

MISA Workshop #4

Github

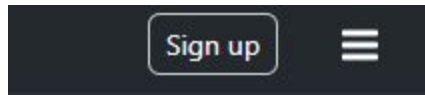
What we will do in this workshop:

- **Create a GitHub account**
- **Create a Repository**
- **Fork some Code!**
- **Edit that Code!**
- **Branch the Code!**
- **Setup a GitHub.io landing page**

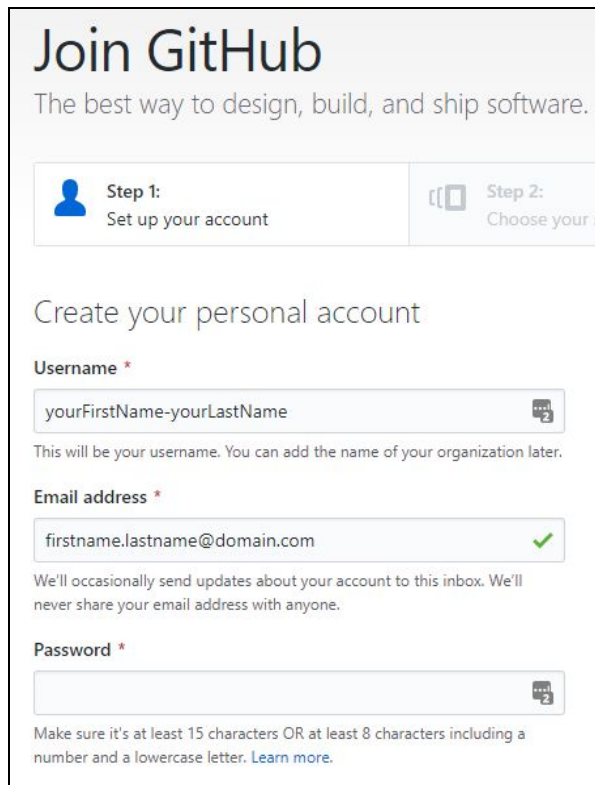
Create a GitHub account

1. Go to <https://github.com/>

2. Click "Sign up"

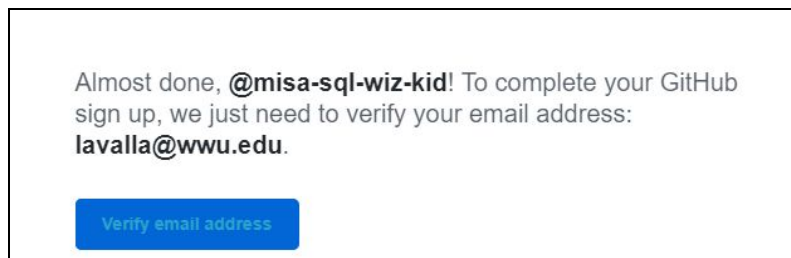


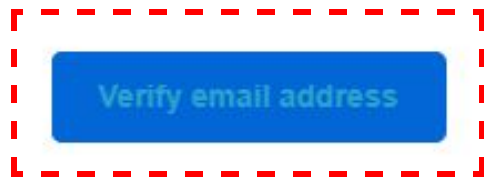
3. Enter a **Username**, **Email**, and **Password**

A screenshot of the GitHub "Join GitHub" page. The page has a light gray header with the text "Join GitHub" and "The best way to design, build, and ship software." Below the header, there are two steps: "Step 1: Set up your account" (with a person icon) and "Step 2: Choose your settings" (with a gear icon). The main section is titled "Create your personal account" and contains three form fields: "Username *" with the placeholder "yourFirstName-yourLastName", "Email address *" with the placeholder "firstname.lastname@domain.com" and a green checkmark, and "Password *" with a placeholder and a strength indicator. Below the password field, there is a note: "Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)"

4. Verify your email address:

- Go to your email and find the verification email from GitHub
- Click "**Verify email address**"





