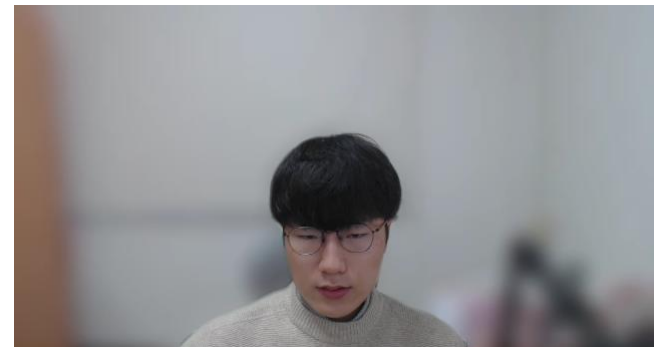


Open-Source Software Practice

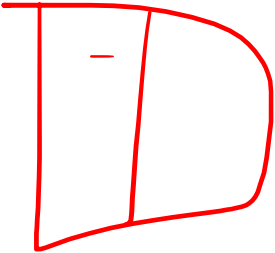
11. Collaboration

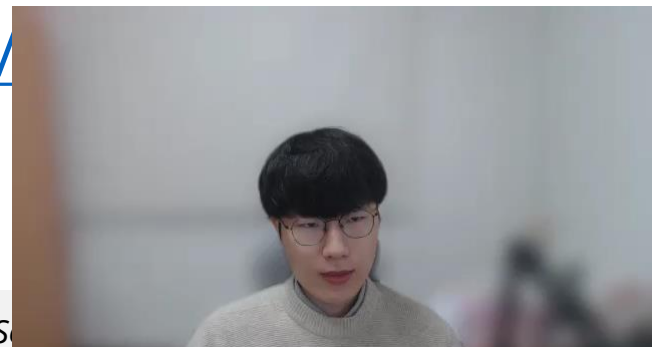
Instructor: Jaemin Jo (조재민, jmjo@skku.edu)
Interactive Data Computing Lab (*IDCLab*),
College of Computing and Informatics,
Sungkyunkwan University



Markdown

HTML

- **Markdown** is a lightweight markup language that you can use to format text documents.
 - Similar to HTML, but much simpler.
 - It is used everywhere in GitHub, e.g., documentation, issues, pull requests, etc.
 - (Heading, lists, tables, images, links, ...)
- 
- ✓ <https://guides.github.com/features/mastering-markdown/>
 - ✓ <https://www.markdownguide.org/basic-syntax>



Markdown

Markdown	HTML	Rendered Output
# Heading level 1	<h1>Heading level 1</h1>	Heading level 1
## Heading level 2	<h2>Heading level 2</h2>	Heading level 2
### Heading level 3	<h3>Heading level 3</h3>	Heading level 3
#### Heading level 4	<h4>Heading level 4</h4>	Heading level 4
##### Heading level 5	<h5>Heading level 5</h5>	Heading level 5
##### Heading level 6	<h6>Heading level 6</h6>	Heading level 6

Markdown	HTML	Rendered Output
I just love bold text .	I just love bold text.	I just love bold text .

Markdown	HTML	Rendered Output
Italicized text is the <i>cat's meow</i> .	Italicized text is the cat's meow.	Italicized text is the <i>cat's meow</i> .

Markdown	HTML	Rendered Output
- First item - Second item - Third item - Fourth item	First itemSecond itemThird itemFourth item	<ul style="list-style-type: none"> First item Second item Third item Fourth item

Markdown	HTML	Rendered Output
1. First item 2. Second item 3. Third item 4. Fourth item	First itemSecond itemThird itemFourth item	1. First item



Markdown

(u)
=

(=) (1)

My favorite search engine is [Duck Duck Go](https://duckduckgo.com).

The rendered output looks like this:

My favorite search engine is [Duck Duck Go](https://duckduckgo.com).

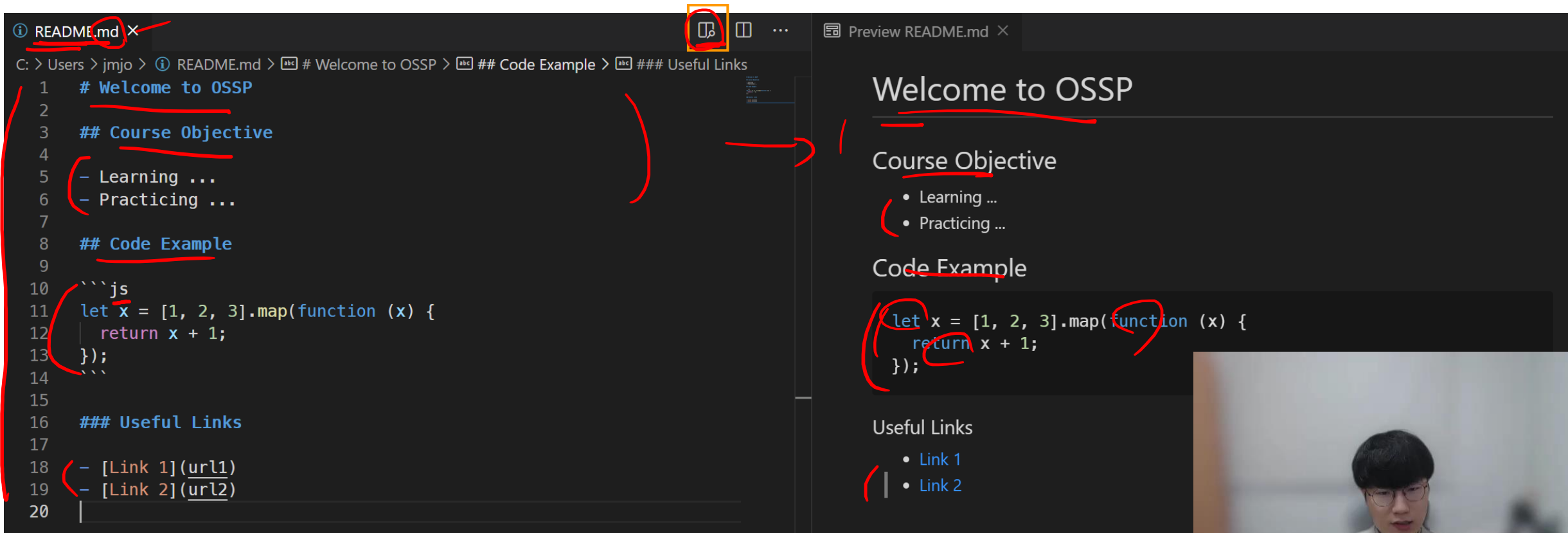
```
```ruby  
require 'redcarpet'
markdown = Redcarpet.new("Hello World!")
puts markdown.to_html
```
```

```
( ( require 'redcarpet'  
  markdown = Redcarpet.new("Hello World!")  
  puts markdown.to_html  
)
```



Markdown

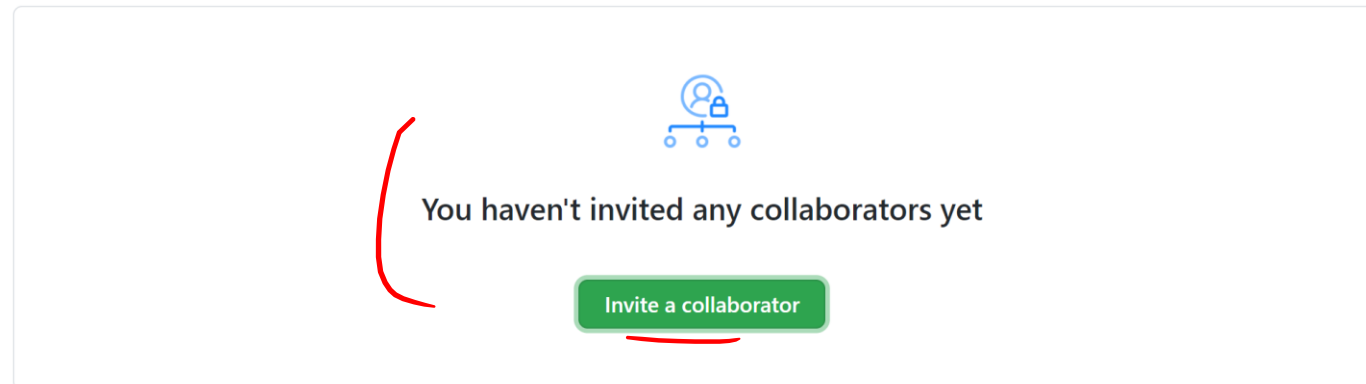
- VSCode supports Markdown!
- Create a .md file and open it on VSCode.



Owner vs Collaborators

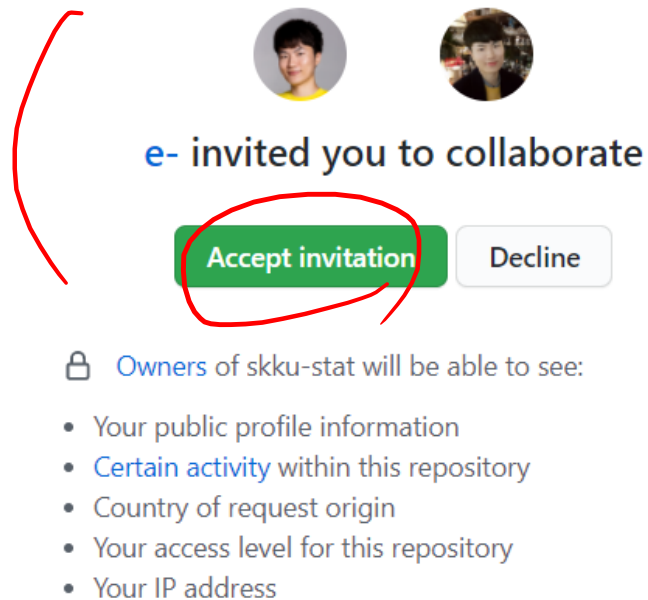
- **Owner**: the user who created a repository.
 - <https://github.com/e-/skku-stat>
- You can invite collaborators to your repo as you want.
 - Settings -> Manage Access

Manage access

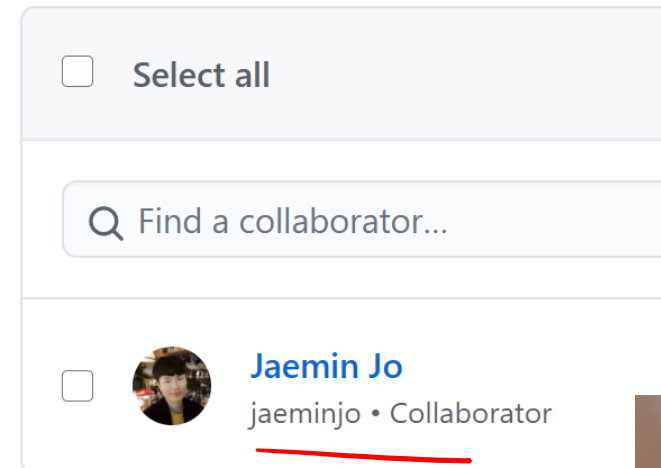


Owner vs Collaborators

- Send the invitation link after inviting a collaborator



Manage access



Owner vs Collaborators

- Things that both the owner and collaborators can do:
 - Pull the content (read)
 - Push changes (write)
 - Rename a branch
 - Create, assign, close, and re-open issues
 - Create, merge, and close pull requests
 - ... almost everything that you do to maintain the repository



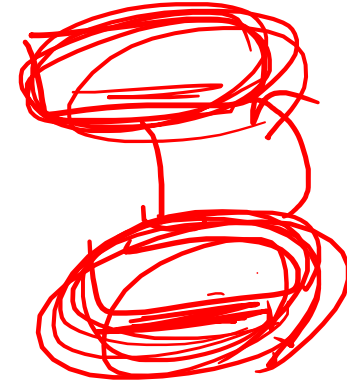
Owner vs Collaborators

- Things that the owner can do but collaborators cannot do:
 - Invite collaborators
 - Change the visibility of the repo (e.g., public -> private)
 - Delete the repo
 - Manage security settings
- In your project, one of your team member should be an owner and invite others to the repo.



