Get Your GitHub Classroom Repo

Stat 133, Spring 2019

This document contains the instructions to get your own GitHub Classroom repository. This is the repo that you will use to submit your HW assignments.

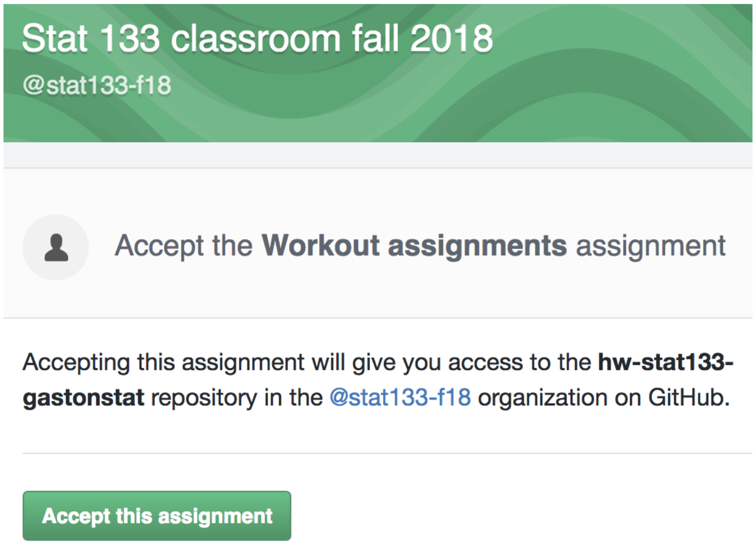
Note: The images and screenshots in this document display the github repo from Fall 2018. However, you will be running commands with the Spring 2019 repository.

GitHub Classroom

1) Sign-in to your github account.
2) Use the following invitation link to get your **GitHub Classroom** repository

https://classroom.github.com/a/L9VBb4Ti

3) You should get redirected to `https://classroom.github.com`, more specifically, to the "Stat 133 classroom spring 2019" organization. Here, you will see a message *"Accept the Workout assignments"* assignment (image from Fall 2018).



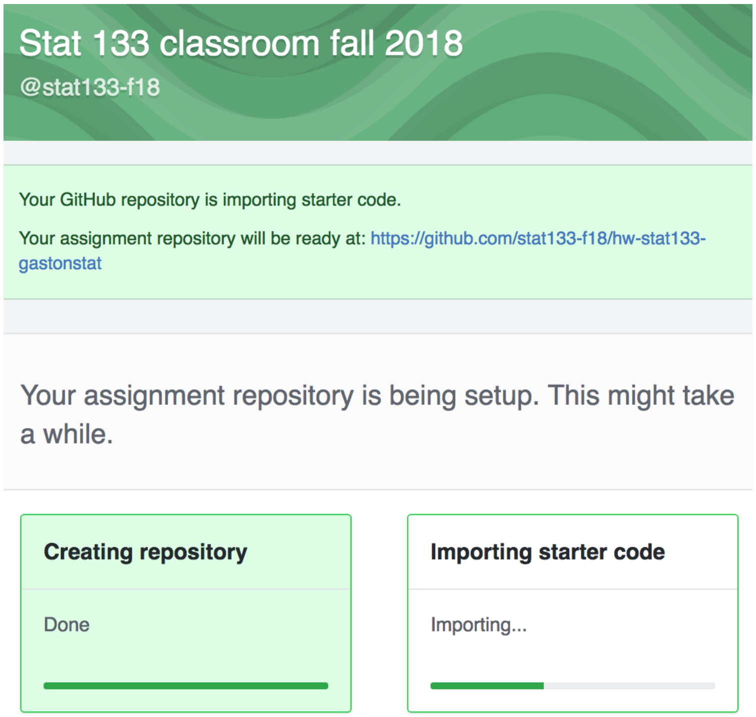
Stat 133 classroom fall 2018

@stat133-f18

Accept the **Workout assignments** assignment

Accepting this assignment will give you access to the **hw-stat133-gastonstat** repository in the @stat133-f18 organization on GitHub.

4) After clicking on the **Accept this assignment** button, you should be able to see a new message indicating that your assignment repository is being setup: this involves 1) creating the repository, and 2) importing starter code.



Stat 133 classroom fall 2018

@stat133-f18

Your GitHub repository is importing starter code.

Your assignment repository will be ready at: https://github.com/stat133-f18/hw-stat133-gastonstat

Your assignment repository is being setup. This might take a while.

