MLH Localhost – Get Crackin' With Git using GitKraken

Facilitator's Guide

STEP 0: Before The Workshop

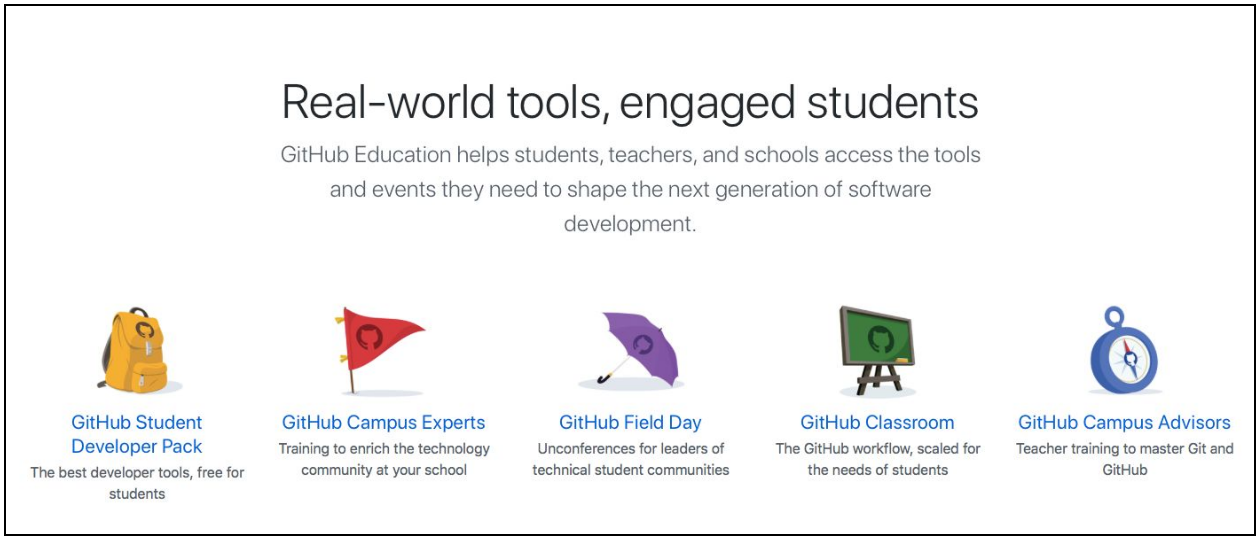
- We're glad to hear that you're organizing an MLH Localhost event with GitKraken! The *Get Crackin' with Git using GitKraken* workshop requires you and all participants to create GitKraken accounts. Make sure to create your account first, and then provide your referral link in the presentation slides for others to use when creating their accounts. GitKraken has an awesome referral program that allows you to win prizes like a custom Nintendo Switch, and following this process will ensure that all workshop participants' accounts are associated with your referral code.

STEP 1: Introduce Concepts

- Git is a distributed version-control system for tracking changes to software. It is used to keep the integrity of the code while numerous software developers work on it in distinct **branches** and then a project manager can approve branches to fold into the **master** branch, or main branch. Git is used primarily from the command line.
- GitKraken is a web-based hosting service that creates a visual interface that makes it easier for developers to understand how their branches interact and better identify conflicts in a way that is more intuitive than the command line.
- In this workshop you will learn best practices for collaborating using Git and how GitKraken simplifies collaboration by creating a simple HTML/CSS webpage with a team.

STEP 2: Sign Up for GitHub and GitKraken

- There are two different processes to sign up for GitHub, one if you are a student and one if you are not.
  - Students should navigate to: http://mlhlocal.host/github-edu

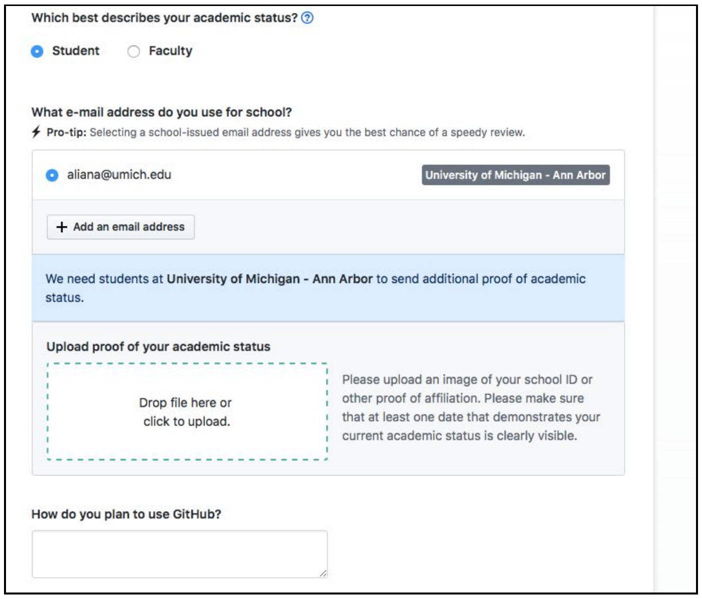


Real-world tools, engaged students

GitHub Education helps students, teachers, and schools access the tools and events they need to shape the next generation of software development.

GitHub Student Developer Pack
The best developer tools, free for students
GitHub Campus Experts
Training to enrich the technology community at your school
GitHub Field Day
Unconferences for leaders of technical student communities
GitHub Classroom
The GitHub workflow, scaled for the needs of students
GitHub Campus Advisors
Teacher training to master Git and GitHub

- You will need to take a photo of your student ID to show proof of your student status. If verification takes too long, you can simply sign up as a non-student.



Which best describes your academic status?

[✓ Student] Faculty

What e-mail address do you use for school?

Pro-tip: Selecting a school-issued email address gives you the best chance of a speedy review.

aliana@umich.edu — University of Michigan - Ann Arbor

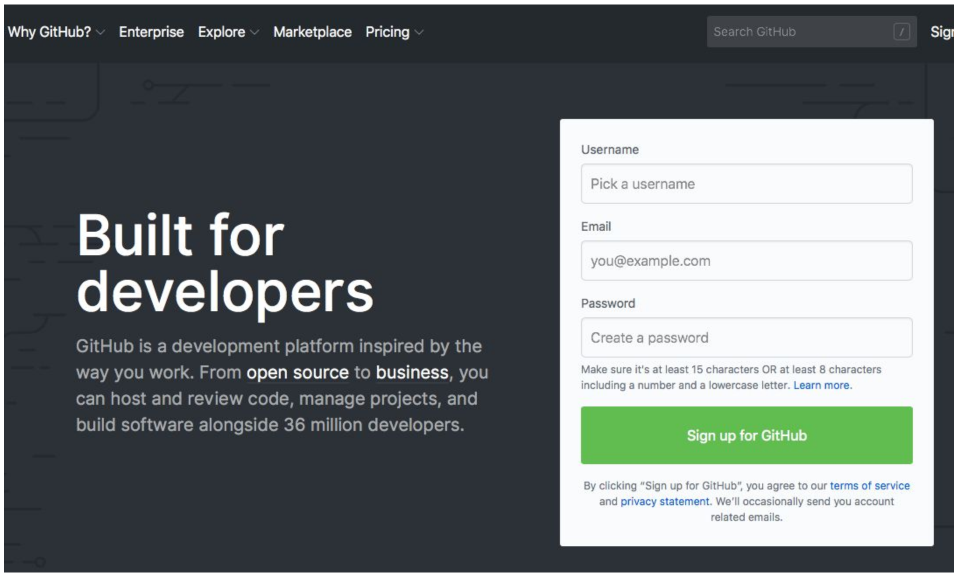
We need students at **University of Michigan - Ann Arbor** to send additional proof of academic status.

