Overview

GitHub is a site for hosting files (usually code) and their history. I'll be using it to share course materials with you; you'll be using it as a host for the websites you build, for collaboration on group projects, and to turn in assignments.

OrionHub is a site that adds a code editor on top of GitHub; the editor will color your code to make it easier to read, autocomplete to save you typing, and detect and highlight common coding mistakes, much like a spellchecker catches errors in text documents.

Some terms that you'll see around both sites, which are worth knowing:

- **Repository:** a set of files and their history
- **Clone:** to copy a repository
- **Fork:** to clone and also notify the source repository of the cloning
- **Commit:** a set of contemporaneous changes and their description; to group changes into a commit and add them to a repository
- **Push:** to send changes from one repository to another
- **Pull:** to receive changes from one repository into another
- **Merge:** a combining of changes and histories from two repositories
- **Conflict:** a case where a merge is ambiguous and needs human input

For this assignment we'll be using three repositories. Following standard naming convention, the "upstream" repository is the one I'll be providing for you, "origin" is the copy you'll create on GitHub, and "local" is the copy you'll have on OrionHub.

A Reminder

Remember that much of your activity on these sites will be publicly logged and easily associated with you and NWU. It is also mirrored by third parties, meaning it may be accessible to prospective employers, even if you have since deleted your account. Please act professionally.

Instructions, Part 1 (due September 4)

A caveat: OrionHub sometimes changes their user interface. If you find that some of these instructions no longer apply, please let me know.

1. Create your accounts.
   (a) Go to `https://github.com/`.
   (b) Enter credentials for a new account and click "Sign up for GitHub".
   (c) Choose the free option.
   (d) Go to `https://orionhub.org/`.
   (e) Click on the "Create a new account" button.
   (f) Enter credentials for a new account and click "Sign up".

2. Configure your accounts.
   (a) While logged in to OrionHub, click on the gear button in the left toolbar. If you can't see the toolbar, click on the hamburger (three-line icon) in the upper left. If you can't see the hamburger, press Ctrl-Shift-M.
   (b) Under the "Git" tab, enter whichever e-mail and username you want to appear on your commits.
   (c) Click the associated update button.
   (d) Under the "Git" tab, check the "Enable Storage" box.
   (e) Click the associated update button.
3. Practice forking a repository from the class (upstream) to your GitHub account (origin).
   (a) While logged in to GitHub, go to `https://github.com/nwu-computer-science-1000/worksheet-1`.
   (b) Click on the "Fork" button in the upper right corner.
4. Practice cloning a repository from your GitHub account (origin) to your OrionHub account (local).
   (a) While logged in to GitHub, click on your username in the upper right corner.
   (b) Click on the "Repositories" tab just below the upper toolbar.
   (c) Click on "worksheet-1" in your repository list.
   (d) Find, near the bottom of the right toolbar, a text box labeled "clone URL".
   (e) Make sure that the label reads "HTTPS clone URL". If not, click on the "HTTPS" link below the text box.
   (f) Copy the URL in the text box.
   (g) While logged in to OrionHub, click on the pencil button in the left toolbar.
   (h) From the menu, choose "File→New→Project→Git Repository.
   (i) Paste the URL you copied from GitHub.
   (j) Click on the "Submit" button.
5. Practice making an edit.
   (a) While logged in to GitHub, click on your username in the upper right corner.
   (b) Copy the URL in your browser's address bar.
   (c) While logged in to OrionHub, click on the pencil button in the left toolbar.
   (d) Under "worksheet-1", find "`Worksheet1.md`".
   (e) Find your name and, after the colon, paste the URL to your GitHub user page.
   (f) After the colon on the following line, write a few sentences about what you hope to get out of the course.
   (g) After the colon on the line below that, list some of the kinds of websites that you would be interested in building.
   (h) Press Ctrl-S to save the file.
6. Practice making a commit.
   (a) While logged in to OrionHub, click on the road-sign button in the left toolbar.
   (b) If the "worksheet-1" repository is not already selected, click on its name.
   (c) Check the box for "`Worksheet1.md`" in the "Changed Files" tab.

(d) Type a commit message describing your changes, something along the lines of "Added myself to the GitHub roster."

(e) Click on the "Commit" button in the upper right corner of the pane.