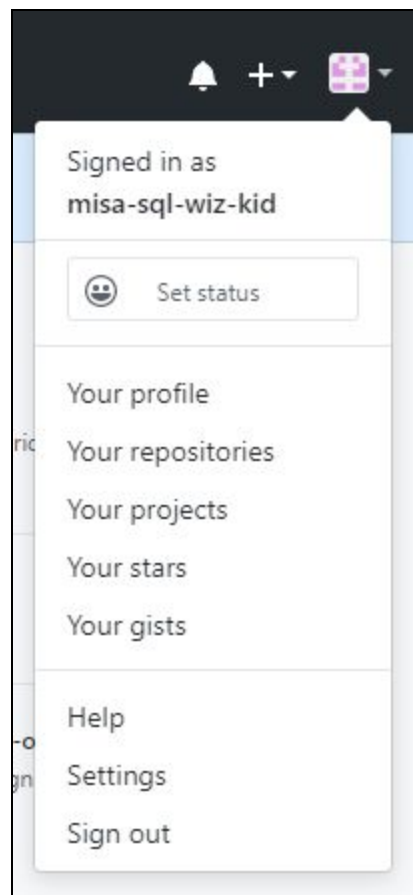
5. Congrats you made a Github account!

Create a Repository

1. Click the Profile icon in the top right corner of the browser window




A drop down menu will appear:



Click on **"Your repositories"**





2. Click "New"



Type: All ▼

Language: All ▼






Set status

misa-sql-wiz-kid

Edit profile

Overview Repositories 0 Projects 0 Stars 0 Followers 0 Following 0

Find a repository... Type: All ▼ Language: All ▼ 

misa-sql-wiz-kid doesn't have any public repositories yet.

3. Enter a **Repository name** using this format:

[yourUsername].github.io


- Make sure your repository is set to **"Public"**
- Check the checkbox for **"Initialize this repository with a README"**
- Click **"Create repository"**

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner

Repository name *


 misa-sql-wiz-kid ▾

 /


misa-sql-wiz-kid.github.io ✓

Great repository names are short and memorable. Need inspiration? How about [urban-octo-happiness?](#)

Description (optional)

☒  **Public**

Anyone can see this repository. You choose who can commit.


☐  **Private**

You choose who can see and commit to this repository.

☒ **Initialize this repository with a README**

This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** ▾

Add a license: **None** ▾ 

Create repository

4. Open a new tab in your browser and in the URL put:

https://[yourUsername].github.io

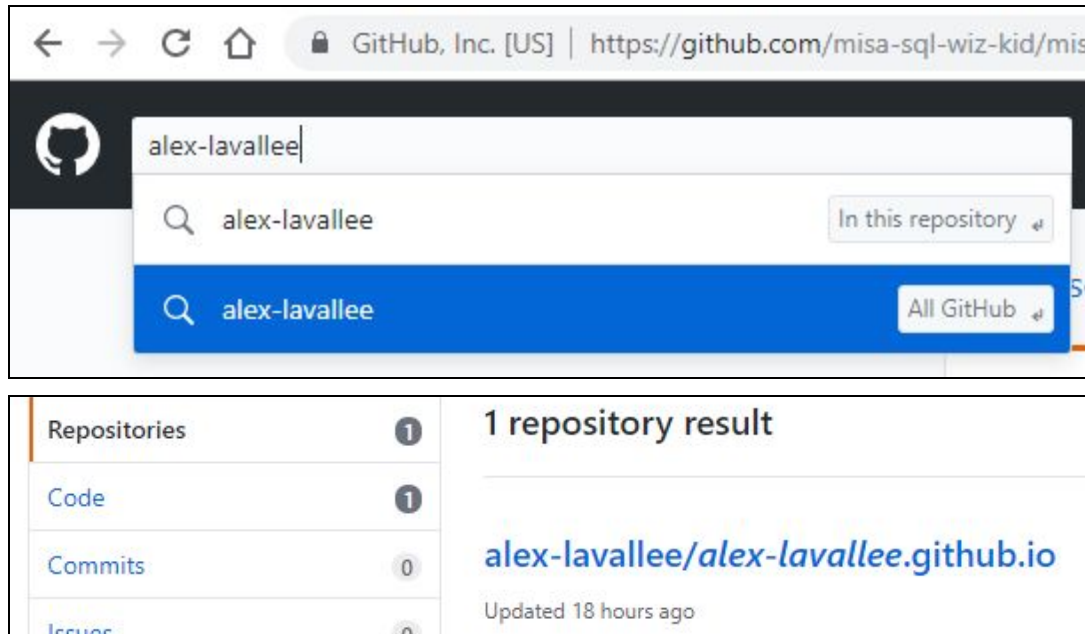
Example:



