

Lecture 6 - Working with Remotes

DSE 511

Drew Schmidt
2022-09-13

Announcements

- Nothing unresolved from last time
- Homework soon (some time next week)
 - Probably 2 parts
 - "Give the command(s) to ..."
 - A set of git tasks, published to GH
- Questions?

Today's Lecture

- Refresher: Remotes
- Credentialing
- GitHub
- Working With a GitHub Repository

Refresher: Remotes

What Is a Remote?

- A copy of your git repository on another computer, accessible via the internet
- Usually with extra tools/utilities/services built around git
- Use is not strictly necessary (but why wouldn't you?)

Example Remotes

- GitHub
- Bitbucket
- GitLab
- Run the git server yourself (don't recommend)
- ...

Why Remotes?

- Enables collaboration
- Free backups!
- Graphical (via web browser) interface to git's worst parts
 - Merging
 - Searching
- Offers
 - Bug tracking
 - Feature requests

Collaborating

- Using remotes (e.g. GitHub)
- Usual pipeline (distributed model)
 1. fork
 2. make changes
 3. create pull request (PR)
- For centralized codebases (e.g. proprietary ones), forking may be blocked
 1. branch
 2. make changes
 3. create PR

Credentialing

Credentialing

- Remotes need a way of authenticating you
- Options
 - Username/password
 - ssh keys (passwordless!)
- Setting up ssh keys is extremely helpful!

What Are ssh Keys?

- ssh: secure shell
- keys: a passwordless identification system
- key pair
 - private key: used to "sign"
 - public key: used to verify a "signature"
- Public key cryptography https://en.wikipedia.org/wiki/Public-key_cryptography

Public vs Private

- Never reveal your private key!
- Public key can be given away safely (assuming your private key doesn't leak)
- Public keys use file extension `.pub`
- Private keys use no extension

Public key

```
ssh-ed25519 publicKeyGoesHere your@email.com
```

Private key

```
-----BEGIN OPENSSH PRIVATE KEY-----  
privateKeyGoesHere  
-----END OPENSSH PRIVATE KEY-----
```

Setting Up ssh Keys

- Generate ssh key

<https://docs.github.com/en/authentication/connecting-to-github-with-ssh/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent>

- Adding ssh key to GitHub account

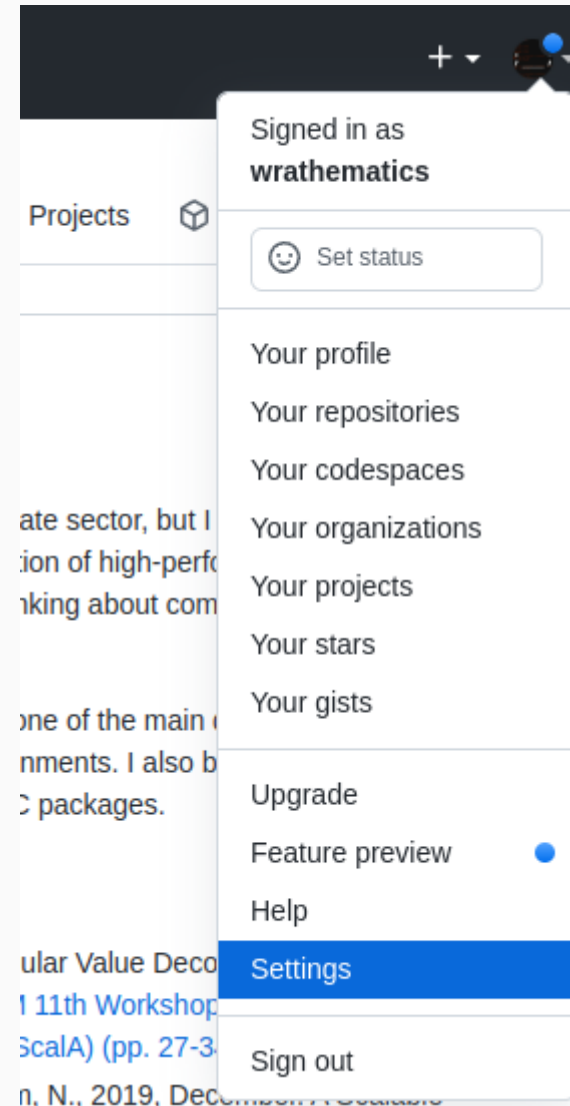
<https://docs.github.com/en/authentication/connecting-to-github-with-ssh/adding-a-new-ssh-key-to-your-github-account>

