison with others. [Learn about pull requests](#)

Create pull request

↓ PR 내용 작성

Open a pull request


Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#). [Learn more about diff con](#)

↕

base: main

←

compare: feature/mypage


 Add a title

Add a description

Write

Preview

H B I ≡ <> 🔗 ≡ ≡ ≡ 📎 @ ↶ ↷

 마이페이지 1차 작업 검토바랍니다.

Markdown is supported Paste, drop, or click to add files

Create pull request

↓ 병합은 할 수 없음. 여기까지 끝.


🔗 Open [kimsoshee-around](#) wants to merge 1 commit into `main` from `feature/mypage` 📄


💬 Conversation 0


📄 Commits 1


🔍 Checks 0


📄 Files changed 2

 kimsoshee-around commented [now](#)





 `mypage_menu`



✖ Review required

At least 1 approving review is required by reviewers with write access.

⚠ Merging is blocked

At least 1 approving review is required by reviewers with write access.

Merge pull request

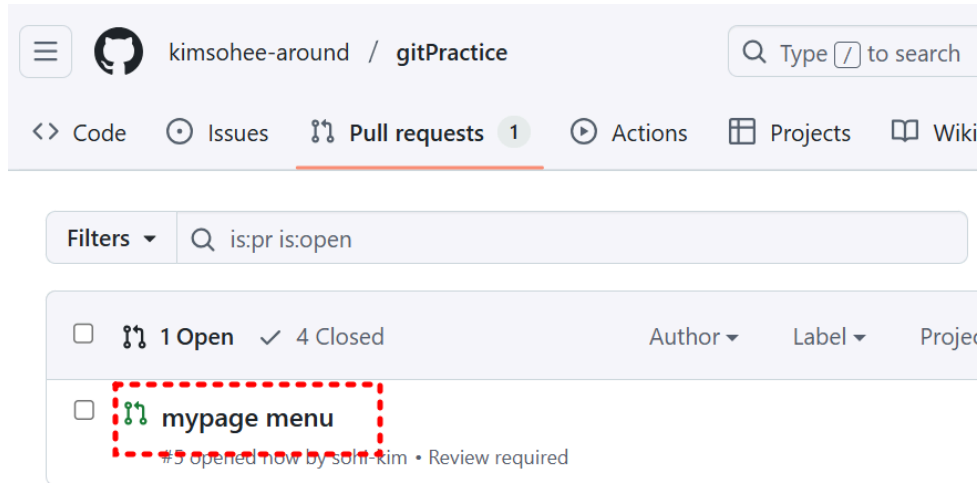
You can also merge this with the command line. [View command line instructions.](#)

브랜치 보호 정책에 따라 코드 작성자 이외에 1명이 소스코드를 검토해야 함.