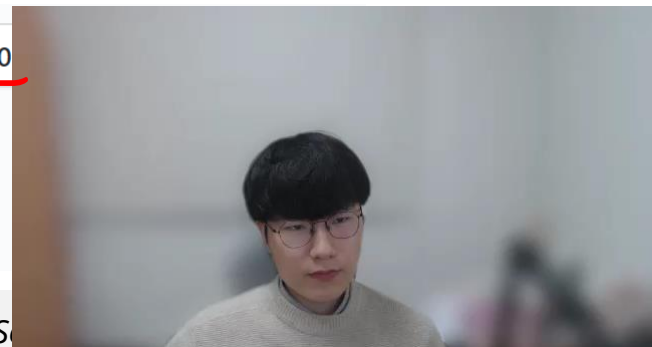
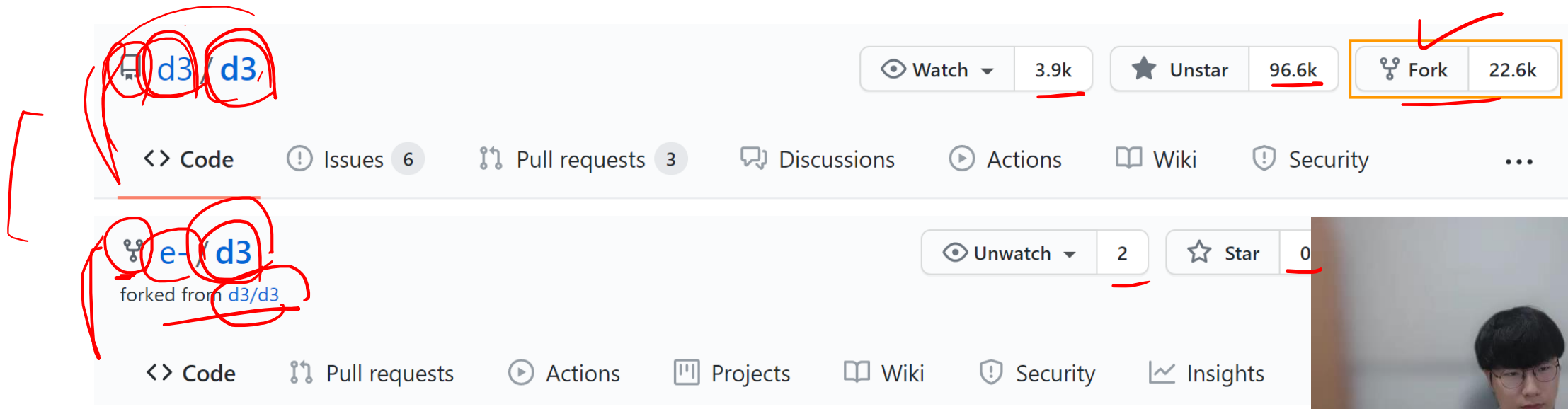
External Contributors

- But, how can I contribute to an open-source project if I am not invited as a collaborator?
- In this case, you are **an external contributor.**
- Fork the target repo to your account.
- Modify your repo.
- Write **a pull request.**
- The collaborators of the original repo will approve your pull request



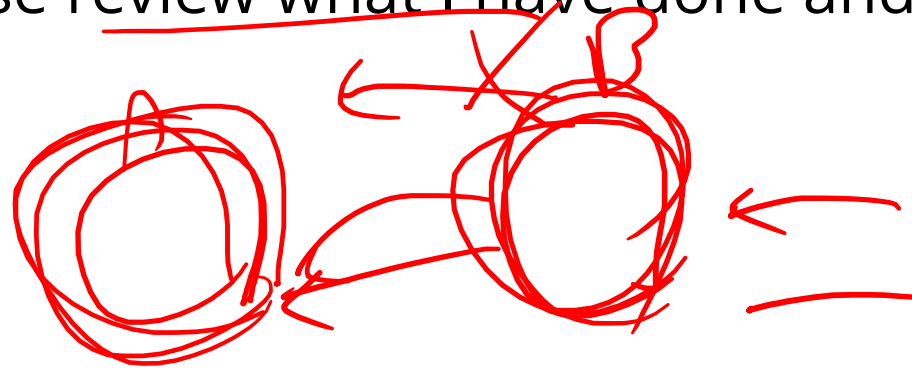
Forking a Repository

- If you fork a repo, you will get a clone of the original repo under your account.
 - e.g., *d3/d3* becomes *e-/d3*
- You are the owner of the repo, so you can do anything, e.g., committing changes.



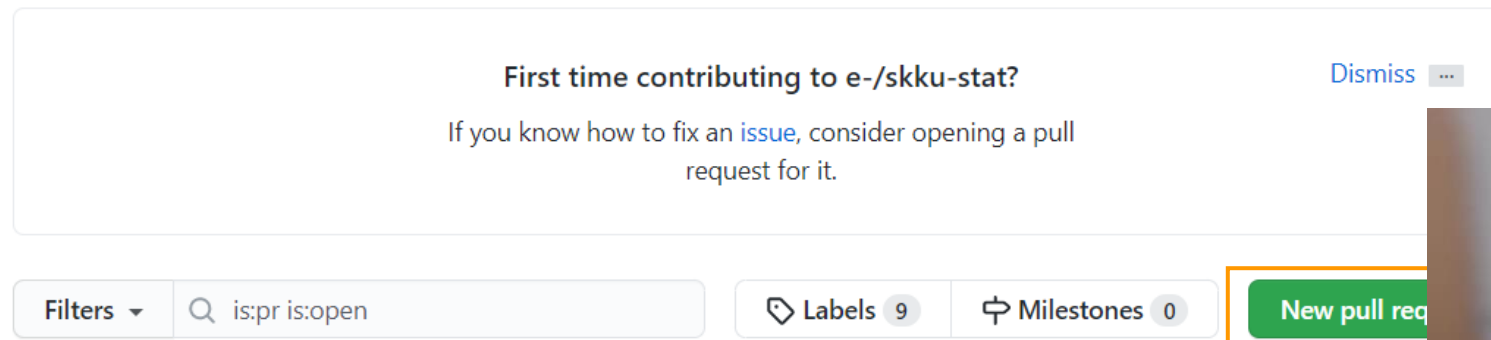
Creating a Pull Request

- However, changes you made (to e-/d3) are not visible in the original repo (d3/d3).
- To reflect the changes on the original repo, you need to create **a pull request** to d3/d3.
- **Pull request:** “please review what I have done and merge the changes if you like”.



Creating a Pull Request

- **User A** creates a repo A/skku-stat.
- **User B** forks the repo as B/skku-stat.
- **User B** clones B/skku-stat into B's local machine.
- **User B** makes changes and push them to B/skku-stat.
 - *git commit* and *git push*
- **User B** goes to the “Pull requests” tab of A/skku-stat.



Creating a Pull Request

- **User B** clicks on “New pull request”.
- **User B** clicks on “Compare across forks”.

A


Compare changes

Compare changes across branches, commits, tags, and more below. If you need to, you can also [compare across forks](#).

 ←

Compare changes

Compare changes across branches, commits, tags, and more below. If you need to, you can also [compare across forks](#).

 ←

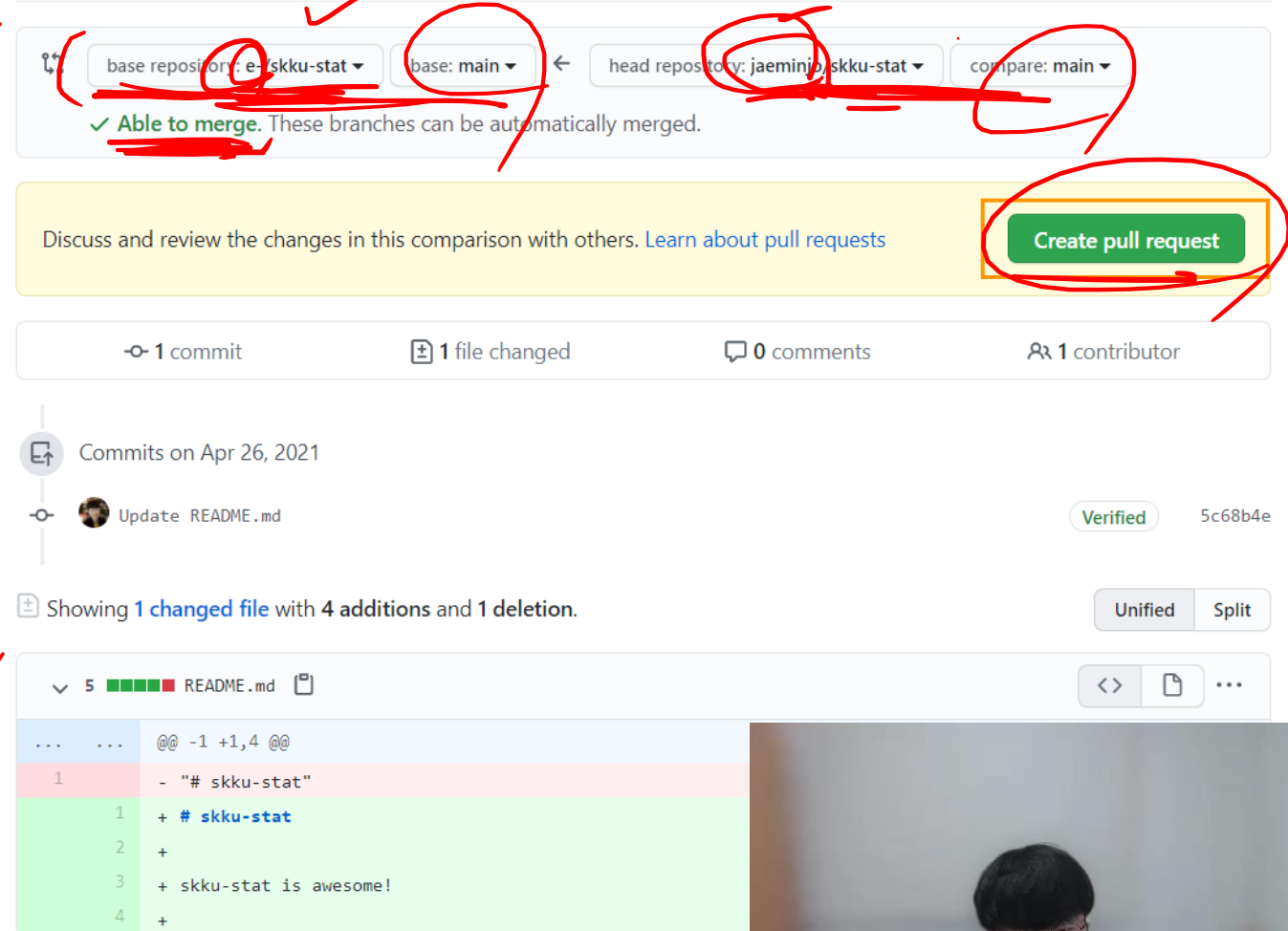


Creating a Pull Request

- Left: A's branch into which B wants to merge B's.
- Right: B's branch (newer)
- The diff view below shows the changes.
- **User B** clicks on "Create pull request".