5. Congrats you created a GitHub repository!

Fork Some Code!

1. Search for "alex-lavallee" in the GitHub search bar.



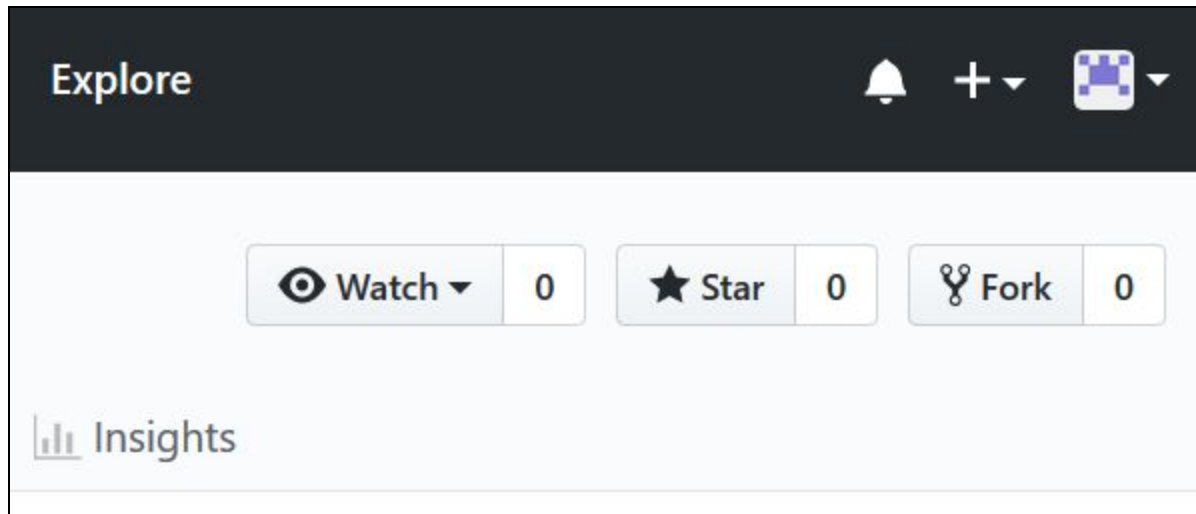
or

Paste the URL:

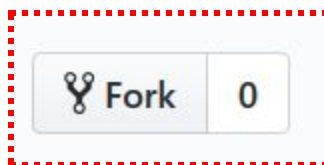
<https://github.com/alex-lavallee/alex-lavallee.github.io>