Generating

1. Open terminal
2. `ssh-keygen -t ed25519 -C "your_email@example.com"`
3. Use default file location (press enter)
4. Add a passphrase if you want
5. Add to ssh agent `ssh-add ~/.ssh/id_ed25519`

Adding to GitHub

1. Go to GitHub settings (see right)
2. Select SSH and GPG keys
3. Click New SSH key
4. Add the key
 1. Copy the contents of your **PUBLIC** key: `cat ~/.ssh/id_ed25519.pub`
 2. Paste into the key box
 3. Give it a name
 4. Click Add SSH key



GitHub

Many Features

- Public and private repos
- Issues
- Fork/PR
- Organizational accounts
- Wikis <https://docs.github.com/en/communities/documenting-your-project-with-wikis/about-wikis>
- Running a website! <https://pages.github.com/>

GitHub Pages Example

<https://fml-fam.github.io/blog/>

Features

- Feature we care about for now
 - private repos
 - issues
 - fork/PR
- We'll talk about these *next time*
- For now, GH is just a place to put stuff!

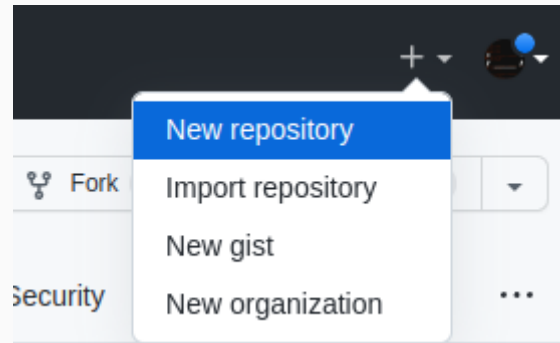
Advanced Features

GitHub is a very powerful tool with lots of advanced features we can't really get into.

- GitHub Actions (e.g. CI) <https://github.com/features/actions>
- Org secrets (e.g. integrating with AWS services)
- Project views for issue tracking (Jira-lite)

Working With a GitHub Repository

Creating a New Repository




Creating a New Repository

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner *

 wrathematics ▾

Repository name *

/ demo ✓

Great repository names are short and memorable. Need inspiration? How about [turbo-robot?](#)

Description (optional)

The demo repository used in the lectures for our CSODS course



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.](#)

Add .gitignore

Choose which files not to track from a list of templates. [Learn more.](#)

.gitignore template: None ▾

Choose a license

A license tells others what they can and can't do with your code. [Learn more.](#)

License: The Unlicense ▾

 You are creating a public repository in your personal account.

Create repository

Creating a New Repository

Putting Our Demo Repo on GitHub

- Example repo from Lecture 05 (/tmp/demo)
- Let's put it on GitHub
 - Add remote to local repo
 - Push changes

Checking Available Remotes

```
git st
```

On branch master
nothing to commit, working tree clean

```
git remote
```

```
# ssh keys  
git remote add origin git@github.com:wrathematics/demo.git  
# without ssh keys  
git remote add origin https://github.com/wrathematics/demo.git
```

Syncing

```
git push origin master
```

To github.com:wrathematics/demo.git

! [rejected] master -> master (fetch first)

error: failed to push some refs to 'github.com:wrathematics/demo.git'

hint: Updates were rejected because the remote contains work that you do

hint: not have locally. This is usually caused by another repository pushing

hint: to the same ref. You may want to first integrate the remote changes

hint: (e.g., 'git pull ...') before pushing again.

hint: See the 'Note about fast-forwards' in 'git push --help' for details.

