

GitHub Setup

STEP 1: Log into [LinkedIn Learning using your HawkID](#)

1. Under Overview - Website on the right-hand side, click on “Log into LinkedIn Learning”
2. Enter your HawkID and password

STEP 2: CHOOSE: Watch either the LinkedIn Learning session called [GitHub for Data Scientists](#) or [Craft a Great GitHub Profile](#)

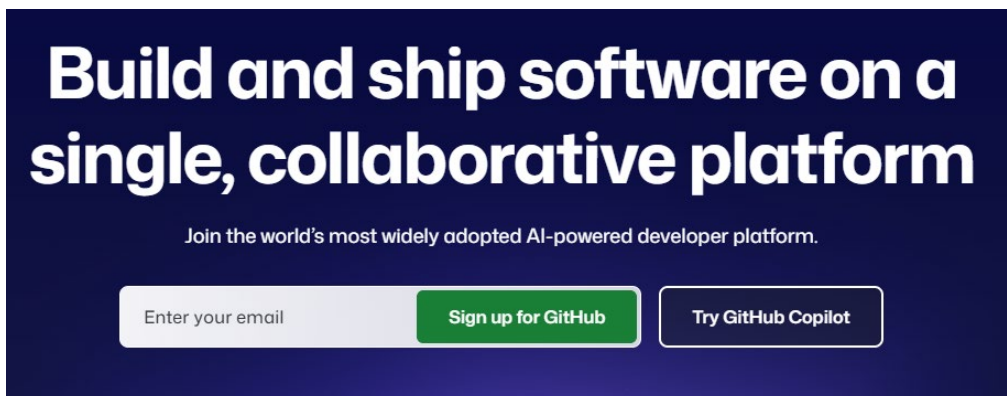
STEP 3: Complete Learning GitHub Course and Earn a Certificate

Once you have completed watching the entire 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.

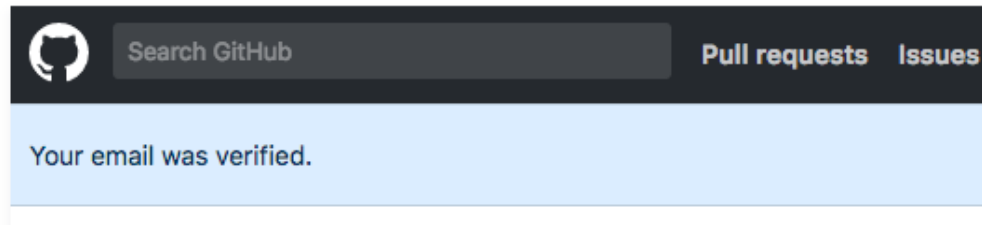
STEP 4: Create a GitHub account

For the next several steps, you can follow my instructions below or if you would like additional details, you can follow the instructions shared by GitHub (<https://github.blog/developer-skills/github/beginners-guide-to-github-setting-up-and-securing-your-profile/>)

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 or your personal email address if you would like to keep using GitHub beyond school.
 - c. Below is a sample of what it will look like.



4. Next you will be asked to Sign up for GitHub. Enter your email and password. Click the Create Account button.
5. Look for an email from GitHub. Either enter the code or click the “verify email address” link.
6. Once your email is verified you will get this screen.

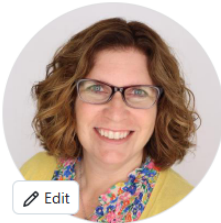


7. YOU DID IT! You are officially ready to log into GitHub.

STEP 5: Upload your picture and complete your profile

1. It's time to create your profile. Follow the steps located in this GitHub guide: [Personalizing Your Profile](#)
2. I would like to see you update the following:
 - a. Add your name
 - b. Create a short bio
 - c. Upload a picture – This can be a picture of you or something else.
3. Once you are done, it should look something like this:

Public profile

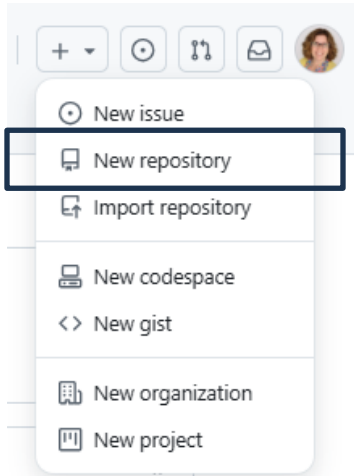
<p>Name</p> <input type="text" value="Kristine Arens"/> <p>Your name may appear around GitHub where you contribute or are mentioned. You can remove it at any time.</p> <p>Public email</p> <input type="text" value="Select a verified email to display"/> <p>You have set your email address to private. To toggle email privacy, go to email settings and uncheck "Keep my email address private."</p> <p>Bio</p> <input type="text" value="Associate Professor of Practice for the Business Analytics department at The University of Iowa"/> <p>You can @mention other users and organizations to link to them.</p> <p>Pronouns</p> <input type="text" value="Don't specify"/>	<p>Profile picture</p>  <p>Edit</p>
---	--