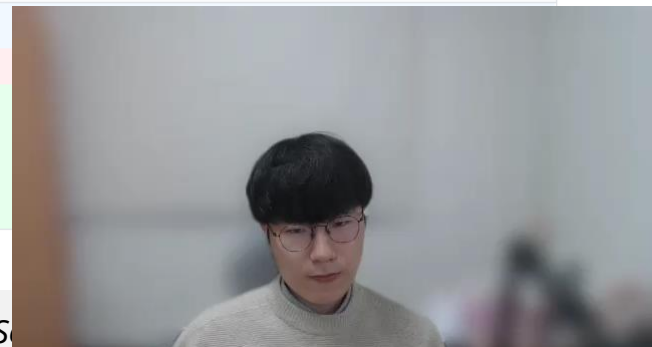
Comparing changes

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](#).



The screenshot shows the GitHub 'Comparing changes' interface. At the top, the 'base repository' is 'e-skku-stat' with 'base: main' selected. The 'head repository' is 'jaeminb-skku-stat' with 'compare: main' selected. A green checkmark indicates 'Able to merge. These branches can be automatically merged.' Below this is a yellow bar with the text 'Discuss and review the changes in this comparison with others. [Learn about pull requests](#)' and a green 'Create pull request' button. The interface shows '1 commit', '1 file changed', '0 comments', and '1 contributor'. Under 'Commits on Apr 26, 2021', there is a commit 'Update README.md' with a 'Verified' badge and hash '5c68b4e'. Below this, it says 'Showing 1 changed file with 4 additions and 1 deletion.' The diff view for 'README.md' shows a change from a single line to four lines. The diff content is as follows:

| Line | Diff |
|------|-------------------------|
| ... | @@ -1 +1,4 @@ |
| 1 | - "# skku-stat" |
| 1 | + # skku-stat |
| 2 | + |
| 3 | + skku-stat is awesome! |
| 4 | + |

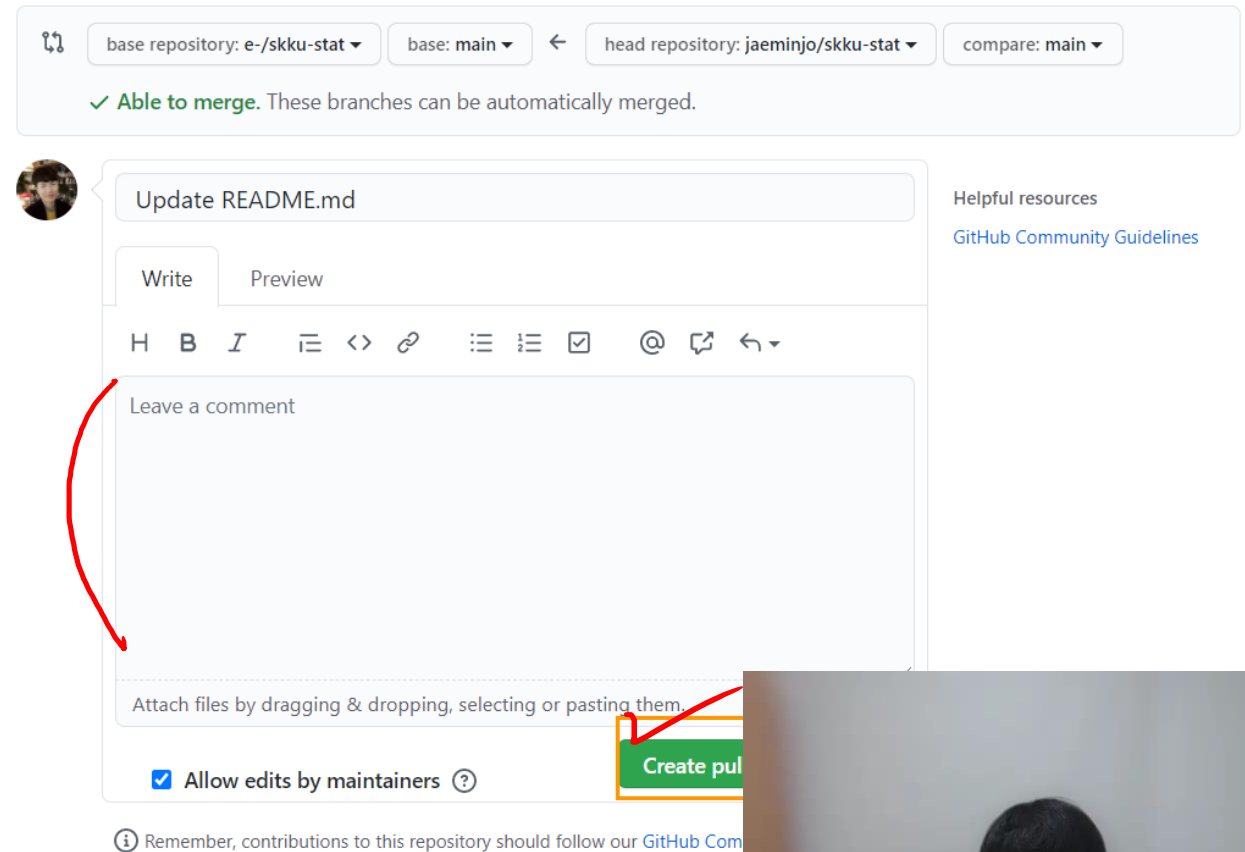


Creating a Pull Request

- Leave a message so that the owner and collaborators of the repo can understand what you did.
 - Use the markdown syntax!
- **User B** clicks on “Create pull request”.

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](#).



base repository: e-/skku-stat base: main head repository: jaeminjo/skku-stat compare: main

✓ Able to merge. These branches can be automatically merged.

Update README.md

Write Preview

Leave a comment

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

☒ Allow edits by maintainers ?

Create pull request

Remember, contributions to this repository should follow our [GitHub Community Guidelines](#)



Pull Request

- GitHub automatically runs the tests the repo has and checks merge conflicts.
- You can discuss with the maintainers (owner or collaborators).
 - Maybe they require some changes.

The screenshot shows a GitHub Pull Request titled "Update README.md #1" by user "jaeminjo". The PR is targeting the "e::main" branch from the "jaeminjo:main" branch. The interface includes a header with "Open" and "Edit" buttons, and a summary bar showing "Conversation 0", "Commits 1", "Checks 4", and "Files changed 1". A comment from "jaeminjo" is visible, stating "No description provided." Below the comment, a commit titled "Update README.md" is shown with a "Verified" status. A green box highlights two status messages: "All checks have passed" and "This branch has no conflicts with the base branch". A red arrow points from the text "Maybe they require some changes." to the "Write" tab of the comment section, which includes a rich text editor and a "Close pull request" button. A small inset image of a person is visible in the bottom right corner.

Update README.md #1

jaeminjo wants to merge 1 commit into e::main from jaeminjo:main

Conversation 0 Commits 1 Checks 4 Files changed 1

jaeminjo commented 1 minute ago

No description provided.

Update README.md

Verified 5c68b4e

Add more commits by pushing to the main branch on jaeminjo/skku-stat.

All checks have passed
4 successful checks

This branch has no conflicts with the base branch
Only those with write access to this repository can merge pull requests.

Write Preview

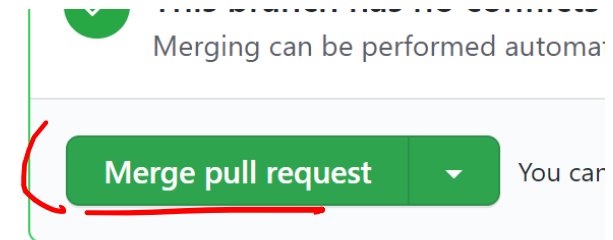
Leave a comment

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

Close pull request

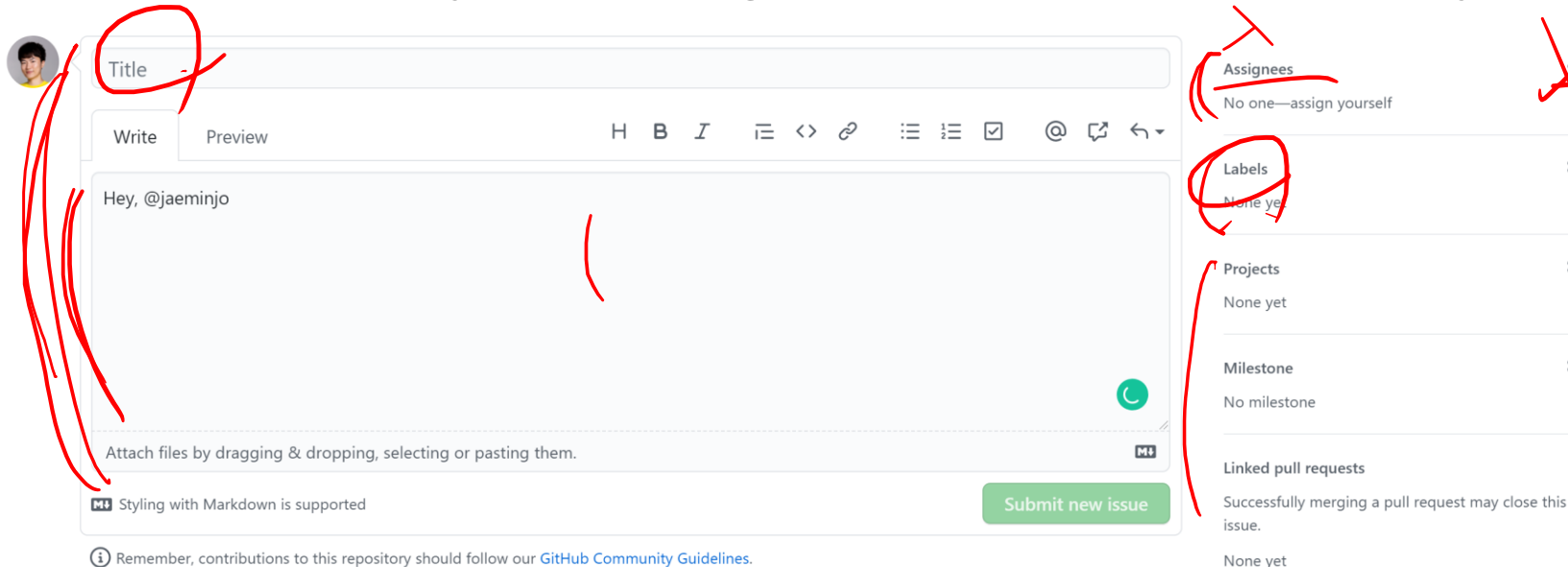
Creating a Pull Request

- **User A** finds the pull request that **User B** left and comments on it.
- **User B** modifies the source code again.
- The additional changes that **User B** made is automatically reflected on the old pull request.
 - No need to remove the old one and create a new one.
- **User A** clicks on “Merge pull request”.
- B's code has been applied to A's repo.



