
Getting Set Up for Submissions

By: Kelly Slatery

Google Classroom & Submissions Repo

Google Classroom

We will use Google Classroom to submit/give feedback on:

- Labs
- Blog posts
- Quizzes
- Projects
- Capstone Check-ins

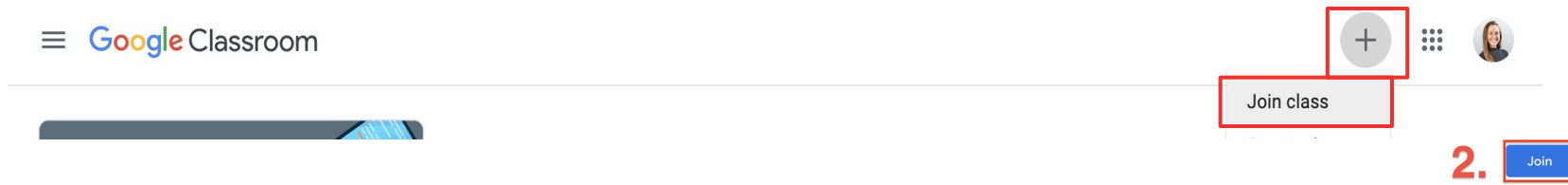
Submissions Repo

You will create a private repo on GHE and add all instructors as collaborators.

You will be submitting the url links to labs/quizzes/projects in this repo on Google Classroom.

1. Join Google Classroom


1. Go to classroom.google.com
2. Click on the “+” in the upper right and select “Join Class”



3. Enter code: **2k3ay2y** and click “Join”

1

You're currently signed in as

 **Kelly Slatery**
kelly.slatery@gmail.com

[Switch account](#)

Class code
Ask your teacher for the class code, then enter it here.

Class code

To sign in with a class code

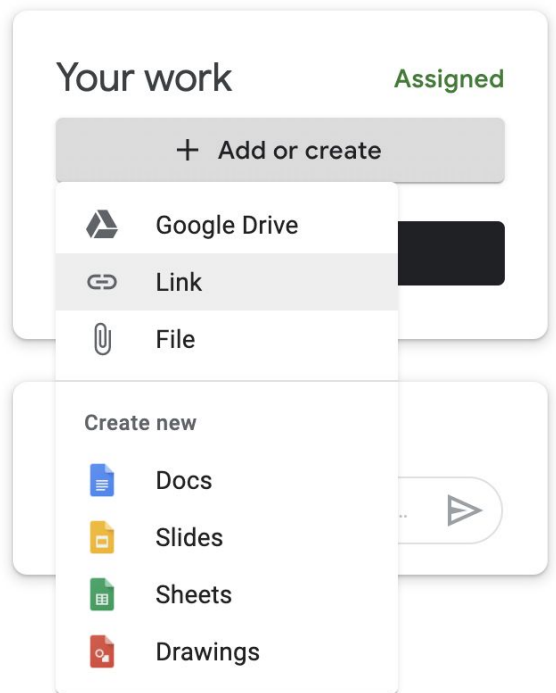
- Use an authorized account
- Use a class code with 5-7 letters or numbers, and no spaces or symbols

If you have trouble joining the class, go to the [Help Center article](#)



Steps to Submit

1. Go to our [Google Classroom](#) (bookmark this page)
2. Via “Stream” or “Classwork”, find the assignment you are submitting
3. Click “View Assignment”
4. Click “+ Add or create” in the upper right
5. Choose “Link”
6. Paste the url link to that lab notebook in your Submissions repo on GHE



2. Set up a private Submissions repo on GHE

1. Go to [GHE](#)
2. In the upper right-hand corner, click + > New repository

New repository

New gist

New organization

Create a new repository

A repository contains all project files, including the revision history.

Owner: KellySlattery / Repository name: Submissions-demo

Great repository names are short and memorable. Need inspiration? How about sturdy-disco?

Description (optional)

☐ Public
Anyone can see this repository. You choose who can commit.

☒ Private
You choose who can see and commit to this repository.

Skip this step if you're importing an existing repository.

☐ Initialize this repository with a README
This will let you immediately clone the repository to your computer.

Add .gitignore: None

Create repository

3. Verify your username appears under “Owner”
4. Name your repository
“Submissions”
5. Ensure that “Private” is selected
6. You do not *need* to initialize with a README, but you may. You can always go back and create one later.

Create a new repository

A repository contains all project files, including the revision history.

Owner



KellySlattery ▾

Repository name *

Submissions-demo ✓

Great repository names are short and memorable. Need inspiration? How about **sturdy-disco**?

Description (optional)



Public

Anyone can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Skip this step if you're importing an existing repository.