Upload proof of your academic status

Please upload an image of your school ID or other proof of affiliation. Please make sure that at least one date that demonstrates your current academic status is clearly visible.

How do you plan to use GitHub?

- For those who are not students, navigate to Github. Choose a user name and password, or login to an existing account: http://mlhlocal.host/github-signup



Built for developers

GitHub is a development platform inspired by the way you work. From open source to business, you can host and review code, manage projects, and build software alongside 36 million developers.

Username: Pick a username

Email: you@example.com

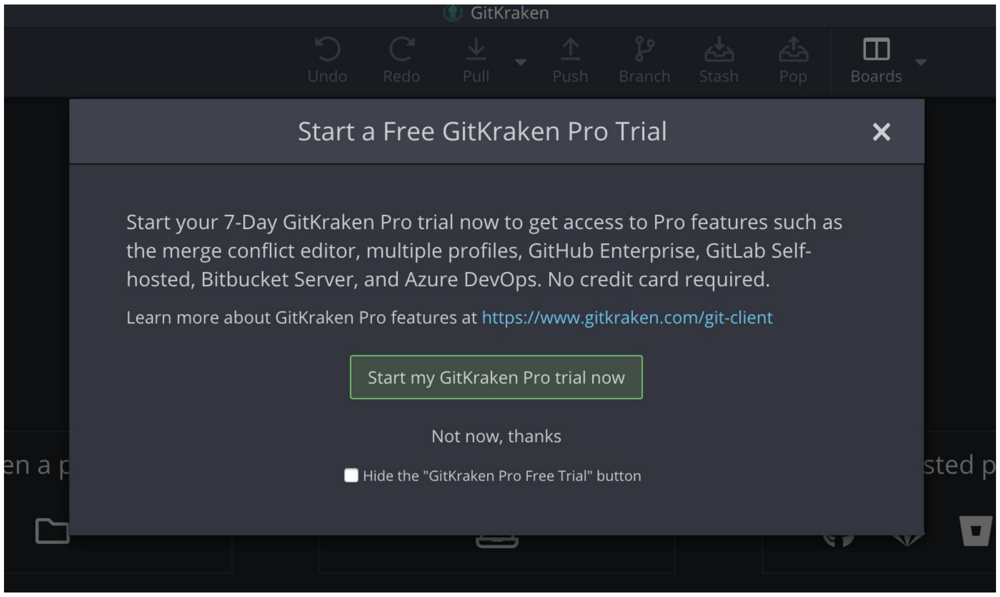
Password: Create a password

Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. Learn more.

Sign up for GitHub

By clicking "Sign up for GitHub", you agree to our terms of service and privacy statement. We'll occasionally send you account related emails.

- Then sign up for GitKraken using your new Github account: http://mlhlocal.host/gitkraken
- You will have two options of products you can download. Choose the Git Client option.
- Once you choose the installation package appropriate for your computer, drag the GitKraken package into your Applications folder.
- When you open the GitKraken application, you will need to sign in to GitKraken using your GitHub account to link the two platforms together.
- For those who signed in using their Edu address, they will be automatically upgraded to Pro. For those who do not have an Edu address, they will need to start a free trial.



Start a Free GitKraken Pro Trial

Start your 7-Day GitKraken Pro trial now to get access to Pro features such as the merge conflict editor, multiple profiles, GitHub Enterprise, GitLab Self-hosted, Bitbucket Server, and Azure DevOps. No credit card required.

Learn more about GitKraken Pro features at https://www.gitkraken.com/git-client

Start my GitKraken Pro trial now

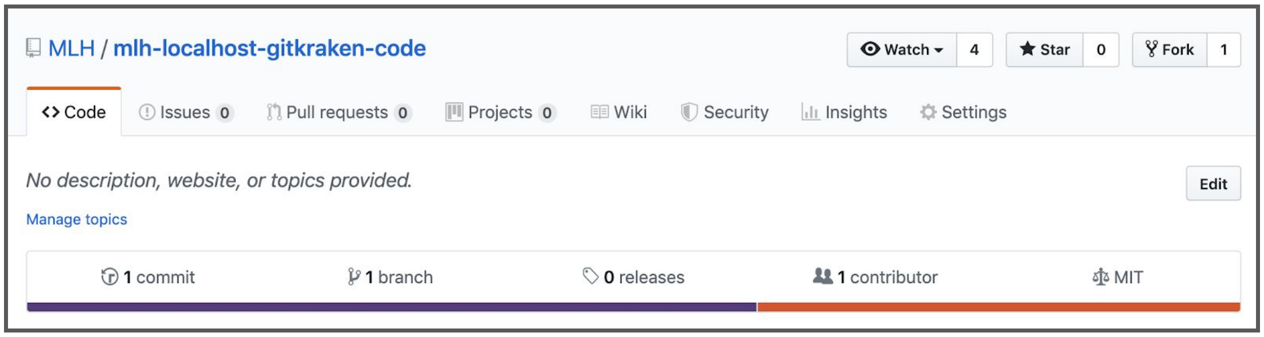
Not now, thanks

☐ Hide the "GitKraken Pro Free Trial" button

- What if you have already used your Pro Trial and are not a student?