Issue Management

- Go to "Issues" tab
- Use Markdown
- You can notify a user using @(username).
- Assign to a specific user (maintainers only).
- Labels are very useful. Categorize the issues (maintainers only).



Title

Write Preview

Hey, @jaeminjo

Assignees
No one—assign yourself

Labels
None yet

Projects
None yet

Milestone
No milestone

Linked pull requests
Successfully merging a pull request may close this issue.
None yet

Submit new issue

| 9 labels | |
|------------------|--------------------------------------------|
| bug | Something isn't working |
| documentation | Improvements or additions to documentation |
| duplicate | This issue or pull request already exists |
| enhancement | New feature or request |
| good first issue | Good for newcomers |
| help wanted | Extra attention is needed |
| invalid | This doesn't seem right |
| question | Further information is requested |
| wontfix | This will not be worked on |

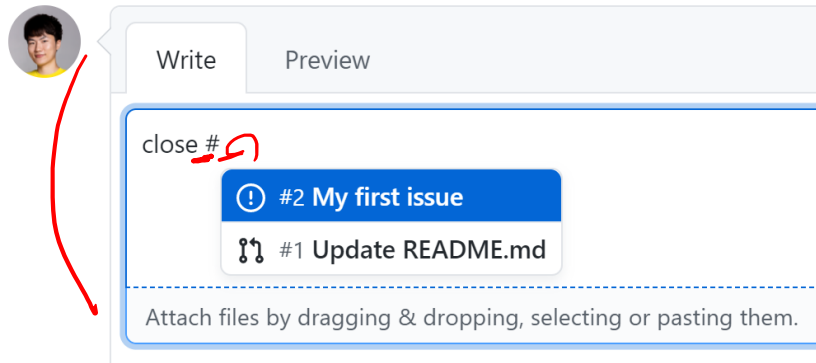


| <div> <div>30 Open</div> <div>70 Closed</div> </div> | <div> <div>Author</div> <div>Label</div> <div>Projects</div> <div>Milestones</div> <div>Assignee</div> <div>Sort</div> </div> |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------|
| <div> <div>Frontend stucks at loading</div> <div>question</div> <div>#125 opened 3 hours ago by huzeyfecoskun</div> </div> | |
| <div> <div>Add ccxt proxy configuration</div> <div>enhancement</div> <div>#124 opened 3 hours ago by 556isback</div> </div> | |
| <div> <div>How to get a Binance API test key?</div> <div>question</div> <div>#122 opened 10 hours ago by baptisteArno</div> </div> | <div> <div>2</div> </div> |
| <div> <div>Filter symbols in the Frontend</div> <div>enhancement</div> <div>#120 opened 3 days ago by chrisleekr</div> </div> | |
| <div> <div>Set last buy prices for same quote assets</div> <div>enhancement</div> <div>wontfix</div> <div>#119 opened 3 days ago by chrisleekr</div> </div> | <div> <div>1</div> </div> |
| <div> <div>Calculation of % of available balance asset</div> <div>question</div> <div>#118 opened 3 days ago by gilmillasseau</div> </div> | <div> <div>20</div> </div> |
| <div> <div>Enhancement: graphing on UI</div> <div>enhancement</div> <div>wontfix</div> <div>#117 opened 3 days ago by horstefan</div> </div> | |
| <div> <div>docker-stack.yml is not compatible with Portainer running on aarch64</div> <div>enhancement</div> <div>#114 opened 4 days ago by Licrefin</div> </div> | |



Issue Management

- Issues and pull requests are closely related.
 - User C reports a bug as an issue.
 - User B fixes the issue and creates a pull request.
 - User A approves the pull request.
- You can re-open or close a relevant issue in a pull request message.



Code Review

- To ensure the code quality, you can enforce people to do code review before merging a pull request.
- Settings -> Branches -> Branch protection rules

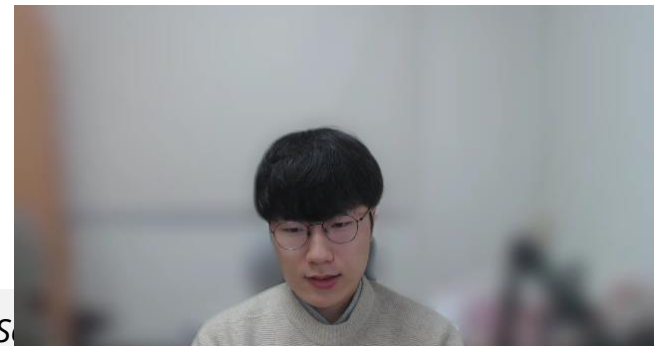
Branch protection rules



You haven't protected any of your branches

Define a protected branch rule to disable force pushing, prevent branches from being deleted, and optionally require status checks before merging. [Learn more about protected branches](#)

Add branch protection rule



Code Review

- Make at least one reviewer review the pull request before merging it.

- Set the branch name pattern to

"main".

der

Branch name pattern *

main

Protect matching branches

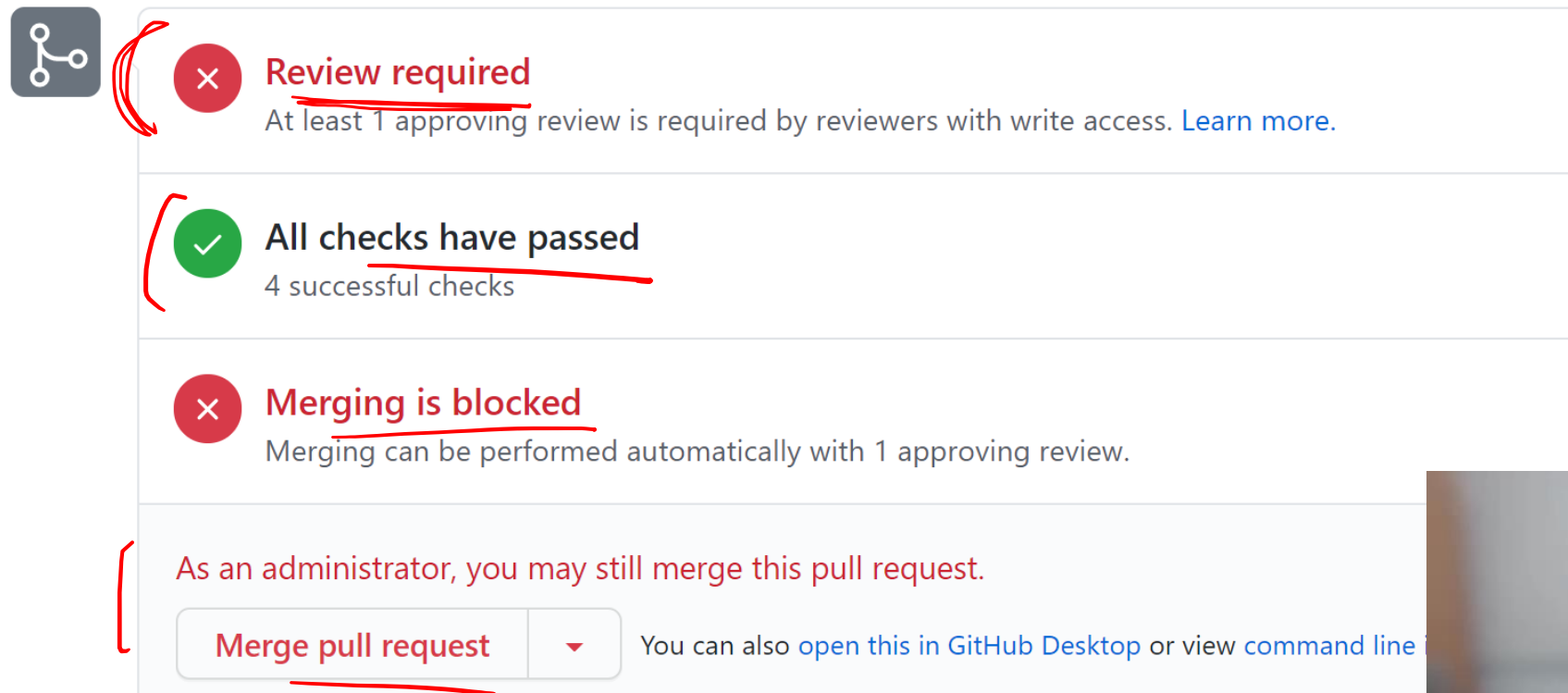
- ☒ Require a pull request before merging
When enabled, all commits must be made to a non-protected branch and submitted via a pull request before they can be merged into a branch that matches this rule.
- ☒ Require approvals
When enabled, pull requests targeting a matching branch require a number of approvals and no changes requested before they can be merged.
Required number of approvals before merging: 1
- ☐ Dismiss stale pull request approvals when new commits are pushed
New reviewable commits pushed to a matching branch will dismiss pull request review approvals.
- ☒ Require review from Code Owners
Require an approved review in pull requests including files with a designated code owner.
- ☒ Require approval from someone other than the last pusher
Require review approval from someone other than the last pusher to the pull request branch.
- ☒ Require status checks to pass before merging
Choose which [status checks](#) must pass before branches can be merged into a branch that matches this rule. When enabled, commits must first be pushed to another branch, then merged or pushed directly to a branch that matches this rule after status checks have passed.

- ☐ Require conversation resolution before merging
When enabled, all conversations on code must be resolved before merging. [Learn more.](#)
- ☐ Require signed commits
Commits pushed to matching branches must have verified signatures.
- ☐ Require linear history
Prevent merge commits from being pushed to matching branches.





