GitHub Setup

STEP 1: Create a GitHub account

1. Go to github.com
2. Click sign up (upper right-hand corner)
3. Create your account (see below)
   a. Consider using a professional\formal username - NOT something silly like doggernation or jellyfish15.
   b. You can use your uiowa.edu email address.
   c. Below is a sample of what it will look like.

Join GitHub

First, let's create your user account

Username *

Email address *

Password *

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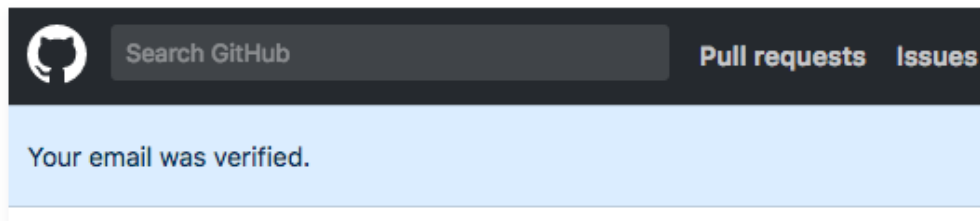Learn more.

Email preferences

☑ Send me occasional product updates, announcements, and offers.

Verify your account
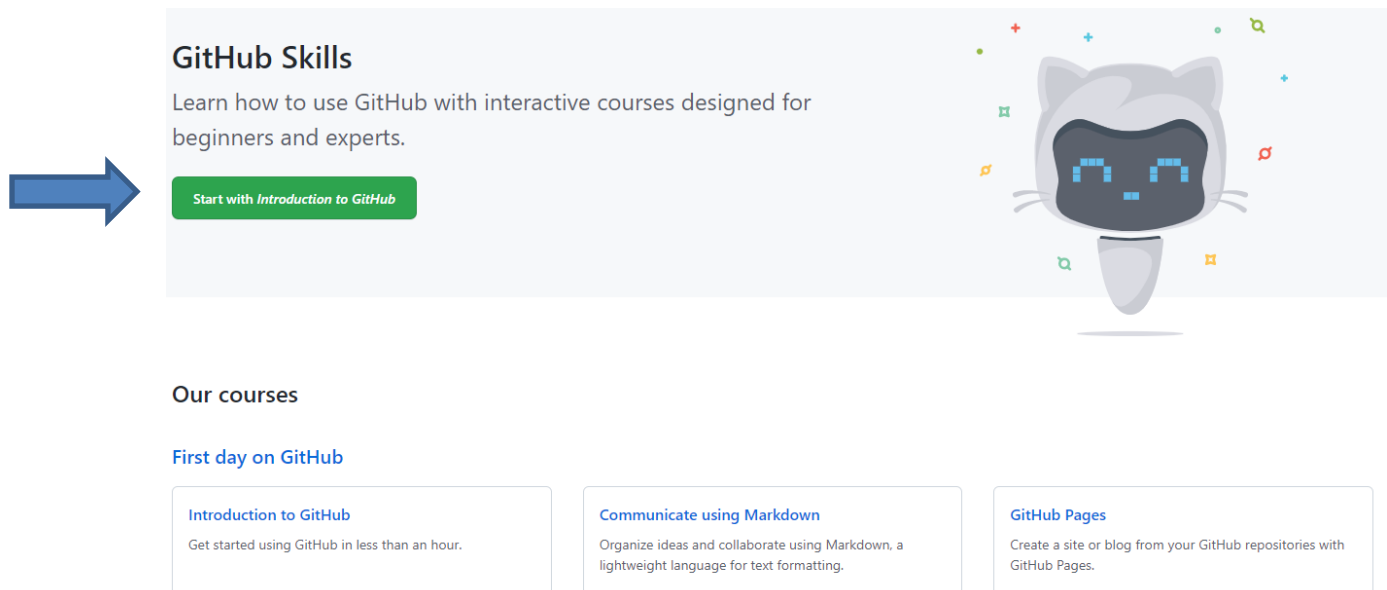
Please solve this puzzle so we know you

4. Next you will be asked to either enter a launch code or verify your email address. Go to your email and look for an email from GitHub. Either enter the code or click the "verify email address" link.
5. Once your email is verified you will get this screen.

Search GitHub   Pull requests   Issues

Your email was verified.

6. YOU DID IT! You are officially ready to log into GitHub.

STEP 2: GitHub Skills Lab (this step takes ~45 min – 1 hr)

1. Next, take some time to learn a little about GitHub. We hear from industry professionals that having some GitHub experience is a highly sought after skill. It can really set you apart from other candidates.
2. You may be directed to the GitHub Skills site, if not please go to this page: https://skills.github.com/.
3. From here, you can go through the process of interactive learning by following any of the beginner courses.

4. Select the "Start with *Introduction to GitHub*" course.
5. After the module is loaded, follow the interactive steps by clicking "Use this template".
6. NOTE: You often have to refresh the page for it to update from step to step.
7. Once you are finished with the course, you should have practiced creating a new branch, adding a file and merging sub-branch back into the main branch.

STEP 3: Create a Hello World Repository and Customize the Readme.md file (this step takes ~30 min)

1. From your GitHub home page, Click on Repositories.
2. Click New
3. Add a name for the repository – "Hello-World"
4. Select "Initialize this repository with a README"
5. Select "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.

Repository template

Start your repository with a template repository's contents.