Oh! We need to pull first



...Syncing?

```
git pull origin master
```

```
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 1.24 KiB | 1.24 MiB/s, done.
From github.com:wrathematics/demo
 * branch                master      -> FETCH_HEAD
+ b76a2cb...5db35fe master      -> origin/master (forced update)
hint: You have divergent branches and need to specify how to reconcile them.
hint: You can do so by running one of the following commands sometime before
hint: your next pull:
hint:
hint:   git config pull.rebase false  # merge (the default strategy)
hint:   git config pull.rebase true   # rebase
hint:   git config pull.ff only       # fast-forward only
hint:
hint: You can replace "git config" with "git config --global" to set a default
hint: preference for all repositories. You can also pass --rebase, --no-rebase,
```

Yes, that's a real pickle



Syncing!

```
git config pull.rebase false  
git pull --allow-unrelated-histories origin master
```

From github.com:wrathematics/demo

* branch master -> FETCH_HEAD

Merge made by the 'ort' strategy.

LICENSE | 24 ++++++

1 file changed, 24 insertions(+)

create mode 100644 LICENSE

Syncing!

```
git push origin master
```

Enumerating objects: 8, done.

Counting objects: 100% (8/8), done.

Delta compression using up to 16 threads

Compressing objects: 100% (5/5), done.

Writing objects: 100% (7/7), 742 bytes | 742.00 KiB/s, done.

Total 7 (delta 0), reused 0 (delta 0), pack-reused 0

To github.com:wrathematics/demo.git

5db35fe..b62f530 master -> master

Syncing

```
git log
```

```
commit b62f530d2563a7fa2245f510ee705a2f6e25bf51 (HEAD -> master, origin/master)
```

```
Merge: b76a2cb 5db35fe
```

```
Author: Drew Schmidt <wrathematics@gmail.com>
```

```
Date:    Sun Sep 11 17:17:08 2022 -0400
```

```
Merge branch 'master' of github.com:wrathematics/demo
```

```
commit 5db35fe0c40b3616e664eb4f41718b5c6dc9e565
```

```
Author: Drew Schmidt <wrathematics@gmail.com>
```

```
Date:    Sun Sep 11 17:13:12 2022 -0400
```

```
Initial commit
```

```
commit b76a2cba88bb731d7a5b47e34dfb15dd01580f3f
```

```
Author: Drew Schmidt <wrathematics@gmail.com>
```

```
Date:    Sun Sep 4 09:37:17 2022 -0400
```

```
added cool new feature
```

Checking Available Remotes

```
git remote
```


```
origin
```





Quick Comment

That's probably one of the more complicated merges you'll ever have to do. And it wasn't *that* bad.











Finally!




 **wrathematics / demo** Public

 Pin  Unwatch **1**  Fork **0**  Star **0**


[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [...](#)

 master  Go to file  Add file  Code **About** 

 **wrathematics** Merge branch 'master' of github.com:wrath...  5 minutes ago  **4**


 LICENSE	Initial commit	9 minutes ago
 new_feature.txt	added cool new feature	7 days ago
 x.txt	adding x.txt	7 days ago


Help people interested in this repository understand your project by adding a README.