2. Fork the repository!

In the top right corner of the repository page, find the **"Fork"** button

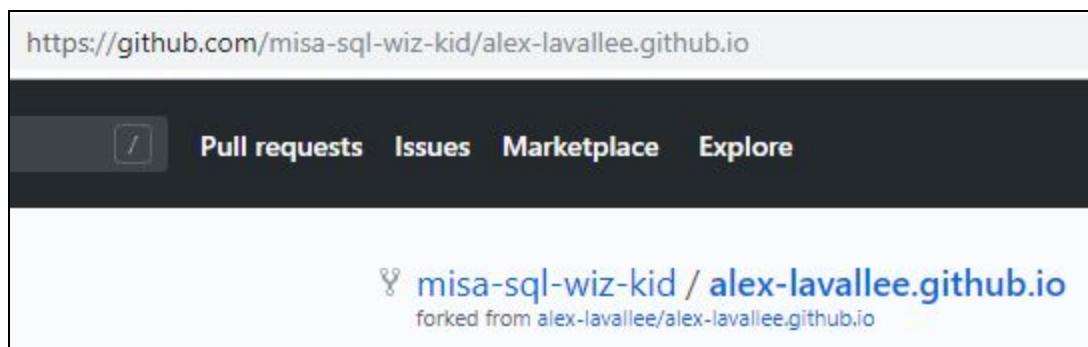


Click **"Fork"**



You should be looking at a replica of the repository you just forked.

This replica can be changed without impacting the original repository and its files.



3. Congrats, you forked some code!

Edit That Code!

Now, in our newly forked repository, let's edit some code within the file "index.html"

1. Click on "index.html"



2. We are looking at the contents of the file.

On line 5, there is an html <title> tag that contains the string "My Awesome Name!!!"
We are going to replace this placeholder with your name!

```
1 <!doctype html>
2 <head>
3
4 <!-- Title -->
5 <title>My Awesome Name!!!</title>
6
7 <!-- Styles -->
8 <link rel="stylesheet" href="styles.css">
```

To the right of the code there is a small row of buttons



Raw: Open the raw file content in a new tab

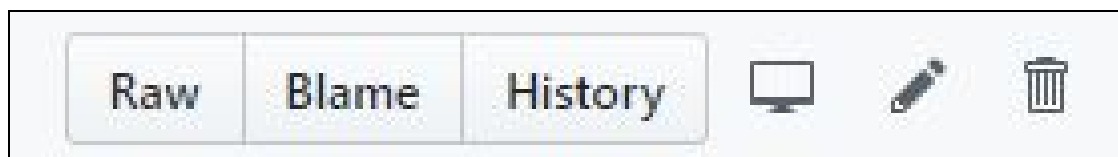
Blame: Review code changes, line by line

History: Review the edit history

Desktop: Open the file in Desktop version of GitHub

Edit: Edit the file

Delete: Delete the file



OK, let's replace "My Awesome Name!!!" with your name!

3. Click the "Edit button"



We are presented with a in-browser text editor

Change the sting in the <title> tag to your name (or whatever you want really)

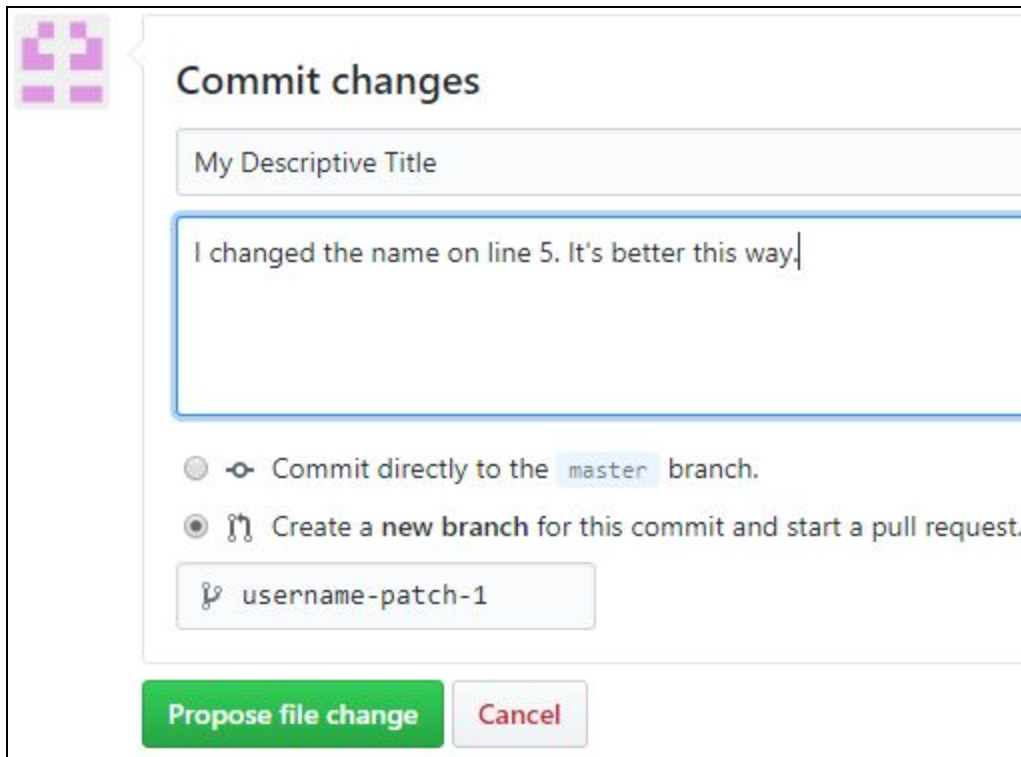
Branch the code!