URL

4. Scroll down and click the green "Update profile" button
5. YOU DID IT! You are done setting up your profile.
6. Compare your site with mine to see if you have all the setup steps completed:
<https://github.com/kristinearens>

STEP 6: Create a Hello World Repository and Customize the Readme.md file (this step takes ~30 min)

1. Follow the instructions on [this site](#) to practice creating your first repository.
2. You will complete the following steps:
 - a. Select the + and click on New Repository



3. On the create screen,
 - a. Add **Hello-World** as your repository name
 - b. Add a description
 - c. Make sure this is a public repository
 - d. Turn on the **Add README**
 - e. Click on **Create Repository**

A screenshot of the GitHub repository creation form. The form is divided into two sections: General and Configuration. In the General section, the Owner is set to 'kristinearens' and the Repository name is 'Hello-World!'. A blue arrow points to the Owner field. In the Configuration section, the visibility is set to 'Public', the 'Add README' toggle is turned on, the '.gitignore' file is set to 'No .gitignore', and the license is set to 'No license'. Blue arrows point to the 'Choose visibility' dropdown, the 'Add README' toggle, and the 'Add license' dropdown. At the bottom right, there is a green 'Create repository' button.

1 General

Owner * / Repository name *

kristinearens / Hello-World!

✓ Your new repository will be created as Hello-World-.
The repository name can only contain ASCII letters, digits, and the characters ., -, and _.

Great repository names are short and memorable. How about [ubiquitous-octo-goggles?](#)

Description

First Repository

16 / 350 characters

2 Configuration

Choose visibility * Choose who can see and commit to this repository Public

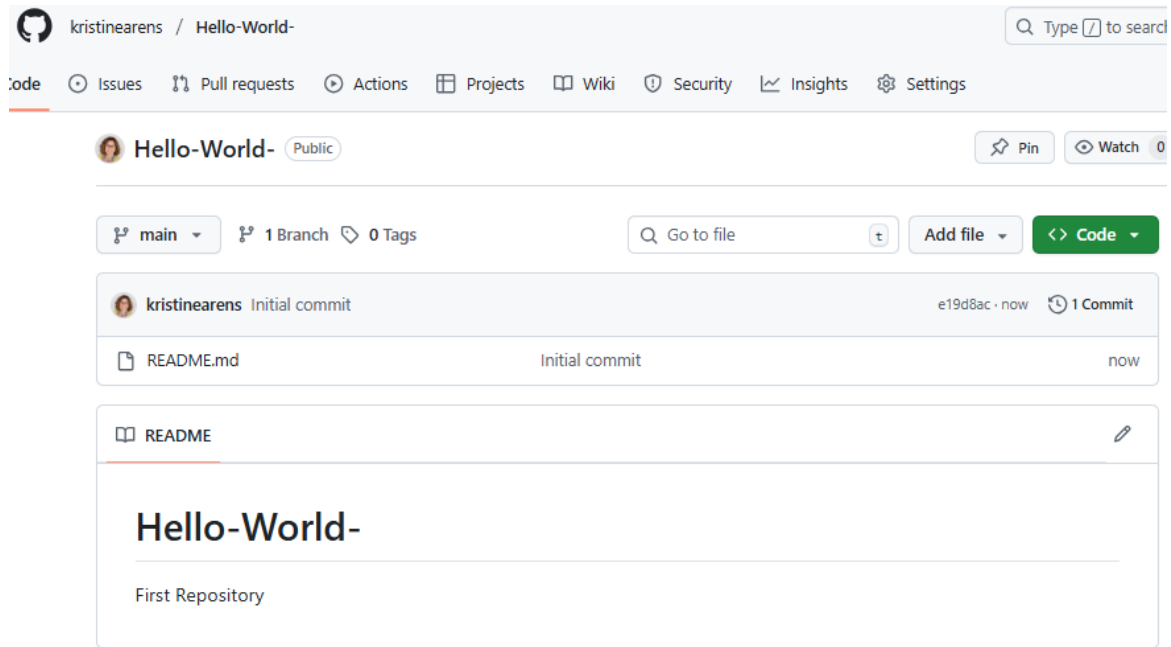
Add README READMEs can be used as longer descriptions. [About READMEs](#) On

Add .gitignore .gitignore tells git which files not to track. [About ignoring files](#) No .gitignore

Add license Licenses explain how others can use your code. [About licenses](#) No license

Create repository

- When you are complete, you will have a repository that looks like the below example.