Code Review


- Before a reviewer approves a pull request, you cannot merge the pull request.



The screenshot shows the GitHub pull request status interface. It features a dark gray icon with a white branching diagram on the left. The main content area is divided into three horizontal sections. The first section has a red circle with a white 'x' and the text 'Review required' in red, with a red underline. Below it, in smaller gray text, it says 'At least 1 approving review is required by reviewers with write access. [Learn more.](#)'. The second section has a green circle with a white checkmark and the text 'All checks have passed' in green, with a red underline. Below it, in smaller gray text, it says '4 successful checks'. The third section has a red circle with a white 'x' and the text 'Merging is blocked' in red, with a red underline. Below it, in smaller gray text, it says 'Merging can be performed automatically with 1 approving review.' At the bottom, there is a red bracket on the left side of a light gray box containing a button labeled 'Merge pull request' in red, with a red underline. To the right of the button is a dropdown arrow and the text 'You can also [open this in GitHub Desktop](#) or view [command line](#)'.

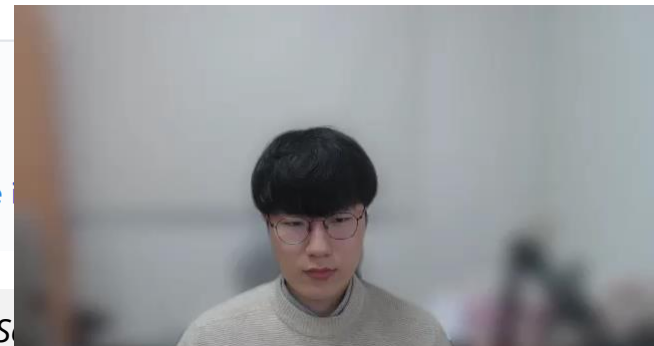
 **Review required**
At least 1 approving review is required by reviewers with write access. [Learn more.](#)

 **All checks have passed**
4 successful checks

 **Merging is blocked**
Merging can be performed automatically with 1 approving review.

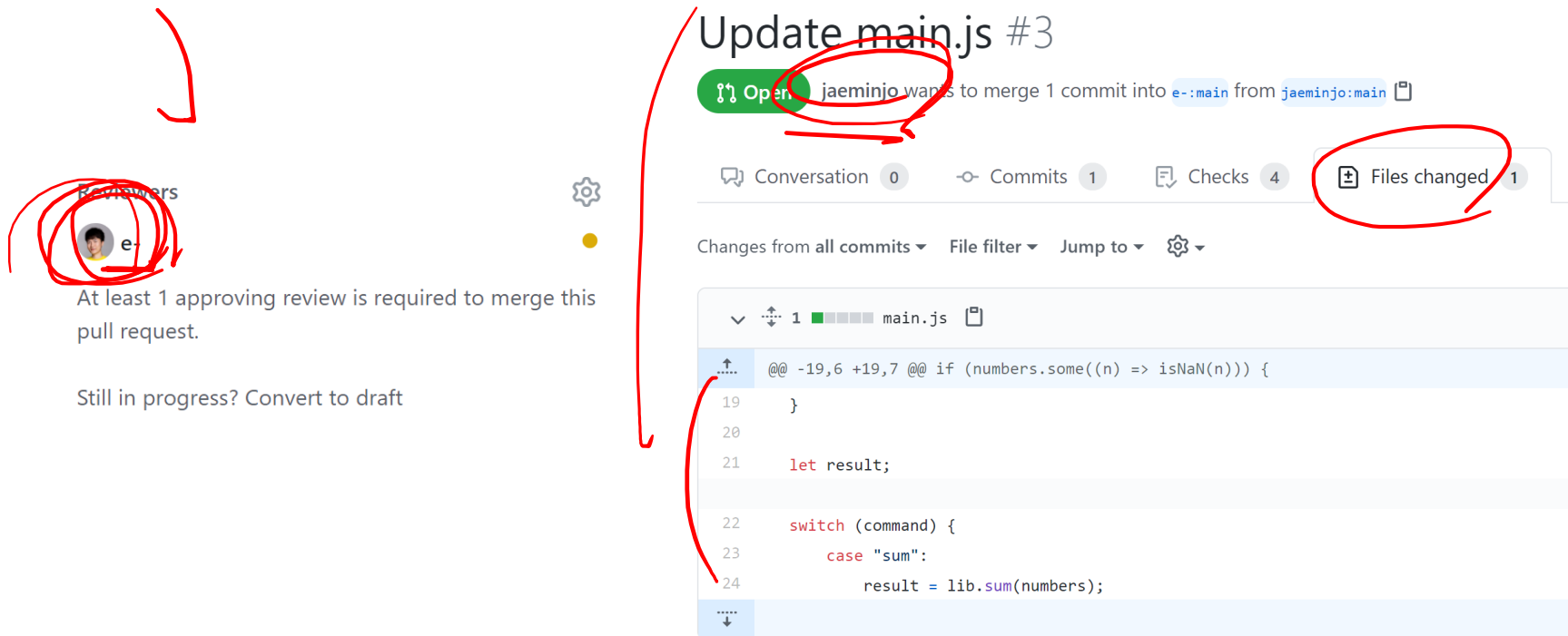
As an administrator, you may still merge this pull request.

Merge pull request ▼ You can also [open this in GitHub Desktop](#) or view [command line](#)



Code Review

- To review the pull request, go to the “File changed” tab.

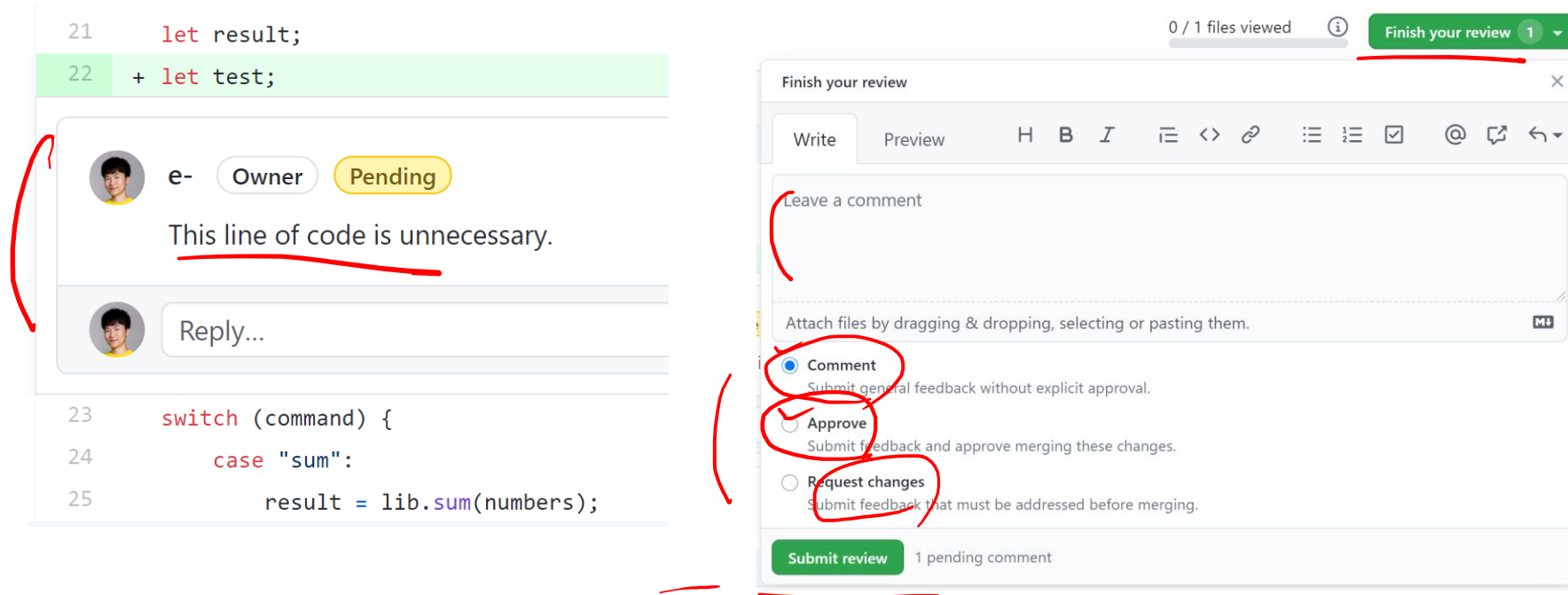


The screenshot shows a GitHub pull request titled "Update main.js #3". On the left, the "Reviewers" section is circled in red, showing a reviewer's profile and a message: "At least 1 approving review is required to merge this pull request." Below this, it says "Still in progress? Convert to draft". In the center, the "Files changed" tab is selected and circled in red. The "Open" button is also circled in red. The code diff for "main.js" is shown, with a red arrow pointing to the "if" statement on line 19: `@@ -19,6 +19,7 @@ if (numbers.some((n) => isNaN(n))) {`. The code snippet includes lines 19 through 24, showing a switch statement for a "sum" command.



Code Review

- You can comment on the files changed line by line.
- Discuss with the person who made the pull request.
- If you are done, finish the review by approving the request.



The screenshot displays a code review interface. On the left, a code diff shows a change on line 22: `+ let test;`. Below the code, a review comment from user 'e-' (Owner, Pending) states: This line of code is unnecessary. A red arrow points to the comment. On the right, a 'Finish your review' modal is open. It has tabs for 'Write' and 'Preview'. The 'Write' tab is active, showing a 'Leave a comment' text area. Below the text area, there are three radio button options: 'Comment' (selected), 'Approve', and 'Request changes'. Each option has a brief description. At the bottom of the modal, there is a 'Submit review' button and a status indicator '1 pending comment'. Red annotations highlight the 'Comment' option and the 'Submit review' button.

```
21  let result;  
22  + let test;  
  
23  switch (command) {  
24      case "sum":  
25          result = lib.sum(numbers);
```

0 / 1 files viewed Finish your review 1

Finish your review

Write Preview H B I

Leave a comment

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

☒ **Comment**
Submit general feedback without explicit approval.

☐ **Approve**
Submit feedback and approve merging these changes.

☐ **Request changes**
Submit feedback that must be addressed before merging.