About

The demo repository used in the lectures for our CSODS course

 Unlicense license

 **0** stars

 **1** watching

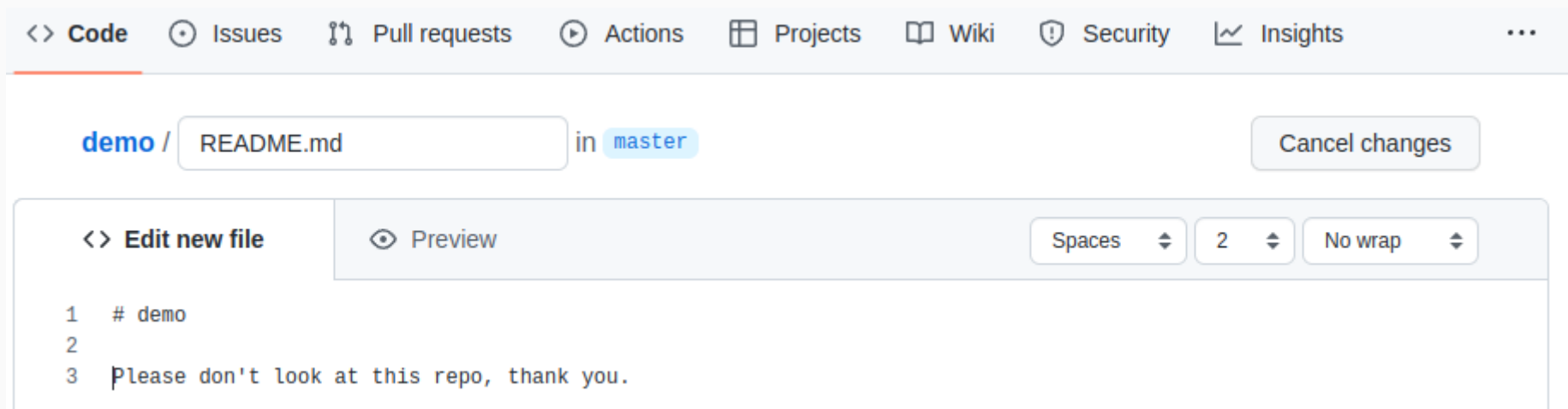