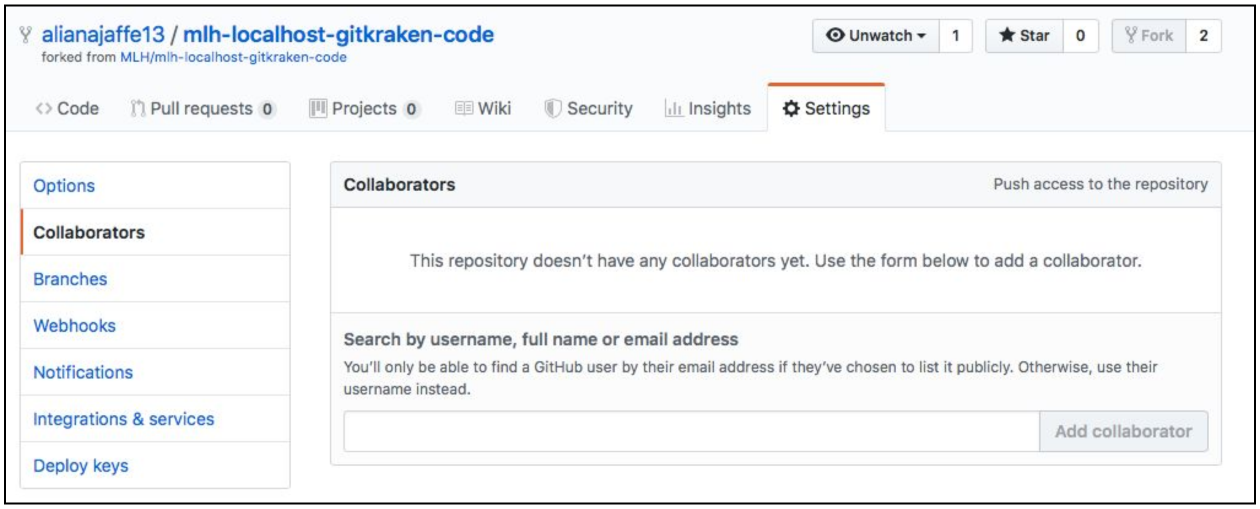
STEP 3: Get Code

- Navigate to: http://mlhlocal.host/gitkraken-code
- Choose *one* team member to fork the code by clicking "Fork" on the right side of the Git repo. Make sure to do this only once or it might be confusing



No description, website, or topics provided.

- Make sure this *same* team member adds the other collaborators. If a different person adds collaborators, you might run into sharing permission problems.
- Navigate to **Settings > Collaborators** and enter each team member's Github name. Collaborators' Github name should be at the Top Left of the page.

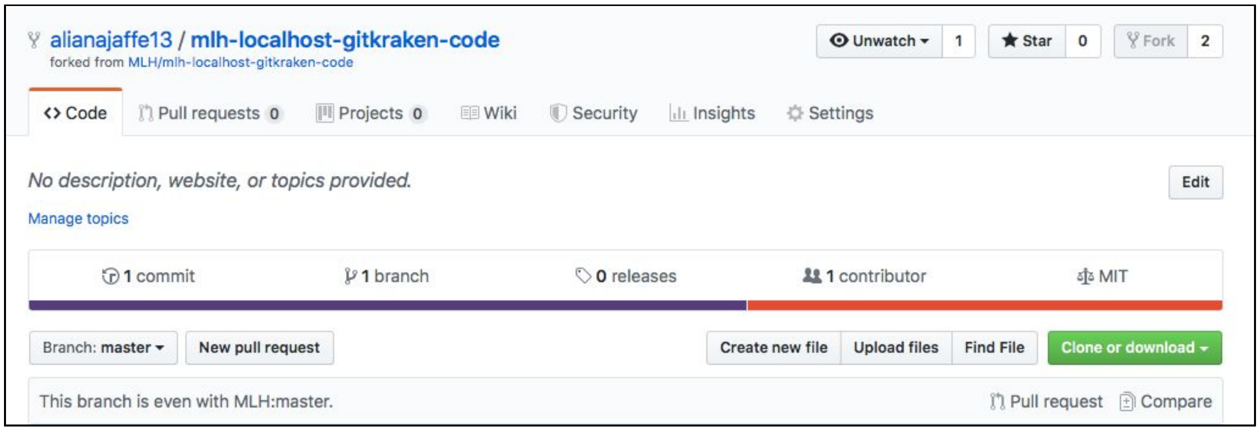


This repository doesn't have any collaborators yet. Use the form below to add a collaborator.

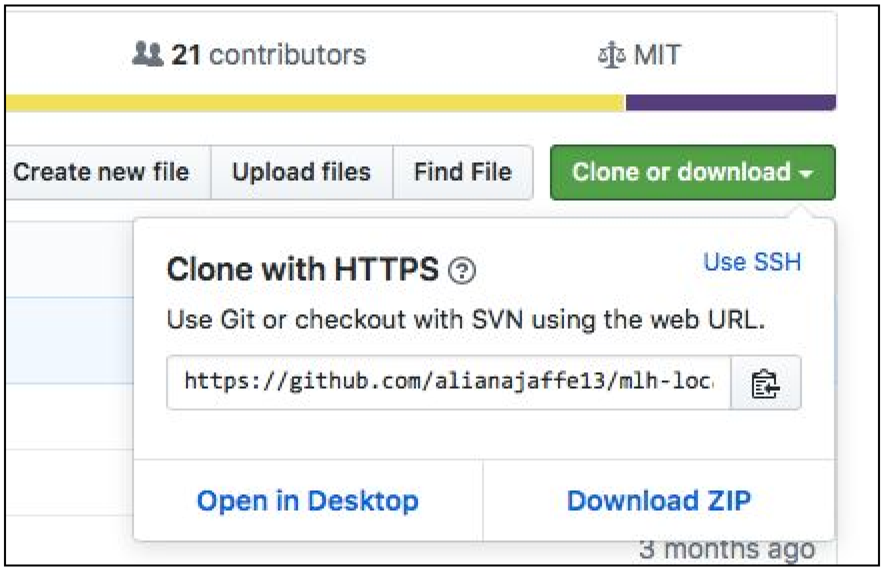
Search by username, full name or email address

You'll only be able to find a GitHub user by their email address if they've chosen to list it publicly. Otherwise, use their username instead.