No template ▾

Owner *   **Repository name** *

kaarens ▾   /   Hello-World   ✓

Great repository names are short and memorable. Need inspiration? How about **expert-adventure**?

Description (optional)

◉ Public
Anyone on the internet can see this repository. You choose who can commit.

○ Private
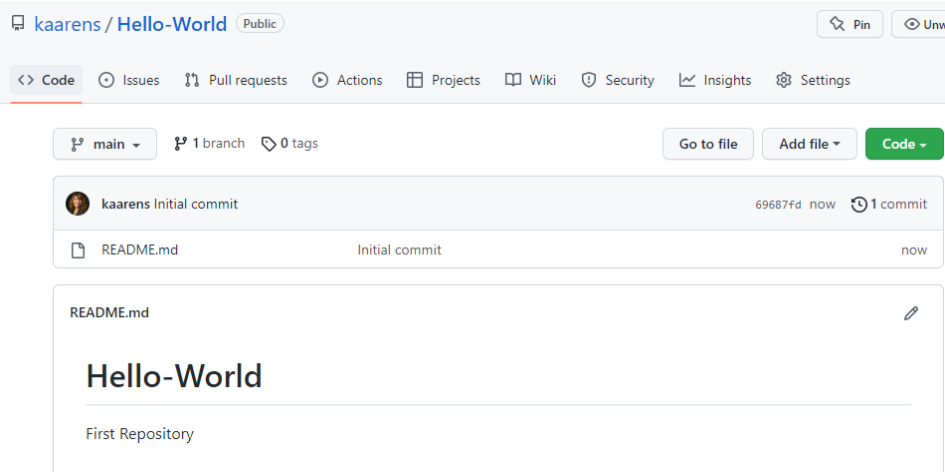You choose who can see and commit to this repository.

Initialize this repository with:
Skip this step if you're importing an existing repository.

☑ **Add a README file**
This is where you can write a long description for your project. Learn more.

6. When you are complete, you will have a repository that looks similar to the below example.

kaarens / **Hello-World** Public

<> Code   ⊙ Issues   ⁂ Pull requests   ⊙ Actions   ⊞ Projects   ▯ Wiki   ⊘ Security   ⋌ Insights   ⚙ Settings

⬍ main ▾   ⑂ 1 branch   ⬨ 0 tags

Go to file   Add file ▾   Code ▾

kaarens Initial commit   69687fd now   ⊙ 1 commit

▯ README.md   Initial commit   now

README.md   ✎

Hello-World

First Repository

STEP 4: Create a sample README.md file

1. Create a useful README.md file in your repository so users can quickly and easily understand the details of your repository.
2. To create an attractive README.md file, you will want to use the Markdown syntax. A quick guide to use Markdown can be found here: https://www.markdownguide.org/cheat-sheet/ or here: https://help.github.com/en/github/writing-on-github/basic-writing-and-formatting-syntax
3. Edit the Hello-World README.md file and create customized section headings and use **AT LEAST 4 different syntax options** (for example, bold, bullet points, headings, links, etc). To create a standard look and feel, you could use the following headings:
   a. Project Title
   b. Description
   c. How to Run Program
   d. Files Used
   e. Additional Documentation
   f. Versioning

   You can look at my GitHub for an example, but try out different heading types, fonts, text styling, lists, and so on. I **DO NOT** want you to just copy my file. You can even customize the headings a little to fit your style. https://github.com/kristinearens/Hello_World.

STEP 5: Upload your picture and complete your profile

1. It's time to create your profile. Up in the right hand corner, select your picture and then "Your Profile"
2. Click on the "Edit profile" button
3. Add your name and write a quick bio. You are welcome to update other profile preferences such as location. But for this exercise I would like you to at least add your name and a bio.

Name

[Name]

Bio

[Add a bio]

You can **@mention** other users and organizations to link to them.

[🏢] [Company]

[⊙] [Location]

[🔗] [Website]

[🐦] [Twitter username]

[**Save**] [Cancel]

4. Add a profile picture by clicking on the avatar and then click Edit next to the photo.

Public profile

Name

[Kristine Arens]

Your name may appear around GitHub where you contribute or are mentioned. You can remove it at any time.

Public email

[Select a verified email to display ▼]

You have set your email address to private. To toggle email privacy, go to email settings and uncheck "Keep my email address private."

Bio

Professional Director for the M.S. Business Analytics program. Teach the BAIS Professional Prep and MSBA Professional Development classes

You can **@mention** other users and organizations to link to them.

URL

[   ]

Twitter username

[   ]

Profile picture

✏ Edit

5. Click the green "Update profile" button
6. YOU DID IT! You are done setting up your profile.
7. Compare your site with mine to see if you have all of the setup steps completed:
   https://github.com/kristinearens/Hello_World

STEP 6: Log into LinkedIn Learning using your HawkID

STEP 7: CHOOSE: Watch either the LinkedIn Learning session called GitHub for Data Scientists or Craft a Great GitHub Profile

STEP 8: Complete Learning GitHub Course and Earn a Certificate

You finish the LinkedIn Learning course, you have an opportunity to gain a certificate. This certificate is a great way to showcase your skills. You can add this certificate to your resume and your LinkedIn profile.

STEP 9: Submit GitHub Assignment

Download your certificate and submit your certificate and enter your personalized GitHub URL as a comment to demonstrate you completed this work.