(f) Enter your name and e-mail under both committer and author.

(g) Check "Remember my committer name and email"; you shouldn't have to enter this information again.

(h) Click "Ok".

7. Practice pushing commits from OrionHub (local) to GitHub (origin).

   (a) While logged in to OrionHub, click on the road-sign button in the left toolbar.

   (b) If the "worksheet-1" repository is not already selected, click on its name.

   (c) Click on the "Push" button (not its dropdown) in the upper right corner of the "OUTGOING" subpane, which is in the "Commits" pane.

   (d) If prompted, enter your GitHub username and password. If you wish, you can check "Don't prompt me again".

   (e) Click on the "Submit" button.

8. Practice sending a pull request from your GitHub account (origin) to the class repository (upstream).

   (a) While logged in to GitHub, click on your username in the upper right corner.

   (b) Click on the "Repositories" tab just below the upper toolbar.

   (c) Click on "worksheet-1" in your repository list.

   (d) Click on the green circles-and-arrows button.

   (e) Click on the green "Create pull request" button. You do not need to change the settings from their defaults.

   (f) Enter a pull request message. (It may default to your commit message.)

   (g) Click on the green "Create pull request" button.

Instructions, Part 2 (due September 23; wait until September 9 to start)

1. Create a conflict.

   (a) Open `Worksheet1.md` in OrionHub and add your name after the colon on the line marked "Conflict line".

   (b) Commit your change. (I will have made a conflicting commit from my account.)

2. Practice adding another repository (upstream) to pull from.

   (a) Go to `https://github.com/nwu-computer-science-1000/worksheet-1`.

   (b) Find, near the bottom of the right toolbar, a text box labeled "clone URL".

   (c) Make sure that the label reads "HTTPS clone URL". If not, click on the "HTTPS" link below the text box.

   (d) Copy the URL in the text box.

   (e) While logged in to OrionHub, click on the road-sign button in the left toolbar.

   (f) If the "worksheet-1" repository is not already selected, click on its name.

   (g) Open the "Branches" pane.

(h) Click on the "New Remote" button.

(i) Enter the name "upstream" and paste the URL you copied.

(j) Click on the "Submit" button.

3. Practice pulling from another repository (upstream) and merging.

   (a) While logged in to OrionHub, click on the road-sign button in the left toolbar.

   (b) If the "worksheet-1" repository is not already selected, click on its name.

   (c) Open the "Branches" pane.

   (d) Click on the cylinder-and-arrow button in the "upstream" entry.

   (e) Expand the "upstream" entry.

   (f) Click on the three-circles "Merge" button in the "upstream/master" entry. You should get a conflict warning with the conflicting file `Worksheet1.md` listed in the "Changed Files" pane.

4. Practice resolving a conflict.

   (a) Edit `Worksheet1.md` so that your name is part of the list of names you pulled from upstream. Remove any conflict markers (lines beginning with `<<<<<<<<<`, `========`, or `>>>>>>>>>`).

   (b) Save and commit your changes. Be careful to wait for the repository page to load fully before marking any checkboxes; there was an Orion bug that would lose one's work otherwise, and it might not be fixed yet. If you do run into the bug, let me know, and I'll help you fix things.

   (c) Push your changes to your GitHub repository.

   (d) Send a pull request to upstream.

5. Practice accepting a pull request.

   (a) While logged in to GitHub, go to `https://github.com/nwu-computer-science-1000/worksheet-1`.

   (b) Click on the "Pull Requests" button in the right toolbar.

   (c) Find the pull request you made and click on its title.

   (d) Scroll down and click on the green three-circles "Merge pull request" button.

   (e) Enter a pull request acceptance message.

   (f) Click on the green three-circles "Confirm merge" button.

When we get to group work, you'll follow this general pattern:

- Fork the project starter from upstream to origin (Part 1, Step 3).
- Clone the project starter from origin to local (Part 1, Step 4).
- Repeatedly:
  - Make edits and commits (Part 1, Steps 5 and 6).
  - If necessary, pull and merge your teammate's changes from origin (Part 2, Step 3, but with origin, not upstream).
  - Resolve any conflicts (Part 2, Step 4, but without sending a pull request).
  - Push commits to origin (Part 1, Step 7).

The other steps may also come up, but in rarer circumstances.