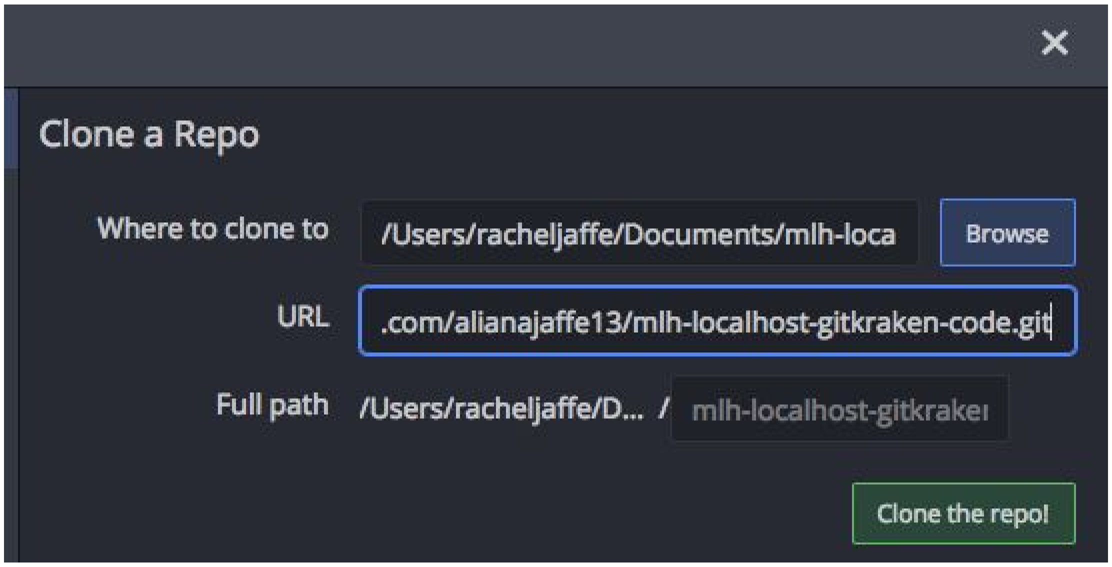
- Once this one person has added all of the collaborators, everyone should refresh their page and then navigate back to the page below.



- Then return to GitKraken and Select **Clone Repo**.



- Then, copy the address. Do not click “Open in Desktop” or “Download Zip.” If you do Download the repo, you will start an entirely new repo that cannot be merged back into the rest of the master branch because the code will have different sha ids.
- Navigate back to GitKraken, and paste the link into the place where it says **URL**.

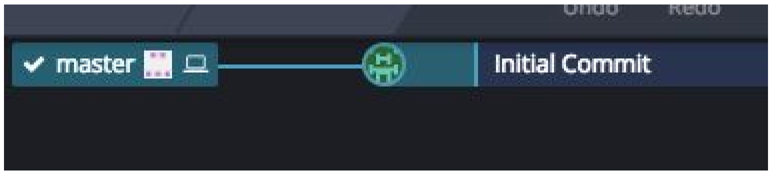


Where to clone to: /Users/racheljaffe/Documents/mlh-loca

URL: .com/alianajaffe13/mlh-localhost-gitkraken-code.git

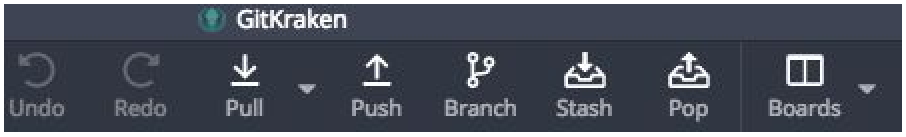
Full path: /Users/racheljaffe/D... / mlh-localhost-gitkraker

- When the repo has been cloned, Select “Open Now.”
- In GitKraken, you should see that you can now commit to the Master branch.

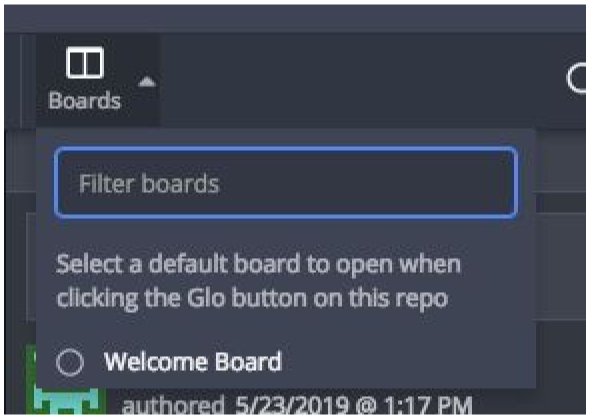


STEP 4: Access Glo Boards

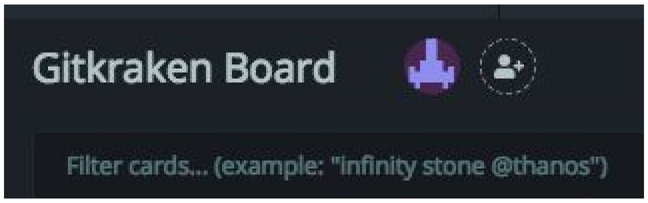
- *One person* should navigate to GitKraken, navigate to the toolbar and Select Boards.



- Select “Welcome Board.”


Boards
Filter boards
Select a default board to open when clicking the Glo button on this repo
Welcome Board
authored 5/23/2019 @ 1:17 PM

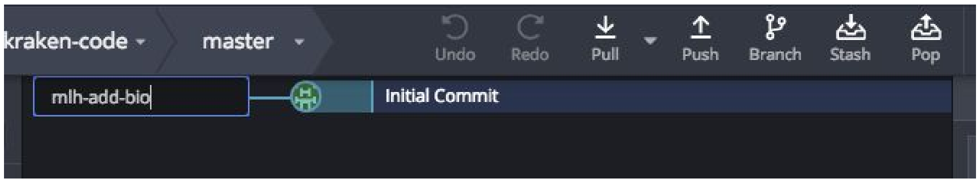
- Once someone has a board open, they can rename it by double-clicking on the Board's name. Tell the rest of your team the name of the board.
- Then, next to the Board's name participants should press the "Add People" icon to add their team members as collaborators. If at this point multiple people have made Boards, it might get confusing which Board collaborators are working on. Choose one.



- You will need either your teammates' Github names or email addresses to invite them to the Board. In later steps, if you have a hard time committing any work to a shared branch, make sure you and your teammates have all accepted being in the shared board.

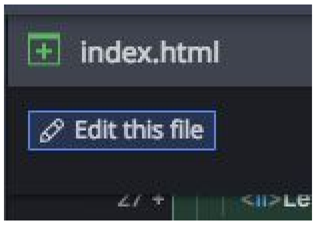
STEP 5: Commit & Pull Changes

- Go to GitKraken and add a new branch by typing a title into the blank branch field.