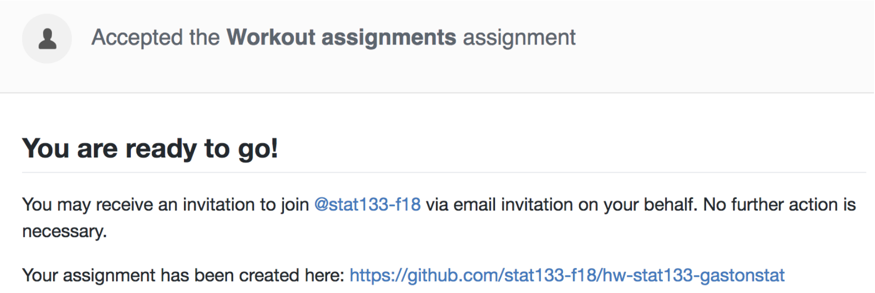
Creating repository

Done

Importing starter code

Importing...

5) When the setup is finished, you should see the message *You are ready to go!*



Accepted the **Workout assignments** assignment

You are ready to go!

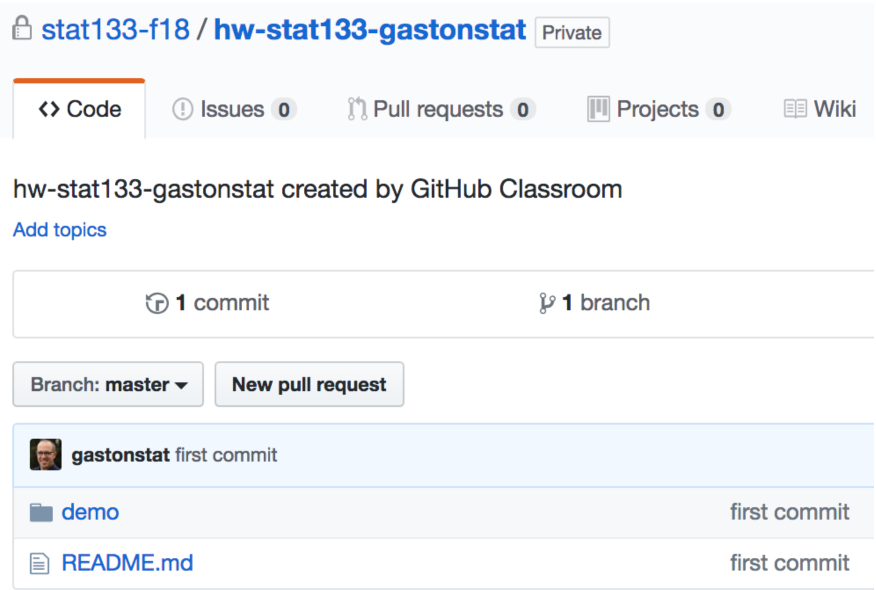
You may receive an invitation to join @stat133-f18 via email invitation on your behalf. No further action is necessary.

Your assignment has been created here: https://github.com/stat133-f18/hw-stat133-gastonstat

6) At this moment, you will receive an email with an invitation to join `@stat133-sp19`. In theory, there is no need to accept the invitation since your repository has already been created.

Your GitHub Classroom Repository

What you need to do next is to click on the assignment link created for your github account. You should be able to see your new repository