 **0** forks

Releases

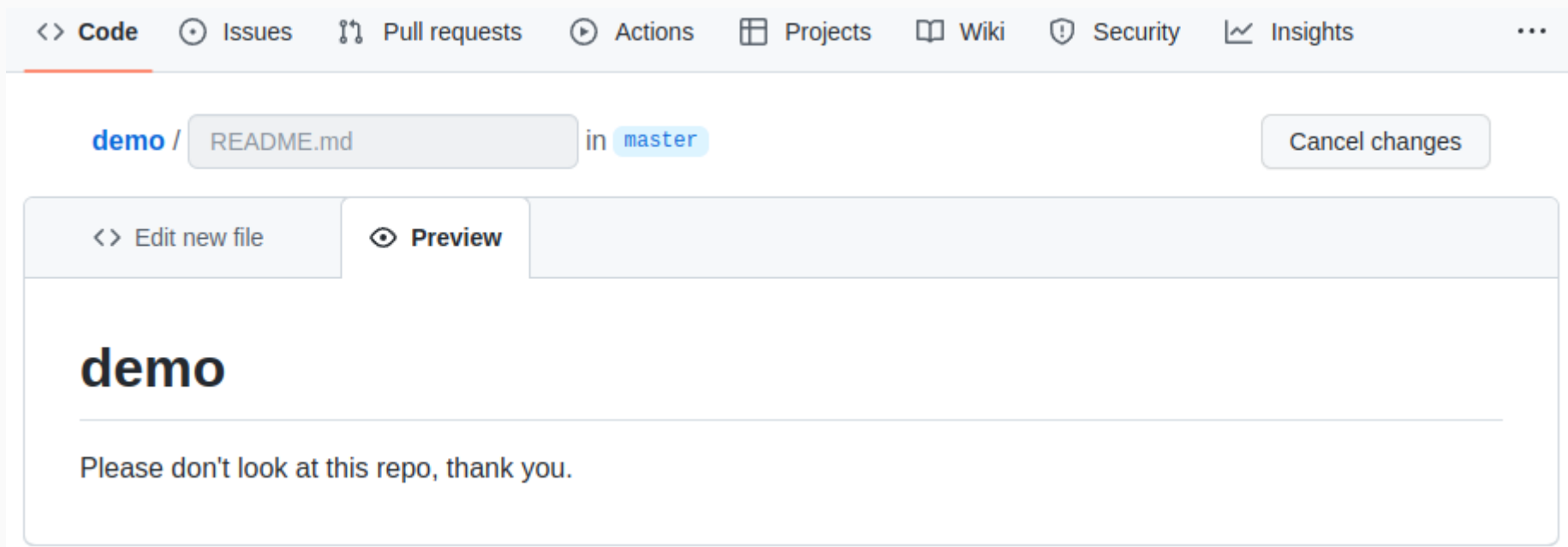
No releases published

[Create a new release](#)

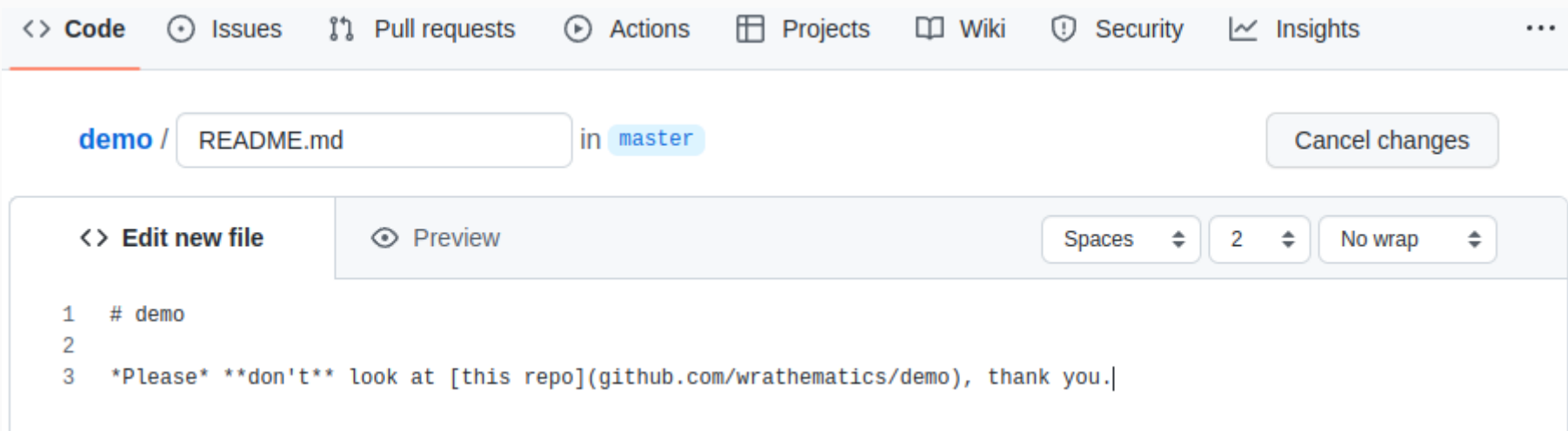
Let's Add a Readme!