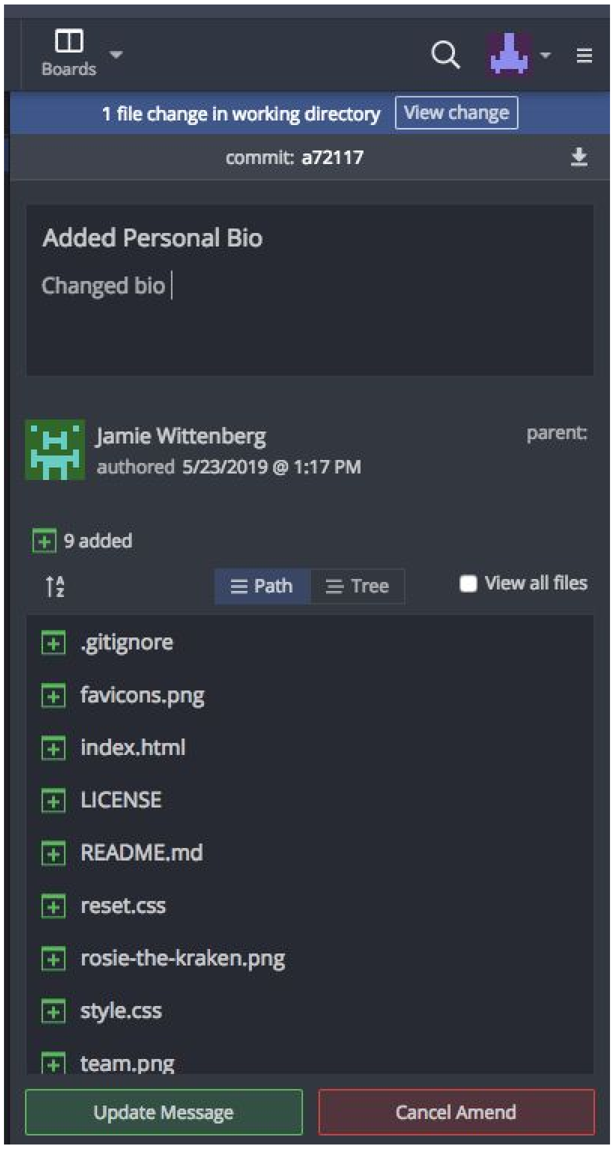


- On the right side there should be a panel. Select **View All Files** (though some might not need this step) then Select **index.html**.

- If you scroll down to line 39, you will see a bio of Rosie the Kraken. You will need to copy this code. If you highlight this code, however, you will see a message that says “Unable to Edit this code.” Before you can copy the code, you will need to go up to the Top Left and Select “Edit this file.”



- Then, copy from lines 40 to 47, and paste them at line 51. Make sure to then change the text from Rosie the Kraken to your own bio. If you have a photo on your desktop you wish to share, copy the photo link and paste it where the current code says “rosie-the-kraken.png” .
- You will need to save these changes. You can do this by double-clicking on the file name and a prompt to Save the file will appear at the top of the screen.
- Once you Select **Save** a new window will open that says “Initial Commit.” Change the **Title** and the **Description** to reflect that you have updated the code with your bio. Then Select **Update Message** at the bottom.



1 file change in working directory View change

commit: a72117

Added Personal Bio

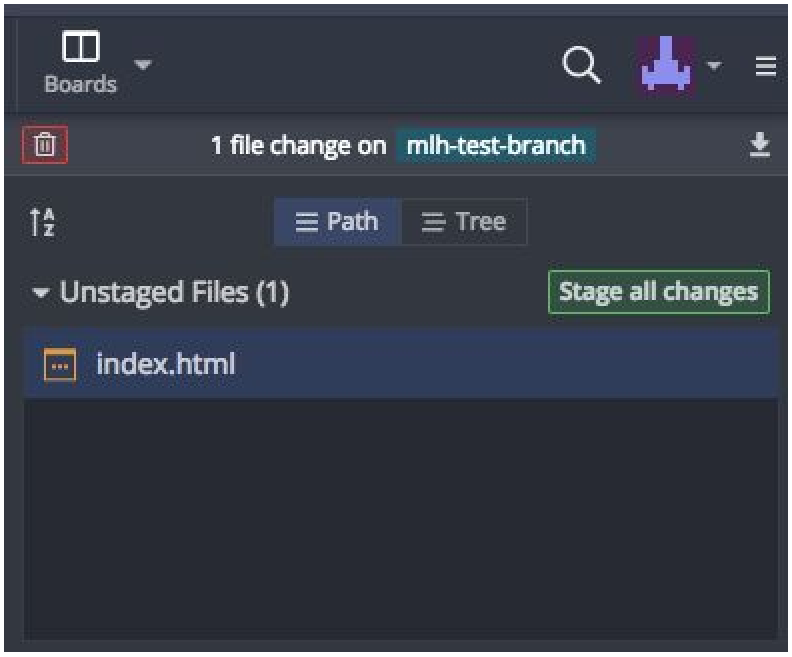
Changed bio

Jamie Wittenberg
authored 5/23/2019 @ 1:17 PM
parent:

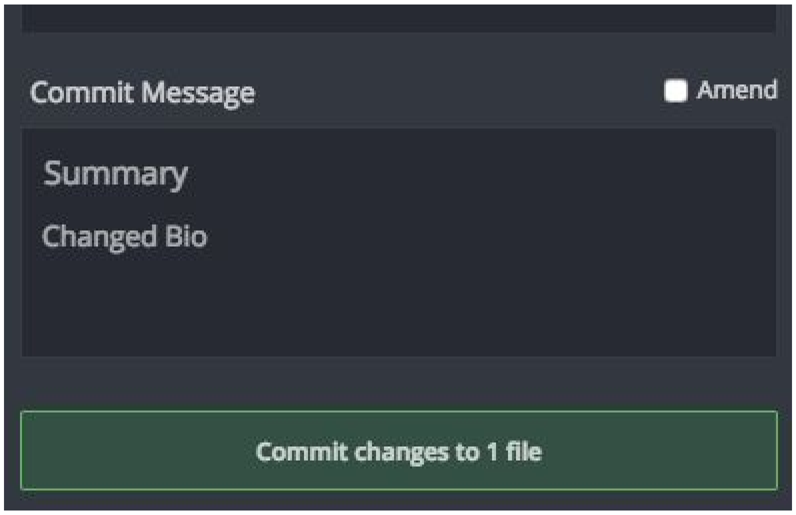
9 added

.gitignore
favicons.png
index.html
LICENSE
README.md
reset.css
rosie-the-kraken.png
style.css
team.png

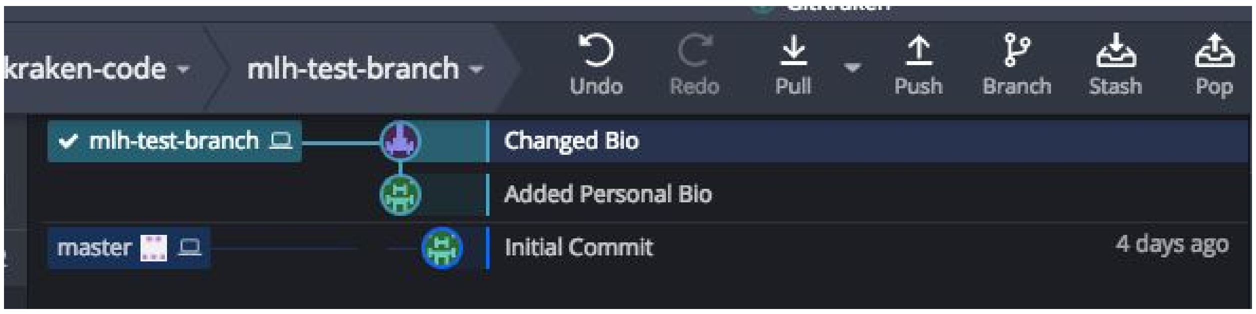
- You should see your commit was saved. There is also another way to do this by Selecting **Stage All Changes** on the right side of the panel.