Submit review 1 pending comment

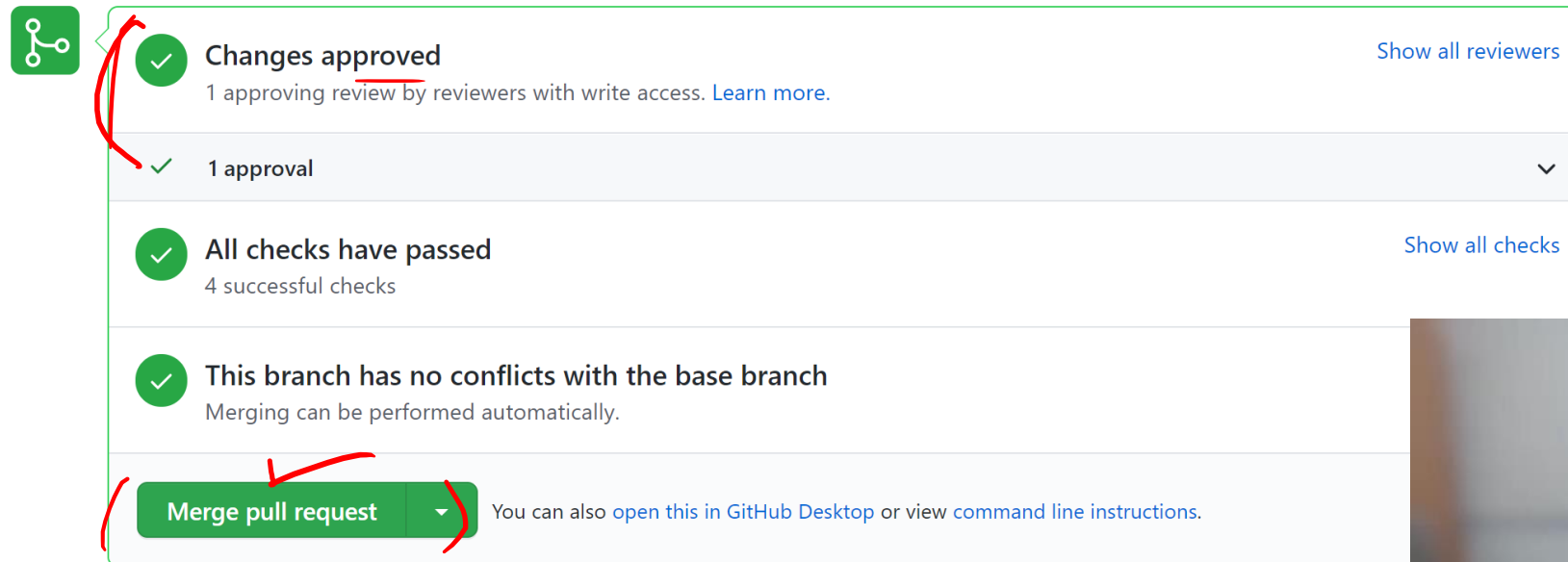


Code Review

- To make your pull request approved:
 - Pass all tests
 - No merge conflict
 - Receive an enough number of reviews

13

Add more commits by pushing to the `main` branch on `jaeminjo/skku-stat`.



The screenshot shows the GitHub pull request status interface. It includes a green checkmark icon, a green circle with a checkmark, and the text "Changes approved". Below this, it says "1 approving review by reviewers with write access. [Learn more.](#)". There is a "Show all reviewers" link. Below this, it says "1 approval" with a dropdown arrow. There is another green circle with a checkmark and the text "All checks have passed". Below this, it says "4 successful checks" and a "Show all checks" link. Below this, it says "This branch has no conflicts with the base branch" and "Merging can be performed automatically.". At the bottom, there is a green button labeled "Merge pull request" with a dropdown arrow, and a note: "You can also [open this in GitHub Desktop](#) or view [command line instructions](#).". Red handwritten annotations include a bracket on the left side of the status box, a checkmark over the "Merge pull request" button, and a bracket on the right side of the button.

Changes approved
1 approving review by reviewers with write access. [Learn more.](#) [Show all reviewers](#)

1 approval

All checks have passed
4 successful checks [Show all checks](#)

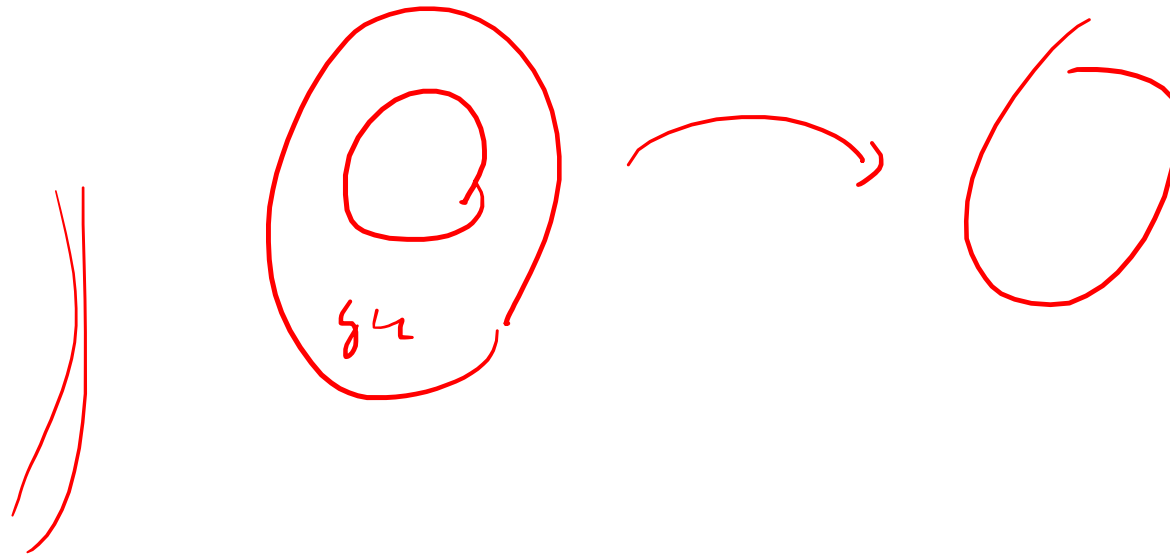
This branch has no conflicts with the base branch
Merging can be performed automatically.

Merge pull request [You can also \[open this in GitHub Desktop\]\(#\) or view \[command line instructions\]\(#\).](#)



Documentation

- To make your code accessible by others, documentation is a must.
 - Usually written in README.md using Markdown
- A brief overview
- Links to resources
- Installation
- API reference
- Examples
- Releases (versions)
- How to contribute
- License
- Code of Conduct



- <https://github.com/microsoft/vscode>

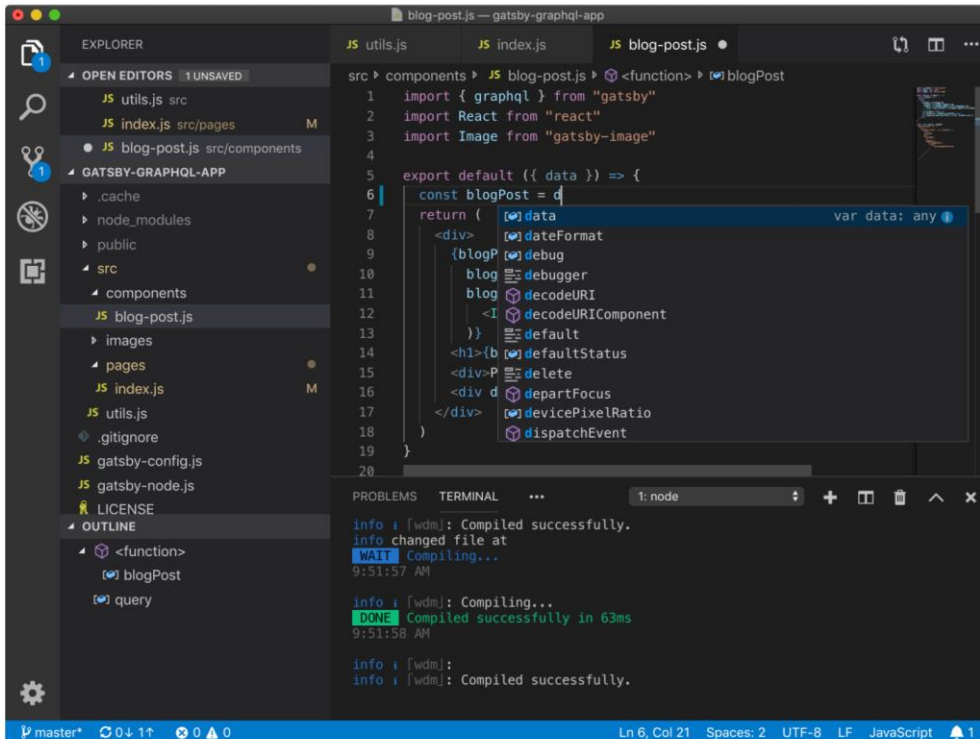
Visual Studio Code - Open Source ("Code - OSS")

feature-request issues 2.2k open bug issues 1.1k open chat on github

The Repository

This repository ("code - oss") is where we (Microsoft) develop the Visual Studio Code product together with the community. Not only do we work on code and issues here, we also publish our [roadmap](#), [monthly iteration plans](#), and our [endgame plans](#). This source code is available to everyone under the standard MIT license.

Visual Studio Code



Contributing

There are many ways in which you can participate in the project, for example:

- Submit bugs and feature requests, and help us verify as they are checked in
- Review source code changes
- Review the documentation and make pull requests for anything from typos to new content

If you are interested in fixing issues and contributing directly to the code base, please see the document [How to Contribute](#), which covers the following:

- How to build and run from source
- The development workflow, including debugging and running tests
- Coding guidelines
- Submitting pull requests
- Finding an issue to work on
- Contributing to translations

Feedback

- Ask a question on [Stack Overflow](#)
- Request a new feature
- Upvote popular feature requests
- File an issue
- Follow [@code](#) and let us know what you think!

See our [wiki](#) for a description of each of these channels and information channels.



Open-Source

Related Projects

Many of the core components and extensions to VS Code live in their own repositories on GitHub. For example, the [node debug adapter](#) and the [mono debug adapter](#) have their own repositories. For a complete list, please visit the [Related Projects](#) page on our [wiki](#).

Bundled Extensions

VS Code includes a set of built-in extensions located in the [extensions](#) folder, including grammars and snippets for many languages. Extensions that provide rich language support (code completion, Go to Definition) for a language have the suffix `language-features`. For example, the `json` extension provides coloring for `JSON` and the `json-language-features` provides rich language support for `JSON`.

Development Container

This repository includes a Visual Studio Code Remote - Containers / Codespaces development container.

- For [Remote - Containers](#), use the **Remote-Containers: Open Repository in Container...** command which creates a Docker volume for better disk I/O on macOS and Windows.
- For Codespaces, install the [Visual Studio Codespaces](#) extension in VS Code, and use the **Codespaces: Create New Codespace** command.

Docker / the Codespace should have at least **4 Cores and 6 GB of RAM (8 GB recommended)** to run full build. See the [development container README](#) for more information.

Code of Conduct

This project has adopted the [Microsoft Open Source Code of Conduct](#). For more information see the [Code of Conduct FAQ](#) or contact opencode@microsoft.com with any additional questions or comments.

License

Copyright (c) Microsoft Corporation. All rights reserved.

Licensed under the [MIT](#) license.



- <https://github.com/d3/d3-array>

Statistics

Methods for computing basic summary statistics.