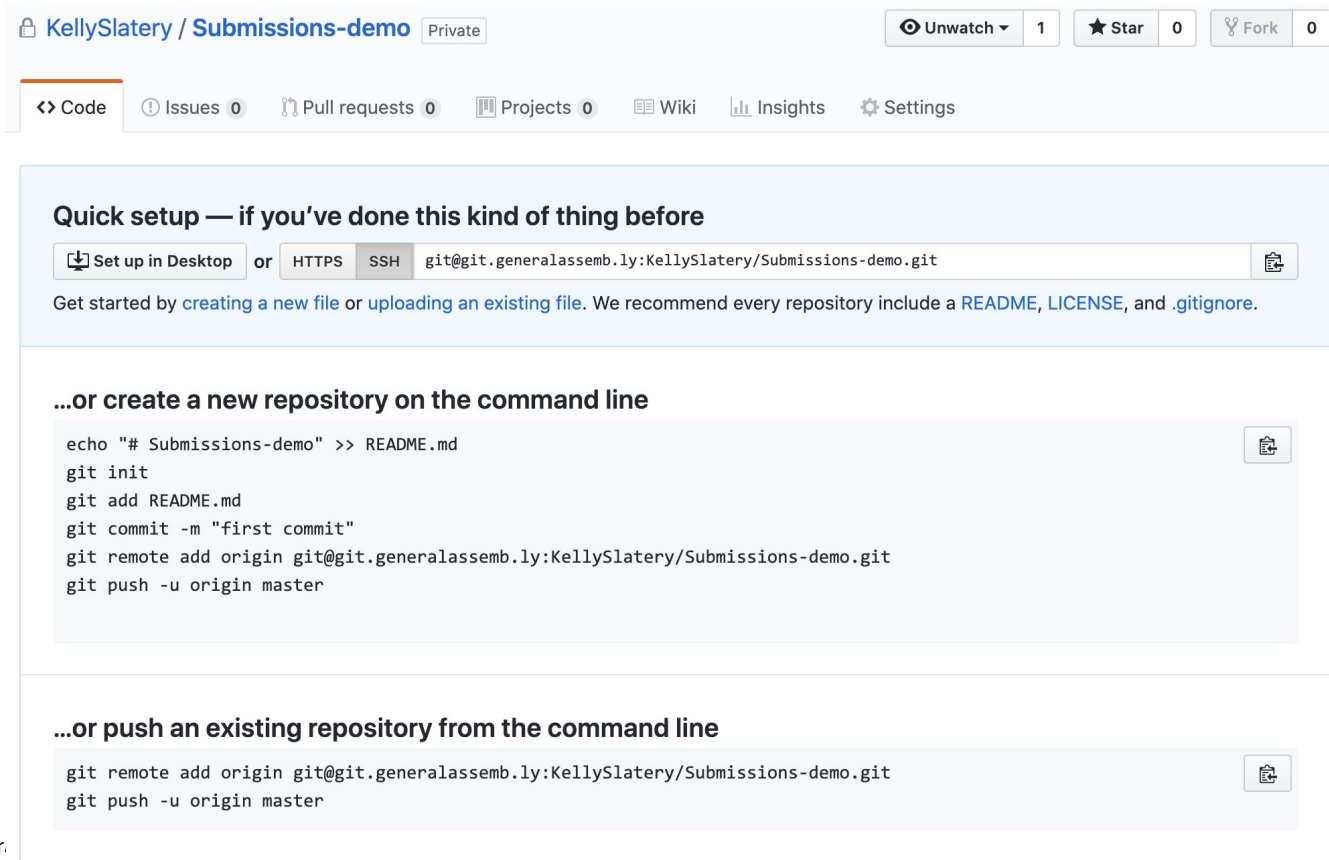
Initialize this repository with a README

This will let you immediately clone the repository to your computer.

Add .gitignore: None ▾

Create repository

If you initialized without a README, you should see something that looks like this:



The screenshot shows the GitHub interface for a repository named 'Submissions-demo' by user 'KellySlatery'. The repository is marked as 'Private'. At the top, there are buttons for 'Unwatch' (1), 'Star' (0), and 'Fork' (0). Below this is a navigation bar with links for 'Code', 'Issues' (0), 'Pull requests' (0), 'Projects' (0), 'Wiki', 'Insights', and 'Settings'. The main content area has a light blue header with the text 'Quick setup — if you've done this kind of thing before'. Below this header, there are two options: 'Set up in Desktop' and 'HTTPS' (selected), with the SSH option also visible. The SSH URL is 'git@git.generalassemb.ly:KellySlatery/Submissions-demo.git'. Below the options, there is a text block that says 'Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).' Below this, there are two sections for creating or pushing a repository from the command line. The first section is titled '...or create a new repository on the command line' and contains a code block with the following commands: 'echo "# Submissions-demo" >> README.md', 'git init', 'git add README.md', 'git commit -m "first commit"', 'git remote add origin git@git.generalassemb.ly:KellySlatery/Submissions-demo.git', and 'git push -u origin master'. The second section is titled '...or push an existing repository from the command line' and contains a code block with the following commands: 'git remote add origin git@git.generalassemb.ly:KellySlatery/Submissions-demo.git' and 'git push -u origin master'.