- Add a description, and then **Commit** these changes.



- Once your changes are committed, at the Top panel Select **Push**.

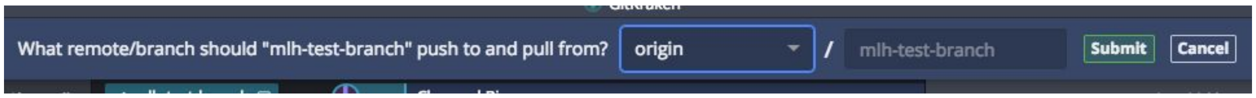


Changed Bio

Added Personal Bio

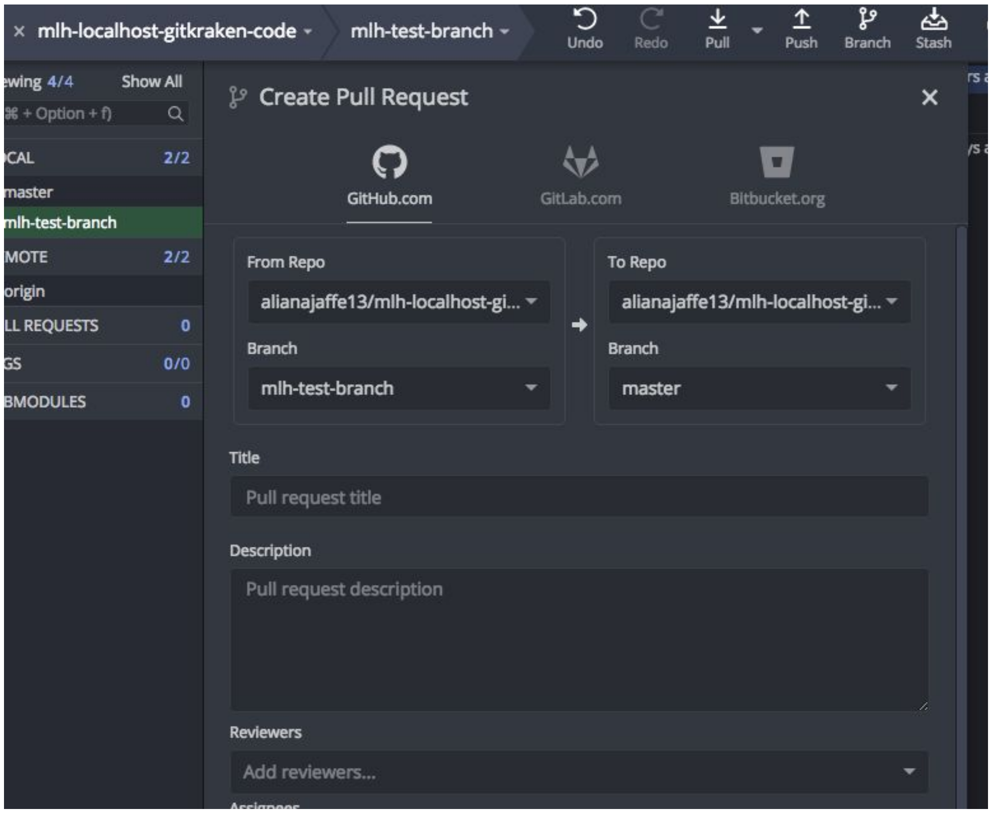
Initial Commit 4 days ago

- There will be a drop down menu that asks where you want to save your commit to. It should say “origin” as the location where you will push your commit to. Hit **Submit**!



What remote/branch should "mlh-test-branch" push to and pull from? origin / mlh-test-branch

- Then Select your branch name and right click. You should Select **Start a Pull Request.** There will also be an option to **Pull (fast-forward if possible).** If you click on this, a pop-up will say that a Pull request has started, but it will not work. Make sure to Select **Start a Pull Request** to get to the right screen.



Create Pull Request

From Repo: alianajaffe13/mlh-localhost-gi...

Branch: mlh-test-branch

To Repo: alianajaffe13/mlh-localhost-gi...

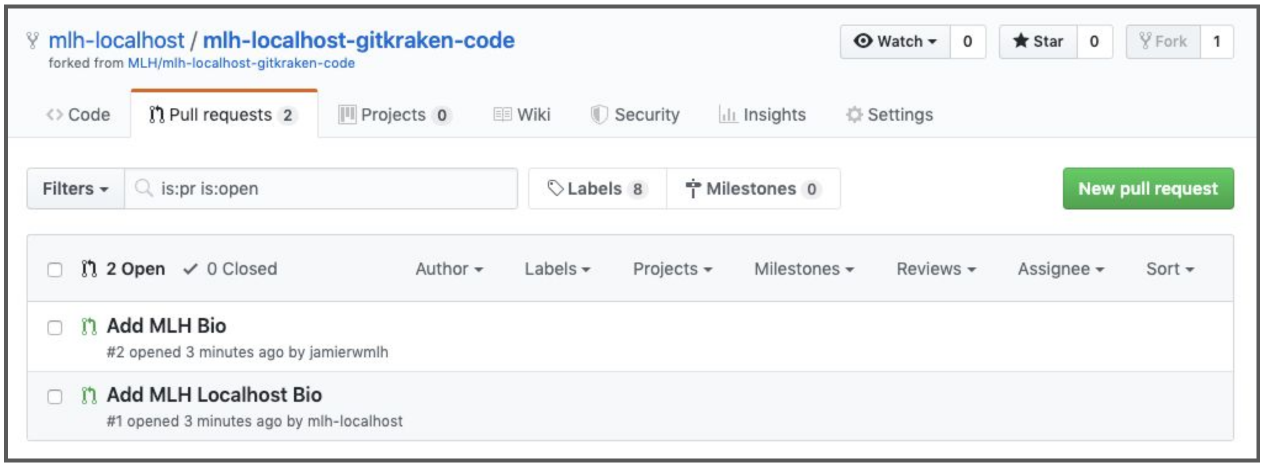
Branch: master

Title: Pull request title

Description: Pull request description

Reviewers: Add reviewers...