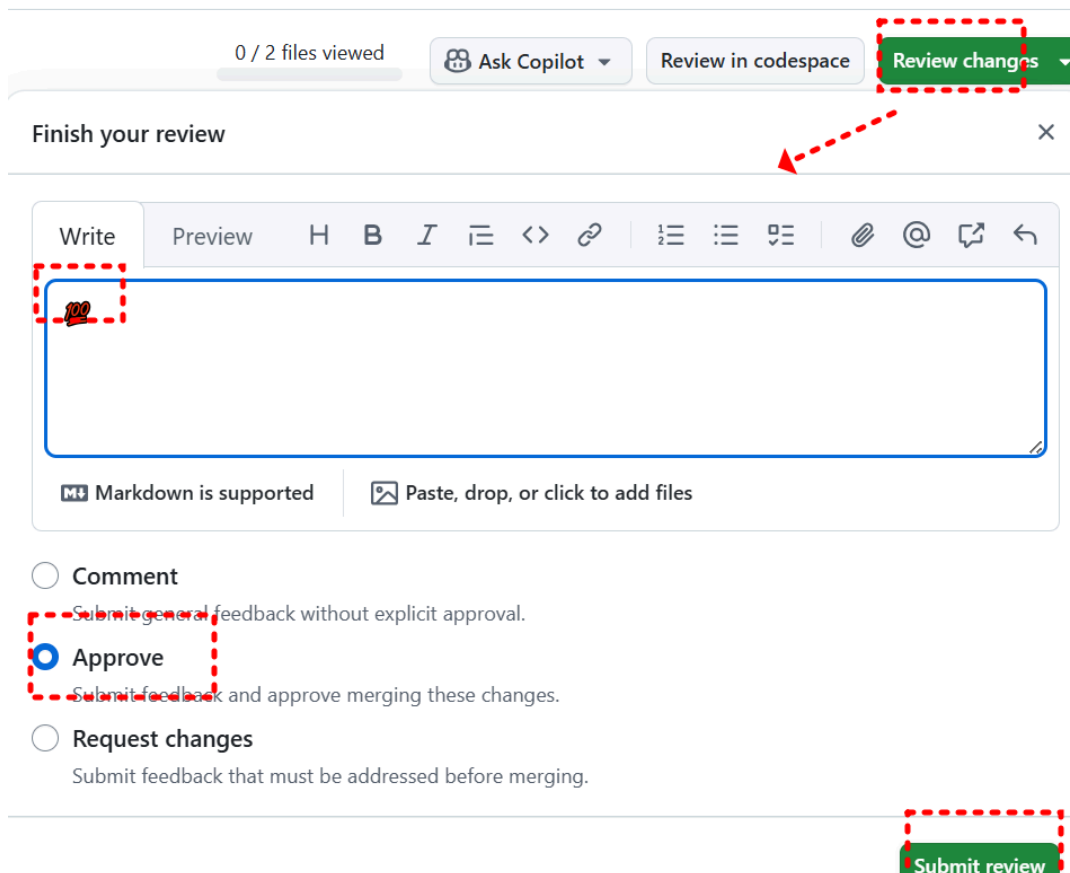
9. PR 검토 후 병합 : 팀 저장소

코드 작성자 및 **PR** 생성자 외의 다른 팀원이 코드 리뷰하기

- 검토 후 병합할 **PR** 선택하기



↓ 코드 검토 후 리뷰 작성하고 승인 제출하기.



↓ 병합하기

The screenshot shows a GitHub Pull Request interface. At the top, a user 'kimsoshee-around' has approved the changes. Below this, a comment box is visible. The main section of the interface shows a green checkmark icon and the text 'Changes approved' with a subtext '1 approving review by reviewers with write access.' Below this, it says '1 approval >'. Another green checkmark icon is followed by 'No conflicts with base branch' and 'Merging can be performed automatically.' At the bottom, there is a green button labeled 'Merge pull request' with a dropdown arrow, which is highlighted by a red dashed box. To the right of the button, it says 'You can also merge this with the command line. [View command line instructions.](#)'

PR은 제안, 반영은 머지

즉, PR은 "변경 사항을 검토해 주세요!"라는 제안서 🙋 반영 여부는 저장소 관리자에게 달려 있어요

협업자 작업 흐름

1. 작업 브랜치 생성 예: **feature/mypage**
→ **main** 브랜치에서 따로 분기해서 브랜치를 만듭니다.
2. 작업 & 커밋 & **push** 이 작업은 모두 본인이 만든 작업 브랜치에서 진행하기.
3. **PR** 생성 작업 브랜치 → **main** 브랜치로 **Pull Request**를 만들어서 팀원들과 리뷰 및 협업을 진행해요.
4. 다른 팀원이 리뷰 후 머지 승인되면 **main**으로 병합 (**Merge**) → 보통은 직접 머지 권한도 있지만, 일부 팀에서는 리뷰어가 머지하기도 해요.

병합 옵션 설명

옵션	설명	커밋 히스토리	추천 상황
 Allow merge commits	헤드 브랜치의 모든 커밋을 한 번에 병합하면서, 병합 커밋(Merge Commit)을 남김	병합 커밋 추가 (e.g. Merge pull request #15)	커밋 이력을 그대로 보존하고 싶은 경우
 Allow squash merging	헤드 브랜치의 모든 커밋을 하나의 커밋으로 압축해서 병합	단일 커밋으로 통합	커밋 로그를 간결하게 유지하고 싶은 경우
 Allow rebase merging	헤드 브랜치의 모든 커밋을 기존 base 브랜치 뒤에 하나씩 추가	커밋은 유지되나 Merge Commit 은 없음	깔끔한 선형 히스토리를 원할 때