

Working with Git and GitHub

HBS workshop led by

Ista Zahn, Bob Freeman, and Melissa Valez

Roadmap

Workshop is informal, example-based, hands-on. In general I will demonstrate a technique and then give you an opportunity to practice it. Topics may include:

- Software and account setup
- Things I like about git and GitHub
- Use the GitHub web interface
- Git commits and history
- Issues and collaborators
- Websites with GitHub pages
- Forking and pull requests
- Configuring and using local Git clients
- Git / GitHub Q&A

Accounts and Software

- Pre-workshop survey: http://bit.ly/rcs_pre_class_survey.
- Sign-up / into <https://github.com> and <https://code.harvard.edu>
- Install GitHub Desktop from <https://desktop.github.com/>
- Install Git:
 - Windows: <https://gitforwindows.org/>
 - Mac: <https://sourceforge.net/projects/git-osx-installer/files/>
- Install Rstudio: <https://rstudio.com/products/rstudio/download/#download>
- Navigate to the class repository at <https://github.com/izahn/demo>

Things I like about Git(hub)

- Collaborate via issues, pull requests, discussions
- Easy web presence for your projects
- Creates a record of the work you do
- Track and compare changes to project files over time

You can do a lot with the GitHub web interface

- **Create a repository**
- Add / Edit files and track changes
- Fork an existing repository
- Make a webpage for your project
- Manage settings and collaborators
- Submit / approve / deny pull requests
- Open / close / track issues

And more! <https://lab.github.com/> is a good place to learn all about it.

Create a new repository

- Click the + sign in the upper right and select "New repository"
- Name it whatever you like
- Select "Public"
- Select "Initialize with a README"
- Click the "Create project" button
- Click the "README" link and add something to the README file.

You can do a lot with the GitHub web interface

- Create a repository
- **Add / Edit files and track changes**
- Fork an existing repository
- Make a webpage for your project
- Manage settings and collaborators
- Submit / approve / deny pull requests
- Open / close / track issues

And more! <https://lab.github.com/> is a good place to learn all about it.

Editing files on github.com

You can add, edit and commit using the web interface. In your new repository:

- Click the README file then the edit icon
- Make some changes, write a commit message, commit
- Click "Add file" => Upload, message, commit
- You can see how your project has changed over time.
- Click the "clock" icon
- Select a commit to view the changes made

Free play!

Take some time to explore the github.com website, trying out things we've discussed or other things you discover.

We will come back together at __:__

You can do a lot with the GitHub web interface

- Create a repository
- Add / Edit files and track changes
- **Fork an existing repository**
- Make a webpage for your project
- Manage settings and collaborators
- Submit / approve / deny pull requests
- Open / close / track issues

And more! <https://lab.github.com/> is a good place to learn all about it.

Forking and pull requests

GitHub has powerful collaboration features, including the famous "pull request", AKA a "PR".

- Navigate to <https://github.com/izahn/demo> and click the "Fork" icon in the upper left
- Make some changes in your fork
- Click the "Pull requests icon" and select "new pull request" then "Create pull request"
- Add a comment and click "Create pull request"

You can do a lot with the GitHub web interface

- Create a repository
- Add / Edit files and track changes
- Fork an existing repository
- **Make a webpage for your project**
- Manage settings and collaborators
- Submit / approve / deny pull requests
- Open / close / track issues

And more! <https://lab.github.com/> is a good place to learn all about it.

Websites with GitHub page

You can easily turn your repository into a web page:

- Click "settings"
- Scroll down to "GitHub Pages" and select "master branch"
- Copy the URL and navigate to it

You can do a lot with the GitHub web interface

- Create a repository
- Add / Edit files and track changes
- Fork an existing repository
- Make a webpage for your project
- **Manage settings and collaborators**
- Submit / approve / deny pull requests
- Open / close / track issues

And more! <https://lab.github.com/> is a good place to learn all about it.

Collaborators and issues

GitHub has powerful collaboration features, including the ability to add collaborators and assign issues.

- Navigate to your fork of <https://github.com/izahn/demo> and click the "Settings" icon in the upper right
- Click "Manage access" in the menu on the left, then click the green "Invite a collaborator" button and invite me ("izahn")
- Click the "Issues" button and create a new issue
- In the right-hand sidebar click "Assignees" and assign the issue to me ("izahn")

Free Play!

Suggestions:

Fork https://github.com/hbs-rcs/just_enough_unix and (optionally) make a pull request

Search GitHub for a repository you might be interested in and fork it.

Explore other features we've covered or new things you've discovered.

We will come back together at __:__

Configuring and using GitHub Desktop

- Start GitHub Desktop on your local computer
- Sign in to Github
- Clone your fork of <https://github.com/izahn/demo>
- Make changes locally (e.g., using your favorite editor or IDE) and commit using GitHub Desktop
- Push changes to GitHub using GitHub Desktop

Configuring and using GIT from your IDE

(RStudio as an example)

- Start RStudio on your local computer
- Create a new RStudio project with version control enabled
 - <https://happygitwithr.com/rstudio-git-github.html#clone-the-new-github-repository-to-your-computer-via-rstudio>
- Make changes locally and commit
 - <https://happygitwithr.com/rstudio-git-github.html#make-local-changes-save-commit>
- Push changes to GitHub (using RStudio or Git from a terminal)
 - <https://happygitwithr.com/rstudio-git-github.html#make-local-changes-save-commit>