- Add a title and description (and get someone to review!) then Select **Create Pull Request**.
- If you got "There is nothing to compare, these repos have different commit histories" then you have accidentally downloaded the repo and are working on a new repo, that while it might look a lot like your group's repo, it cannot be merged back into the master branch. To solve this, you will need to go back to Step 2 and Clone the master repo.
- You can check whether you and your teammates were successful by navigating to GitHub and if they did everything correctly, you should be able to see their commits.



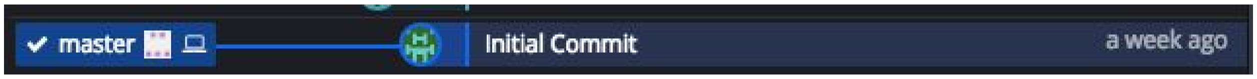
Add MLH Bio
#2 opened 3 minutes ago by jamierwmlh

Add MLH Localhost Bio
#1 opened 3 minutes ago by mlh-localhost

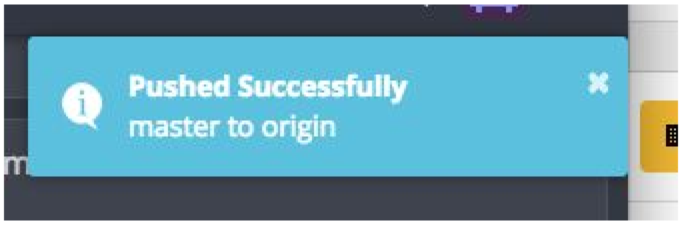
- Select **Add Review** to review your collaborators' commits! Then Select **Approve** to merge their changes.

<u>STEP 6: Merge Changes</u>

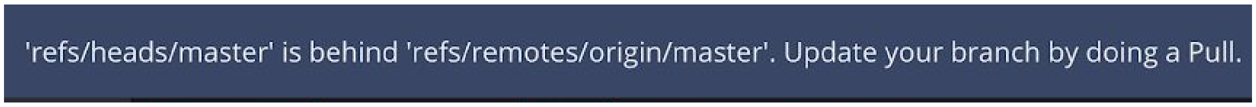
- You will want to change your version of the master branch to be remote. You will know if your branch is on the remote if it has your Git icon next to it. Double-click on the master branch until there is a check mark.



- Click **Push** at the top. You should get a banner that says the Push was successful.

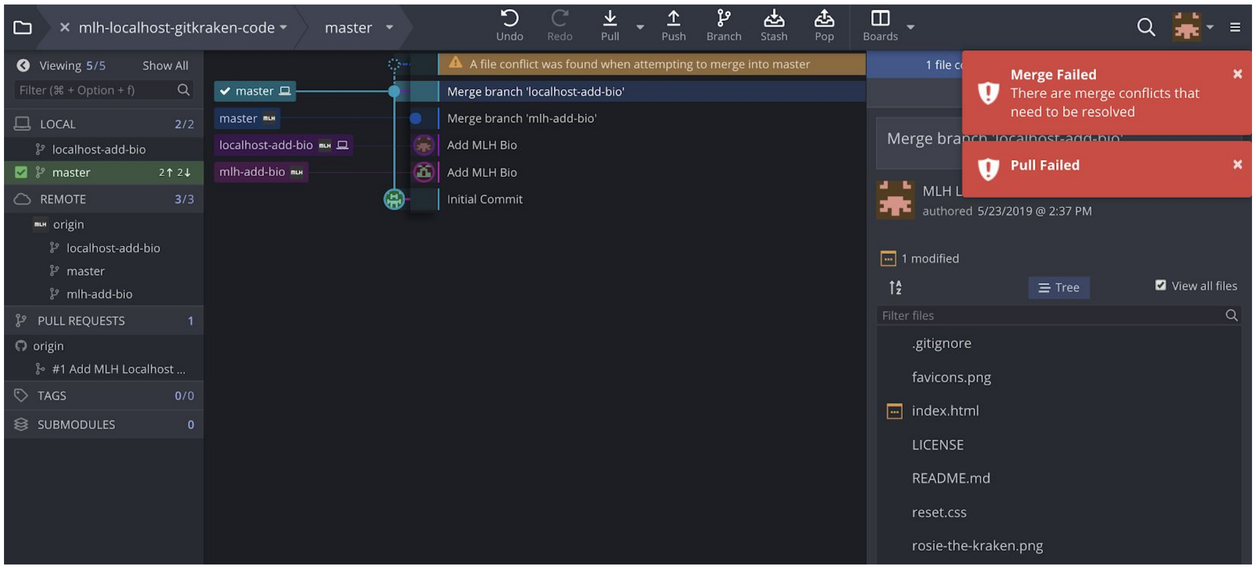


- Next click and drag your branch onto the master. A menu should pop up and you should Select **Merge <your branch name> onto master**. Some participants might ask what rebasing is (since it is located right under the merge option). When you merge code, your code base is joining with another to create one new code base but it preserves the history, which makes it easier to resolve conflicts. On the other hand, rebasing rearranges existing code, which makes a cleaner history, but makes it harder to resolve conflicts.
- If you get the warning below, it means that there has since been changes on the master branch.



'refs/heads/master' is behind 'refs/remotes/origin/master'. Update your branch by doing a Pull.

- If this happens, you will need to **Select Pull (fast forward if possible).**



A file conflict was found when attempting to merge into master

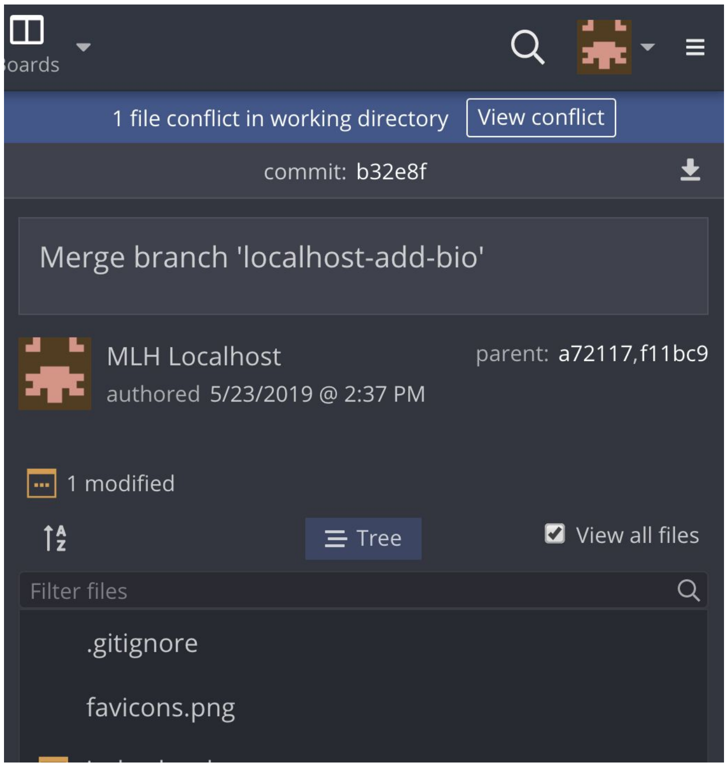
Merge branch 'localhost-add-bio'
Merge branch 'mlh-add-bio'
Add MLH Bio
Add MLH Bio
Initial Commit

Merge Failed
There are merge conflicts that need to be resolved

Pull Failed

authored 5/23/2019 @ 2:37 PM

- To complete the merge, you need to resolve it. To start doing this, you will need to first view it by Selecting **View Conflict**.