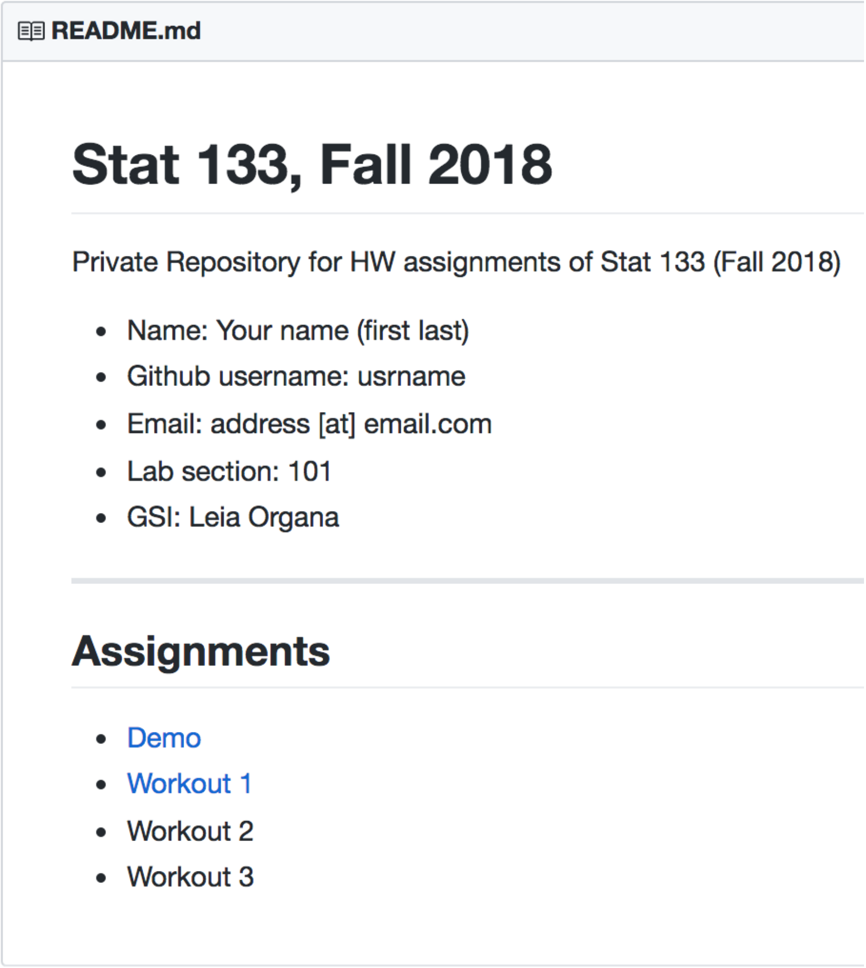


hw-stat133-gastonstat created by GitHub Classroom

gastonstat first commit

demo	first commit
README.md	first commit

As you can tell, your repo contains a `README.md` file with some default content:



Stat 133, Fall 2018

Private Repository for HW assignments of Stat 133 (Fall 2018)

- Name: Your name (first last)
- Github username: usrname
- Email: address [at] email.com
- Lab section: 101
- GSI: Leia Organa

Assignments

- Demo
- Workout 1
- Workout 2
- Workout 3

Edit the `README.md` file

Click on the `README.md` file, and then look for the icon of a pencil so that you can edit the contents of the file.

Customize the information about `Name`, `Github username`, your `Email` address (the one linked to your github account), the `Lab section` you are registered in, and the name of your `GSI`

Look for the button **Commit changes** so you can "save" the modifications done to the `README.md` file.

Your Local Repo and the Remote Repo

So far you have your GitHub Classroom repository. This repository will be your *remote* repository. However, you also need to to create a *local* repository in your computer.

- Open a bash terminal
- Optional: change directory to your preferred location e.g. your Desktop
- Create a directory, and `cd` to it:

```
mkdir hw-stat133
cd hw-stat133
```

- Initialize the directory as a Git repository

```
git init
```

- Locate the name of your github repo (with **your own username!!**)

`https://github.com/stat133-sp19/hw-stat133-gastonstat`

- To add a remote repository use the command below with **your own username**:

```
git remote add origin https://github.com/stat133-sp19/hw-stat133-gastonstat
```

- Pull down the content in the remote repo (`origin`) to your local repo (`master`)

```
git pull origin master
```

Pushing changes to the remote repo

Now that you have linked your local repo with your remote repo, you can start pushing (i.e. uploading) commits to GitHub. Try the following modifications.

In your computer (your local repo), use a text editor (e.g. the editor in RStudio) to open the `README.md` file that is inside the `demo/` folder. Under the title **Demo**, you should be able to see a first line of text:

```
This is just a demo folder with some dummy content.
```

Delete the starting paragraph and replace it with the text below (or any other text that you want to add):

```
This is just a demo folder for testing purposes.
```

Now follow these steps:

- Save the changes made in `README.md`.
- Go to the your bash terminal.
- Change directory to the `demo/` folder:

```
cd demo
```

- Check the status

```
git status
```

- Add the changes to git:

```
git add README.md
```

- Commit the changes with a descriptive message:

```
git commit -m "demo: update readme file"
cd ..
```

- Push the changes to the remote repo (`origin`) from your local repo (`master`)

```
git push origin master
```

Go to your Github repository and refresh the browser. If everything went fine, you should be able to see the contents of the `README.md` file inside `demo/`.

Push your Warm-up 1 assignment

Let's add your warmup1 `.Rmd` file, and its knitted file, to your local repo, and then push the modifications to your remote repo.

- Copy the `.Rmd` file of the first warm-up assignment and save it inside the folder `warmup1/` (of your local repo).

- Open the `.Rmd` file and modify the `output` field of the yaml header: instead of using `output: html_document`, change it to: `output: github_document`.

- Knit the `.Rmd` file. This should generate an `.md` file.

- Go to the bash terminal and check the status

```
git status
```

- Add the changes:

```
git add warmup1/.
```

- Check the status again:

```
git status
```

- Commit and push the changes:

```
git commit -m "warmup1: add Rmd and md files"

git push origin master
```

- Go to your Github repository and refresh the browser.

Push your Workout 1 assignment

Assuming that you've finished the first workout assignment, try adding all the associated files (and directories) to your local repo, and then push the modifications to your remote repo.

- You should have a dedicated folder (i.e. directory) `workout01` with all the files and subdirectories (as indicated in the filestructure of the assignment).
- I'm assuming that your `workout01` is NOT a git repository (you didn't initialize a git repository).
- In case you do have `workout01` as a git repository, you will have to avoid copying the `.git` hidden directory to your local `hw-stat133` repo.
- Copy the `workout01` to your local `hw-stat133` repository (as long as `workout01` is not a git repo already).
- Inside `hw-stat133/workout01`, git `add` and `commit` the files.
- Likewise, git `push` the committed changes to your remote repository.