Hi, this is Jess!

As part of your GitHub onboarding, today I am going to give you a tour of the NDCLab wiki and its associated GitHub repository. We are going to go through all of the steps necessary to add your picture and your personal information to the lab Who’s Who page. This process involves all of the key commands that you have been learning about in the Get with Git program.

Here is our agenda.

Then, we’re going to create a pull request. This is a request for someone else in the lab to “pull” down your branch to their local machine and review your changes. Think of a pull request as asking for someone’s opinion or their approval.

Finally, we’re going to wrap up with your next mission and a note about GitHub notifications.

Ok, let’s get started.

This is the NDCLab wiki. In the left-hand navigation, we have several higher-level folders. Today, we’ll be working in the Welcome folder. There are four subfiles in the Welcome folder. Our target is this one: Who’s Who. So, here, you are seeing the wiki rendered as a webpage.

But this webpage is super cool because all of its content comes from a single repository on GitHub.

This is the lab’s GitHub page, which contains all of the lab’s repositories. These are tools, projects, and resources. The wiki is one of those repositories.

When we were viewing the live wiki, we were seeing the content of the main branch. Under docs, we have the same folders that we saw in the left-hand navigation of the web wiki. So we are going to go to welcome, and Who’s Who. Same content. This is how GitHub renders the markdown operating behind the scenes, which we can see by clicking Raw. In a moment, we are going to open a text file that looks exactly like this.

First, let’s prepare our own content.

I have selected this picture to upload to the wiki. But I need it to be 100x100 pixels. And I don’t really want all that background. Since I’m on a Mac, I’m going to use Preview to crop and resize. Now I rename the picture with my last name. Ok, picture is ready!

I also need to prepare this content. To save time, I’ve drafted my research interests and hobbies already.

It’s time to clone this repository! Here I have opened my shell. Don’t worry that it doesn’t look like yours, it operates the same way. I just happened to customize mine awhile ago. To clone the lab’s wiki repository, we need this link that points GitHub to the exact repo we want to clone. Right now, I’m down in the wiki documents, so I’m going to jump back up to the main code page of the wiki repository. I can get the link that lets me clone this repo here.

Now, in my shell, I need to navigate to the folder where I want to create this clone. Right now, I’m out at the root level. I created a “github” folder to keep all my repositories organized. So I’m going to move into that folder. I like to check myself by using the “ls” command a lot. Here, we can see that I already have a few other repos in this folder.

We use the command “git clone”, together with that link we got from GitHub, to clone the wiki repository right here on my local machine. Next, it’s very important that I create my own branch, which we name with the feature that is being added and the name of the person adding it.

Next, we must checkout that branch. This ensures that all the changes we are making are housed on this special branch, our own personal playground. It’s a good practice to use the “git status” command to double check yourself. When we execute that command, the shell tells us that we are on the right branch.

Now, let’s go checkout the folder that we’ve created. We need to do two things:

--First, we need to add our picture to the “assets” folder. I’ve copied this with command-C (control-C on PC) and I’m going to paste it here.

--Second, we need to update the “whos-who.md” file that contains the written content on the page. An .md file is like a text file; the MD stands for “markdown.” When I open this .md file, I can simply copy the format from an existing entry, in this case George’s, and paste it into a new location for myself. Now I just overwrite the information with my own. Make sure to update the image link so that it points to your image! Because my image is in the same folder as George’s, I just need to provide the correct name. (If you are using a PNG file instead of a JPG, be sure to change the extension in the link, too.)

Save the file, and close it.

Ok, the wiki repository on my local machine is all updated. It’s time commit these changes for posterity and push them up to the lab’s remote repository so it is accessible to everyone else.

To do that, I need to put my changes into the staging area. We do this with “git add”. Remember to include both the whos-who.md file, which we modified, and the new image, which we added.

Let’s do a quick check-in with git status. Great, we can see that git has both files ready to commit with changes. We’ll commit those with a pithy, informative message.

With the commit successful, we can push our local branch back to the remote.

Let’s go check it out on GitHub. Let me refresh this page. Ok, from this dropdown, we can see all active branches. Look, here’s our new one. When we navigate to the whos-who page, we can see our changes. But these aren’t “live” yet, but they aren’t on the main branch. We need to do a pull request so someone else checks our work and merges our new branch with main.

Go to Pull Requests and click to create a new one. You’re asking to merge your named branch into main. Click Create. In general, the lab manager will be the one reviewing pull requests for the whos-who page, so please tag appropriately.

This is a great time to show you how notifications work inside GitHub. Whenever someone @mentions you, you will receive a notification, which you can view by clicking this bell. You should also receive an e-mail. Be sure to watch out for your notifications so that you know if someone in the lab is requesting your input on an issue. You can clear an issue just by clicking the checkmark.

You are now, officially, onboarded to GitHub! Once you follow these steps and complete your first pull request to add your information to the whos-who page, you still have another mission to complete. The wiki a living document, a way of sharing knowledge across lab members and across time. Please explore the wiki and identify something that can be improved. Create a new branch, checkout to that branch, make your suggested changes, and initiate another pull request. Thanks in advance for contributing to the wiki!