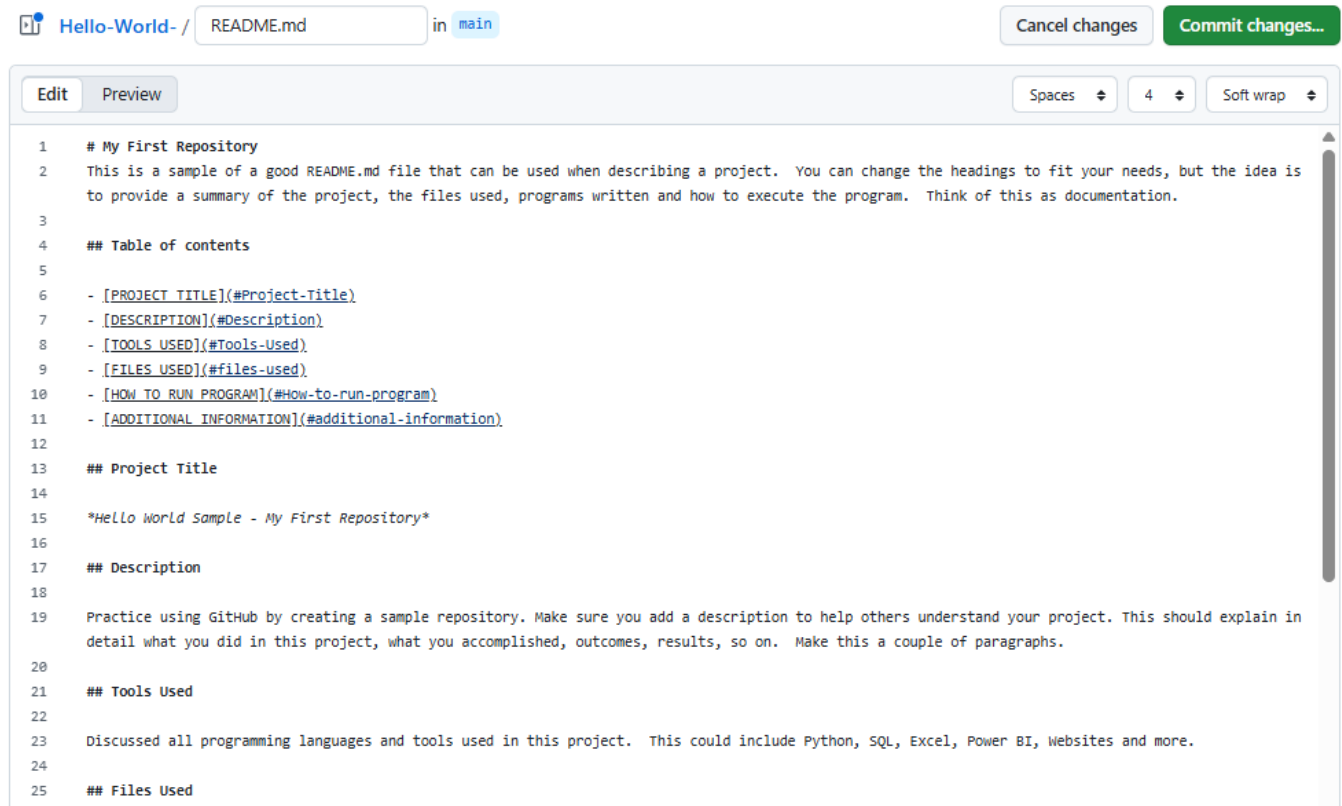
STEP 7: Update the README.md file

- Click on the pencil icon to edit the README.md file.