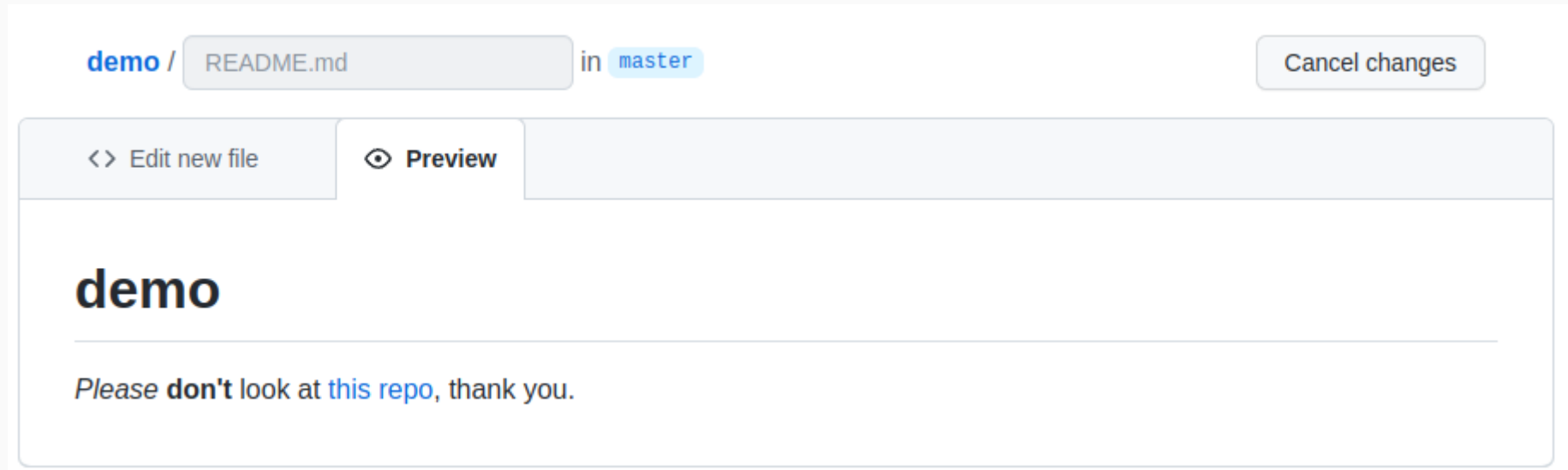
Let's Add a Readme!



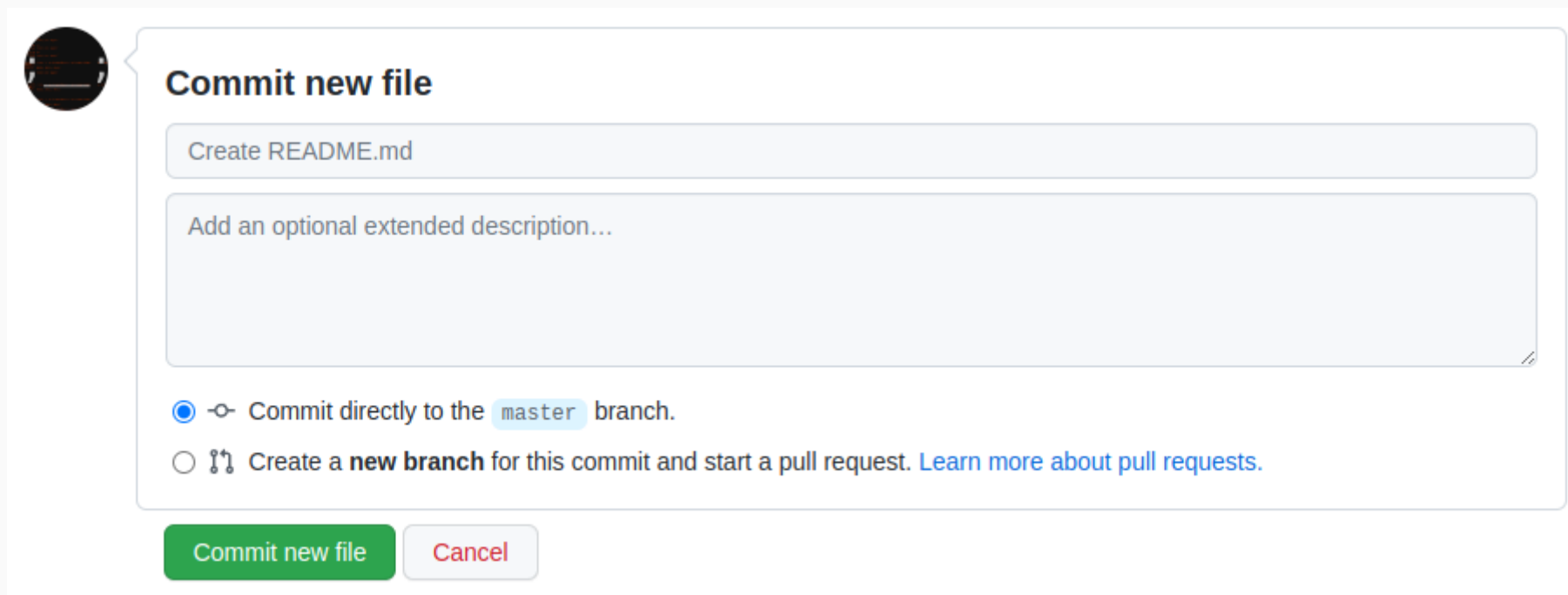
Let's Add a Readme!