`# d3.min(iterable[, accessor])` · [Source](#), [Examples](#)

Returns the minimum value in the given *iterable* using natural order. If the iterable contains no comparable values, returns undefined. An optional *accessor* function may be specified, which is equivalent to calling `Array.from` before computing the minimum value.

Unlike the built-in `Math.min`, this method ignores undefined, null and NaN values; this is useful for ignoring missing data. In addition, elements are compared using natural order rather than numeric order. For example, the minimum of the strings `["20", "3"]` is `"20"`, while the minimum of the numbers `[20, 3]` is `3`.

See also [extent](#).

`# d3.minIndex(iterable[, accessor])` · [Source](#), [Examples](#)

Returns the index of the minimum value in the given *iterable* using natural order. If the iterable contains no comparable values, returns `-1`. An optional *accessor* function may be specified, which is equivalent to calling `Array.from` before computing the minimum value.

Unlike the built-in `Math.min`, this method ignores undefined, null and NaN values; this is useful for ignoring missing data. In addition, elements are compared using natural order rather than numeric order. For example, the minimum of the strings `["20", "3"]` is `"20"`, while the minimum of the numbers `[20, 3]` is `3`.

`# d3.max(iterable[, accessor])` · [Source](#), [Examples](#)

Returns the maximum value in the given *iterable* using natural order. If the iterable contains no comparable values, returns undefined. An optional *accessor* function may be specified, which is equivalent to calling `Array.from` before computing the maximum value.

Unlike the built-in `Math.max`, this method ignores undefined values; this is useful for ignoring missing data. In addition, elements are compared using natural order rather than numeric order. For example, the maximum of the strings `["20", "3"]` is `"3"`, while the maximum of the numbers `[20, 3]` is `20`.

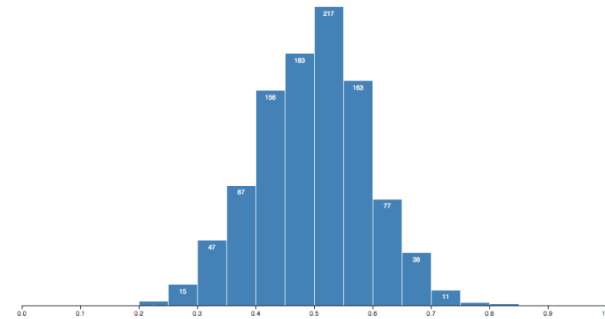
See also [extent](#).

`# d3.disjoint(a, b)` · [Source](#)

Returns true if *a* and *b* are disjoint: if *a* and *b* contain no shared value.

```
d3.disjoint([1, 3], [2, 4]) // true
```

Bins



Binning groups discrete samples into a smaller number of consecutive, non-overlapping intervals. They are often used to visualize the distribution of numerical data as histograms.

`# d3.bin()` · [Source](#), [Examples](#)

Constructs a new bin generator with the default settings.

`# bin(data)` · [Source](#), [Examples](#)

Bins the given iterable of *data* samples. Returns an array of bins, with each bin containing an array of elements from the input *data*. Thus, the `length` of the bin is the number of elements in the bin, plus any additional attributes:

- `x0` - the lower bound of the bin (inclusive).
- `x1` - the upper bound of the bin (exclusive, except for the last bin).



Documentation

- You can embed documentation in the code.
- e.g., jsdoc (<https://jsdoc.app/>)

Handwritten red notes:
// ...
def utc():

Class: Book

```
/**  
 * Represents a book.  
 * @constructor  
 * @param {string} title - The title of the book.  
 * @param {string} author - The author of the book.  
 */  
function Book(title, author) {}
```

Book(title, author)

new Book(title, author)

Represents a book.

Parameters:

| Name | Type | Description |
|--------|--------|------------------------|
| title | string | The title of the book |
| author | string | The author of the book |

Source: [temp.js, line 7](#)



Tips for Documentation

- Apply formatting (Markdown!)
 - Readable and consistent
- ✓ • Provide quick start instructions
- Provide examples (with **expected outputs**)
- Document how to document
- Code of conduct
- A feedback mechanism
- An issue template



Code of Conduct

- Safe work environments
- Mutual respect
- Empathy and kindness
- Constructive feedback
- No trolling, insulting, harassment, etc.

Code of Conduct

This project has adopted the [Microsoft Open Source Code of Conduct](#). For more information see the [Code of Conduct FAQ](#) or contact opencode@microsoft.com with any additional questions or comments.



Microsoft's CoC

- <https://opensource.microsoft.com/codeofconduct/>

Our Pledge

We as members, contributors, and leaders pledge to make participation in our community a harassment-free experience for everyone, regardless of age, body size, visible or invisible disability, ethnicity, sex characteristics, gender identity and expression, level of experience, education, socio-economic status, nationality, personal appearance, race, caste, color, religion, or sexual identity and orientation.

We pledge to act and interact in ways that contribute to an open, welcoming, diverse, inclusive, and healthy community.

Our Standards

Examples of behavior that contributes to a positive environment for our community include:

- Demonstrating empathy and kindness toward other people
- Being respectful of differing opinions, viewpoints, and experiences
- Giving and gracefully accepting constructive feedback
- Accepting responsibility and apologizing to those affected by our mistakes, and learning from the experience
- Focusing on what is best not just for us as individuals, but for the overall community

Examples of unacceptable behavior include:

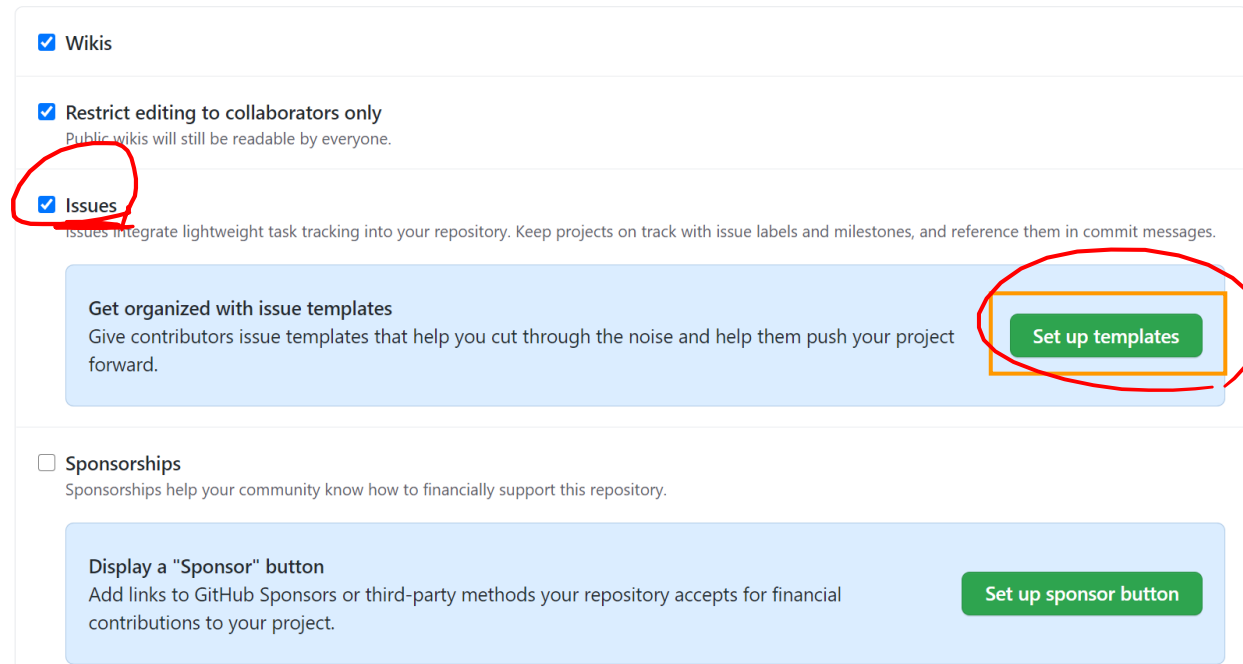
- The use of sexualized language or imagery, and sexual attention or advances of any kind
- Trolling, insulting or derogatory comments, and personal or political attacks
- Public or private harassment
- Disruptive behavior
- Publishing others' private information, such as a physical or electronic address, without explicit permission
- Other conduct which could reasonably be considered inappropriate for the community



Issue Templates

- Use a consistent format for issues or bug reports
- Setting -> Options -> Features

Features



☒ Wikis

☒ Restrict editing to collaborators only
Public wikis will still be readable by everyone.

☒ Issues
Issues integrate lightweight task tracking into your repository. Keep projects on track with issue labels and milestones, and reference them in commit messages.

Get organized with issue templates
Give contributors issue templates that help you cut through the noise and help them push your project forward.

☐ Sponsorships
Sponsorships help your community know how to financially support this repository.

Display a "Sponsor" button
Add links to GitHub Sponsors or third-party methods your repository accepts for financial contributions to your project.



Issue Template

Bug report

Create a report to help us improve

Close preview



Issue: Bug report

Create a report to help us improve

Describe the bug

A clear and concise description of what the bug is.

To Reproduce

Steps to reproduce the behavior:

1. Go to '...'
2. Click on '....'
3. Scroll down to '....'
4. See error

Expected behavior

A clear and concise description of what you expected to happen.

Screenshots

If applicable, add screenshots to help explain your problem.

Desktop (please complete the following information):

- OS: [e.g. iOS]
- Browser [e.g. chrome, safari]
- Version [e.g. 22]

Smartphone (please complete the following information):

- Device: [e.g. iPhone6]
- OS: [e.g. iOS8.1]
- Browser [e.g. stock browser, safari]
- Version [e.g. 22]

Additional context

Issue: Bug report

Create a report to help us improve. If this doesn't look right, [choose a different type](#).



Title

Write

Preview

H B I         

****Describe the bug****

A clear and concise description of what the bug is.

****To Reproduce****

Steps to reproduce the behavior:

1. Go to '...'
2. Click on '....'
3. Scroll down to '....'
4. See error

****Expected behavior****

A clear and concise description of what you expected to happen.

****Screenshots****

If applicable, add screenshots to help explain your problem.

****Desktop (please complete the following information):****

- OS: [e.g. iOS]
- Browser [e.g. chrome, safari]
- Version [e.g. 22]

****Smartphone (please complete the following information):****

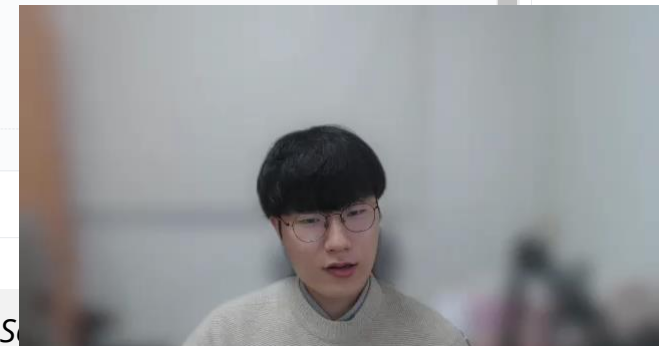
- Device: [e.g. iPhone6]
- OS: [e.g. iOS8.1]
- Browser [e.g. stock browser, safari]
- Version [e.g. 22]

****Additional context****

Add any other context about the problem here.