KellySlatery / Submissions-demo Private

Unwatch 1 Star 0 Fork 0

<> Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH git@git.generalassemb.ly:KellySlatery/Submissions-demo.git

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# Submissions-demo" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin git@git.generalassemb.ly:KellySlatery/Submissions-demo.git
git push -u origin master
```

...or push an existing repository from the command line

```
git remote add origin git@git.generalassemb.ly:KellySlatery/Submissions-demo.git
git push -u origin master
```

7. Add all of the instructors/IAs as collaborators:
- AdiBro, pwalesdi, KellySlattery, JEllena, nchristiansen

1

KellySlattery / Submissions Private

Watch 0 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

2

Options Collaborators Branches

Collaborators Push access to the repository

Adi Bronshtein AdiBro

<> Code

! Issues 0

🔗 Pull requests 0

📁 Projects 0

📖 Wiki

📊 Insights

⚙️ Settings

Quick setup — if you've done this kind of thing before



Set up in Desktop

or

HTTPS

SSH

git@git.generalassemb.ly:KellySlattery/Submissions-demo.git



Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# Submissions-demo" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin git@git.generalassemb.ly:KellySlattery/Submissions-demo.git
git push -u origin master
```

...or push an existing repository from the command line

```
git remote add origin git@git.generalassemb.ly:KellySlattery/Submissions-demo.git
git push -u origin master
```

8. Clone your repo (with HTTPS or SSH -- you can always reclone it later with SSH once it's set up

pwd → check your current working directory

cd dsi → use the appropriate path to get to your dsi folder

git clone
paste-the-link-you-copied

9. In your dsi folder in terminal, create a folder for Labs and one for Projects with `mkdir`

10. Go to the lab/project/quiz on GHE that you want to complete (the link will be in the description on Google Classroom)

11. Fork the lab to your personal GHE account

12. **DOWNLOAD ZIP** -- if you clone the lab inside your Submissions repo, *you will run into git issues*



Lab 1.01 Pokemon

Kelly Slattery • 12:03 PM (Edited 12:09 PM)

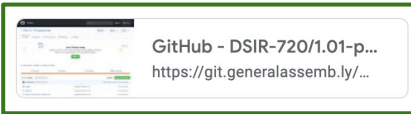
1 point

Due Jul 27, 11:59 PM

Please go to link, fork it, download it and put it in your Submissions folder/repo. Open the lab notebook in Jupyter Notebook, work on it and then add, commit and push it to GitHub Enterprise.
TO SUBMIT - submit the link to this lab in your Submissions repo.

Labs are marked as 1 - "Complete" or 0 - "Incomplete" (you're on the right path but you have incomplete questions or errors in your code)

We will mark it as "Missing" if you haven't submitted it at all.



0 releases

1 contributor

Create new file

Upload files

Find file

Clone or download

Clone with SSH

Use HTTPS

Use a password protected SSH key.

git@git.generalassemb.ly:KellySlattery/1.01-pokemon



Open in Desktop

Download ZIP



—
One more time for the people in the back...


DOWNLOAD ZIP

if you clone the lab inside
your Submissions repo, you
will run into git issues

```
(base) kelly@MacBook-Pro Code % cd ~/Downloads
(base) kelly@MacBook-Pro Downloads % unzip 2.01-lab-pandas-titanic-master.zip
Archive: 2.01-lab-pandas-titanic-master.zip
212df17067e4deffde6b18f1006357db456aec23
  creating: 2.01-lab-pandas-titanic-master/
  creating: 2.01-lab-pandas-titanic-master/.ipynb_checkpoints/
  inflating: 2.01-lab-pandas-titanic-master/.ipynb_checkpoints/Titanic_KellySlattery-checkpoint.ipynb
  inflating: 2.01-lab-pandas-titanic-master/README.md
  inflating: 2.01-lab-pandas-titanic-master/Titanic_KellySlattery.ipynb
  inflating: 2.01-lab-pandas-titanic-master/train.csv
(base) kelly@MacBook-Pro Downloads % mv 2.01-lab-pandas-titanic-master ~/Code/dsi/Submissions/Labs
(base) kelly@MacBook-Pro Downloads %
```

12. Open/Unpack your zip file (on a Mac, just click on the button in the browser OR double-click in Finder) (in terminal: **unzip zip-file-name**)
13. Using your GUI (graphical user interface, on Mac, “Finder”) OR terminal, move the unzipped files into your Labs folder in your Submissions folder (a repo)



- 
14. DONE! To edit your lab, you can open it in jupyter notebook or lab, save, then remember to:

```
git add .
```

```
git commit -m "informative message"
```

```
git push origin master
```

15. To verify that it worked, navigate to your remote Submissions repo (the one in your browser on GHE) and make sure you see your changes in the lab :-))

YOU DID IT!!!!
Now get
working :-)