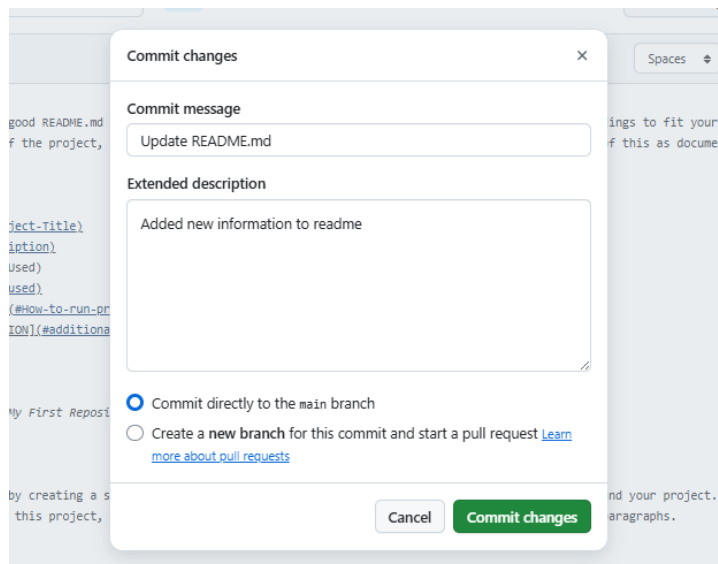
- Update the README.md file using the Markdown syntax.
- In your README.md file, you will add the following descriptions:
 - Project Title
 - Description
 - Tools Used
 - Files Used
 - How to Run Program
 - Additional Information
- You will use a formatting syntax called Markdown. A quick guide to use Markdown can be found here: <https://docs.github.com/en/get-started/writing-on-github/getting-started-with-writing-and-formatting-on-github/basic-writing-and-formatting-syntax>
- Below is an example of how you can create your README.md file.
- Once completed, you will **Commit Changes**

BAIS:3050 BAIS Professional Preparation



The screenshot shows a GitHub repository editor for a repository named "Hello-World-". The file being edited is "README.md" on the "main" branch. The editor has tabs for "Edit" and "Preview". The content of the README.md file is as follows:

```
1 # My First Repository
2 This is a sample of a good README.md file that can be used when describing a project. You can change the headings to fit your needs, but the idea is
  to provide a summary of the project, the files used, programs written and how to execute the program. Think of this as documentation.
3
4 ## Table of contents
5
6 - [PROJECT TITLE](#Project-Title)
7 - [DESCRIPTION](#Description)
8 - [TOOLS USED](#Tools-Used)
9 - [FILES USED](#files-used)
10 - [HOW TO RUN PROGRAM](#How-to-run-program)
11 - [ADDITIONAL INFORMATION](#additional-information)
12
13 ## Project Title
14
15 *Hello World Sample - My First Repository*
16
17 ## Description
18
19 Practice using GitHub by creating a sample repository. Make sure you add a description to help others understand your project. This should explain in
  detail what you did in this project, what you accomplished, outcomes, results, so on. Make this a couple of paragraphs.
20
21 ## Tools Used
22
23 Discussed all programming languages and tools used in this project. This could include Python, SQL, Excel, Power BI, Websites and more.
24
25 ## Files Used
```



The screenshot shows the "Commit changes" dialog box in GitHub. The dialog has a title bar with a close button (X). It contains the following fields and options:

- Commit message:** A text input field with the value "Update README.md".
- Extended description:** A text area with the value "Added new information to readme".
- Commit options:** Two radio buttons. The first is selected and labeled "Commit directly to the main branch". The second is labeled "Create a new branch for this commit and start a pull request" with a link to "Learn more about pull requests".
- Buttons:** "Cancel" and "Commit changes".

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.

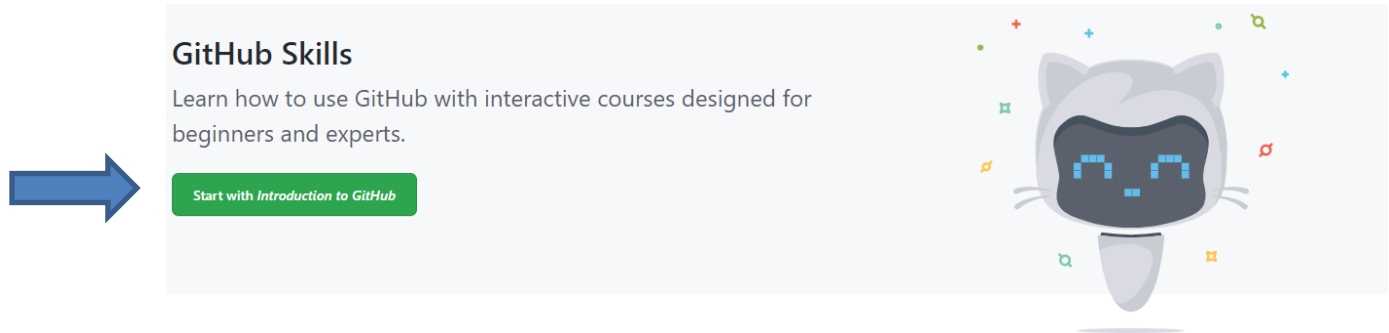
https://github.com/kristinearens/Hello_World.

**** Here are some examples of good README files by other GitHub users:**

<https://github.com/abhisheknaidu/awesome-github-profile-readme>

OPTIONAL STEP 8: 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 9: Submit GitHub Assignment

Follow the instructions from the assignment on how to submit your certificate and your GitHub information.