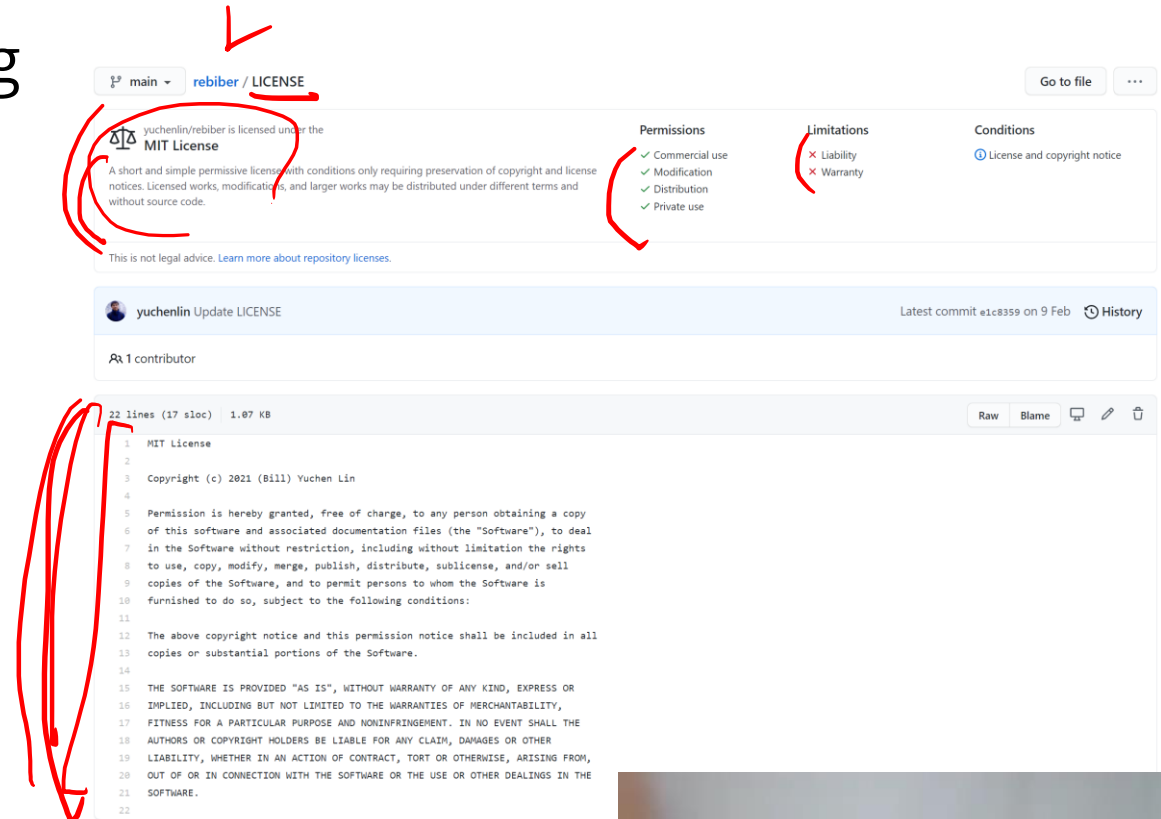
Attach files by dragging & dropping, selecting or pasting them.

 Styling with Markdown is supported



License

- **License:** a legal instrument governing the use or redistribution of software.
- Usually in LICENSE file in a repo
- Commercial use
- Modification
- Distribution
- Private use
- Liability
- Warranty



The screenshot shows the GitHub interface for the file `rebiber / LICENSE`. The license is identified as the MIT License. A table lists the permissions and limitations:

| Permissions | Limitations | Conditions |
|----------------------------------------------------|-----------------------------------------------|------------------------------------------------------------------|
| <input checked="" type="checkbox"/> Commercial use | <input checked="" type="checkbox"/> Liability | <input checked="" type="checkbox"/> License and copyright notice |
| <input checked="" type="checkbox"/> Modification | <input checked="" type="checkbox"/> Warranty | |
| <input checked="" type="checkbox"/> Distribution | | |
| <input checked="" type="checkbox"/> Private use | | |

Below the table, the full text of the MIT License is displayed, starting with "MIT License" and "Copyright (c) 2021 (Bill) Yuchen Lin". Red annotations highlight the license name, the permissions table, and the full license text.



License

- You can include a LICENSE file when you create a repo.
- <https://choosealicense.com/licenses/>
- The MIT license is one of the most permissive licenses.

MIT License

A short and simple permissive license with conditions only requiring preservation of copyright and license notices. Licensed works, modifications, and larger works may be distributed under different terms and without source code.

Permissions

- Commercial use
- Distribution
- Modification
- Private use

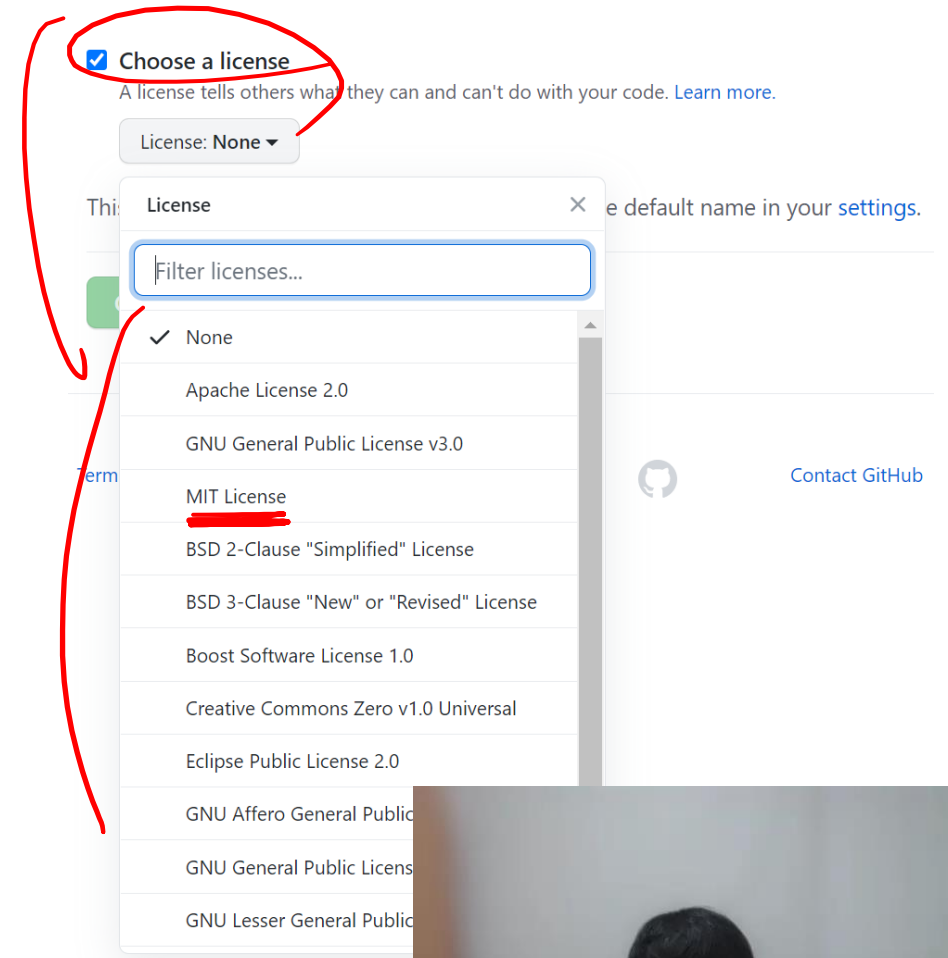
Conditions

- License and copyright notice

Limitations

- Liability
- Warranty

[View full MIT License »](#)



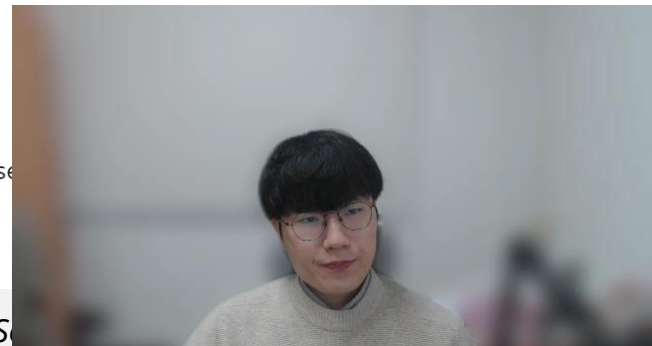
- Some licenses require you to use the same license for modified code when you release your software to public.
 - GNU GPLv3, GNU LGPLv3, ...
- Be careful when you use code under these licenses.
 - You must disclose your source!
 - Software under GPL (Git, MariaDB, MySQL, Notepad++, ...)...

Does the GPL require that source code of modified versions be posted to the public? ([#GPLRequireSourcePostedPublic](#))

The GPL does not require you to release your modified version, or any part of it. You are free to make modifications and use them privately, without ever releasing them. This applies to organizations (including companies), too; an organization can make a modified version and use it internally without ever releasing it outside the organization.

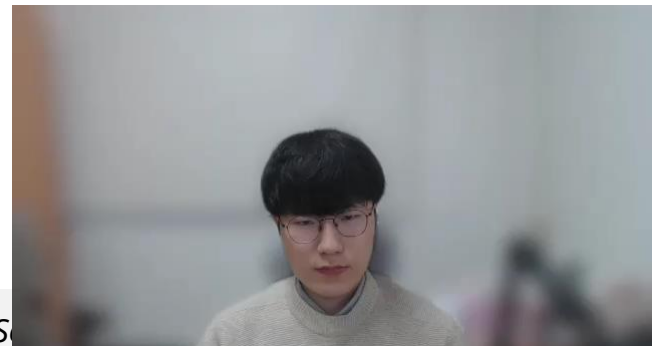
But *if* you release the modified version to the public in some way, the GPL requires you to make the modified source code available to the program's users, under the GPL.

Thus, the GPL gives permission to release the modified program in certain ways, and not in other ways; but the decision of whether to release is up to you.



Summary: Collaboration

- **Markdown** is a simple markup language for documents in GitHub.
 - Issues/pull requests/documentation/...
- External collaborators contribute to an open-source projects by making **a fork and pull requests**.
- Issue management and code review
- Document your source code to make it reusable.
- **License**: commercial use/modification/distribution/liability/warranty



Summary: Collaboration

- Can you figure out what the following files/directories do?

- .github/workflows
- .git
- .editorconfig
- .gitignore
- ISSUE_TEMPLATE
- LICENSE.md
- main.js
- package.json
- README.md
- node_modules