Let's Add a Readme!



Let's Add a Readme!



The image shows a GitHub commit dialog box. On the left is a circular profile picture of a person with dark hair. The dialog has a title bar that says "Commit new file". Below the title bar is a text input field containing "Create README.md". Below that is a larger text area with the placeholder text "Add an optional extended description...". At the bottom of the dialog are two radio buttons. The first radio button is selected and is labeled "Commit directly to the master branch." The second radio button is not selected and is labeled "Create a new branch for this commit and start a pull request. Learn more about pull requests." Below the radio buttons are two buttons: a green "Commit new file" button and a grey "Cancel" button.

Commit new file

Create README.md

Add an optional extended description...

☒ Commit directly to the `master` branch.

☐ Create a **new branch** for this commit and start a pull request. [Learn more about pull requests.](#)

Commit new file Cancel

Let's Add a Readme!

The screenshot shows a web browser with multiple tabs. The active tab is 'github.com/wrathematics/demo/tree/master'. The repository page for 'wrathematics / demo' is displayed, marked as 'Public'. It includes interaction buttons like 'Pin', 'Unwatch' (1), 'Fork' (0), and 'Star' (0). A navigation bar contains links for 'Code', 'Issues', 'Pull requests', 'Actions', 'Projects', 'Wiki', 'Security', and 'Insights'. Below this, there's a 'master' branch selector, 'Go to file', 'Add file', and a 'Code' button. The commit history table shows:

File	Commit Message	Time
wrathematics Create README.md ... now 5		
LICENSE	Initial commit	13 minutes ago
README.md	Create README.md	now
new_feature.txt	added cool new feature	7 days ago
x.txt	adding x.txt	7 days ago

Below the table is a section for 'README.md' with an edit icon. To the right, the 'About' section states: 'The demo repository used in the lectures for our CSODS course'. It also lists 'Readme', 'Unlicense license', '0 stars', '1 watching', and '0 forks'. The 'Releases' section at the bottom indicates 'No releases published'.

Syncing

```
git pull origin master
```

From github.com:wrathematics/demo

* branch master -> FETCH_HEAD

Updating b62f530..78f0473

Fast-forward

README.md | 3 +++

1 file changed, 3 insertions(+)

create mode 100644 README.md

Syncing

```
git log
```

```
commit 78f04733b322f40b648d8c743efee43043d182d9 (HEAD -> master, origin/master)
```

```
Author: Drew Schmidt <wrathematics@gmail.com>
```

```
Date:    Sun Sep 11 17:26:14 2022 -0400
```

```
    Create README.md
```

Wrapup

Ungraded Homework

- You already have a GitHub account (right?)
- Set up ssh keys
- Add ssh keys to GitHub
- Learn some basic markdown <https://www.markdownguide.org/basic-syntax/>

Wrapup

- Working with remotes is a small extension of what we've seen so far
- Mostly straightforward, but be careful with divergent histories!
- You probably want to use ssh keys!
- Next time: Collaborating on GitHub

Questions?