4. Once you've done that, scroll to the bottom of the web page to the **Commit changes** section

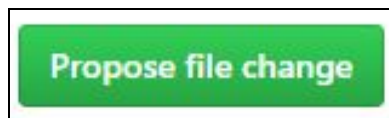
Enter a descriptive title for your changes and additional details
(If left empty, the field will be filled with a default generic title)

Instead of committing to the master branch, select the option to **"Create a new branch for this commit and start a pull request."**

Give the branch a descriptive name

A screenshot of a 'Commit changes' dialog box. On the left is a purple icon. The title 'Commit changes' is at the top. Below it is a text input field containing 'My Descriptive Title'. Underneath is a larger text area with the text 'I changed the name on line 5. It's better this way'. Below the text area are two radio button options. The first option is 'Commit directly to the master branch.' with an unchecked radio button. The second option is 'Create a new branch for this commit and start a pull request.' with a checked radio button. Below the second option is a text input field containing 'username-patch-1'. At the bottom are two buttons: a green 'Propose file change' button and a grey 'Cancel' button.

Click **"Propose file change"**



We are taken to a page that will allow us to **Open a pull request**

Open a pull request

The change you just made was written to a new branch named `username-patch-1`. Create a pull request below to propose these changes.

base: master

compare: username-patch-1

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

My Descriptive Title

Write Preview

AA B i “ < > @

I changed the name on line 5. It's better this way.

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

Create pull request

Reviewers

No reviews

Assignees

No one—assign yourself

Labels

None yet

Projects

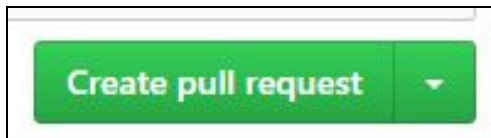
None yet

Milestone

No milestone

The page has lots of cool information but for our purposes we only need to do one thing.

Click **"Create pull request"**



Now the originator of the file, alex-lavallee, has received your pull request.

Let's check out how it looks to have a bunch of pull requests.

Pause here.

Setup a GitHub.io landing page

Download and extract the code.zip file

Go to the repository we made earlier

Click "Upload files"

Choose these files from the zip folder:

- index.html
- Styles.css
- myPhoto.jpg

In the **Commit changes** section, leave the default option (**Commit directly to the master branch**) selected

Click "**Commit changes**"

You should be back on the repository's page.

The URL should look like this:

<https://github.com/misa-sql-wiz-kid/misa-sql-wiz-kid.github.io>

Copy **the highlighted portion** and in a new tab, paste it in the URL bar

Misa-sql-wiz-kid.github.io

6. Congrats! You setup your very own GitHub.io landing page!

If you have picked a professional enough sounding username, you can put the URL on resumes or in email signatures (or wherever!)