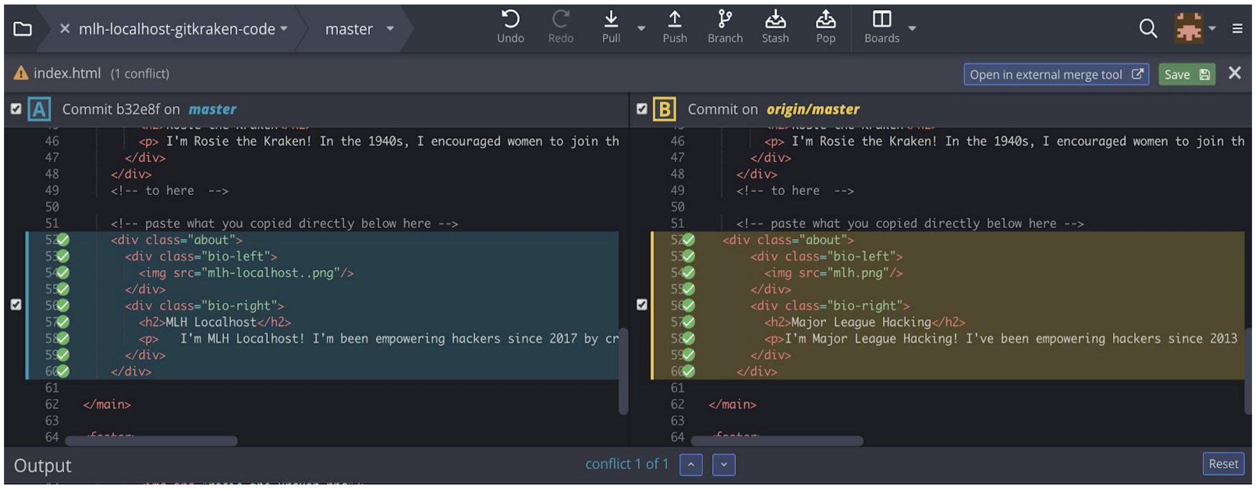
Merge branch 'localhost-add-bio'

MLH Localhost
authored 5/23/2019 @ 2:37 PM
parent: a72117,f11bc9

1 modified

.gitignore
favicons.png

- Select **index.html** to view the conflict. The window that opens will show your code stored locally on the left and the code on the right that is in the remote repository. On each side there is a checkbox. Check it. If you do not, only the bio that already exists on the master branch will appear on your website. Then Select **Save.**



index.html (1 conflict)

Commit b32e8f on *master*

```
        <p> I'm Rosie the Kraken! In the 1940s, I encouraged women to join th
      </div>
    </div>
    <!-- to here  -->

    <!-- paste what you copied directly below here -->
    <div class="about">
      <div class="bio-left">
        <img src="mlh-localhost..png"/>
      </div>
      <div class="bio-right">
        <h2>MLH Localhost</h2>
        <p>   I'm MLH Localhost! I'm been empowering hackers since 2017 by cr
      </div>
    </div>

  </main>
```

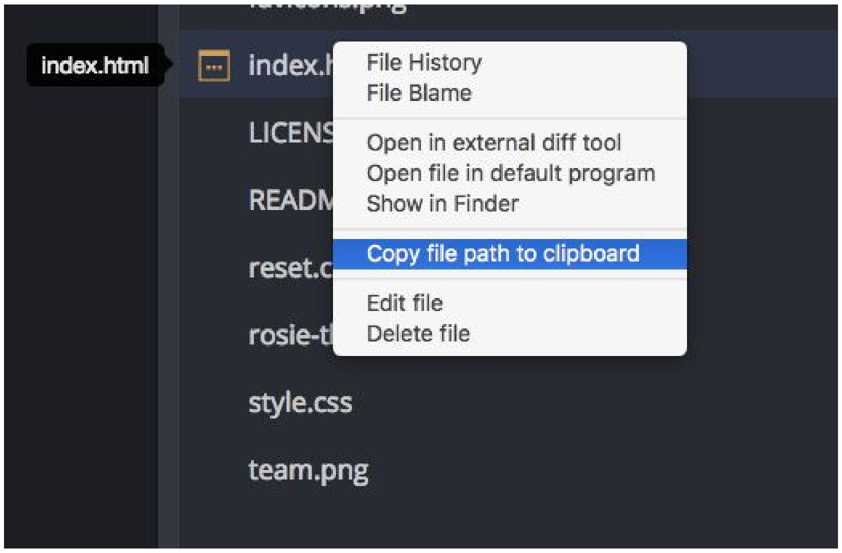
Commit on *origin/master*

```
        <p> I'm Rosie the Kraken! In the 1940s, I encouraged women to join th
      </div>
    </div>
    <!-- to here  -->

    <!-- paste what you copied directly below here -->
  <div class="about">
      <div class="bio-left">
        <img src="mlh.png"/>
      </div>
      <div class="bio-right">
        <h2>Major League Hacking</h2>
        <p>I'm Major League Hacking! I've been empowering hackers since 2013
      </div>
    </div>

  </main>
```

- These changes now need to be committed. Once you Select **Save** to save your changes, you will need to add a Commit message.
- Now, there might be errors that emerge if other people in the workshop are also trying to resolve their code at the same time. This will need to be resolved by talking

to the people around you to make sure there is only one person resolving conflicts at a time.

- Once you have worked out all of the conflicts, Right Click on **index.html.** Then copy the file path and paste it in your browser. You should see the bios of your whole group hosted on the website!



- Though the workshop might be over, you can still stay engaged with GitKraken by becoming a GitKraken Ambassador! You are well on your way by hosting a workshop already ---- make sure to remember to share your referral link with participants to get credit for hosting this workshop. When you become a GitKraken Ambassador you can stay connected with a community of industry professionals and student developers. This program will help increase your software development knowledge, build your professional skills, and provide further value to your local tech community. Get support for your events, improve your content, increase your reach, and get legendary swag! Check out the GitKraken Ambassador Program, and when you apply, don't forget to mention that you're an MLH Localhost.