Module 0: Git and GitHub

Arturo Torres Ortiz

2026-01-13

Learning objectives

By the end of this module, you will be able to:

1. Configure your local Git environment.
2. Clone your class repository.
3. Use the core Git commands (`add`, `commit`, `push`) to submit an assignment.

1. Why version control?

Git is a version control system that tracks changes in your files over time. It allows you to:

1. **Revert** to previous versions of your code if you break something.
2. **Collaborate** with others without overwriting their work.
3. **Document** *why* changes were made using commit messages.

GitHub is a cloud-hosting service for Git repositories that includes powerful tools for project management and collaboration.

2. Setting up Git

Before you can use Git, you need to tell it who you are. This information will be attached to every "save" (commit) you make.

Open your terminal and run the following commands, replacing the name and email with your own:

```
# Set your name (this appears in the project history)
git config --global user.name "Your Name"

# Set your email (must match your GitHub email)
```

```
git config --global user.email "your.email@usc.edu"

# Set the default branch name to 'main' (standard practice)
git config --global init.defaultBranch main
```

To verify your setup, run:

```
git config --list
```

3. The git workflow: clone, edit, commit, push

To submit an assignment in this class, you need to follow these four steps.

Before starting, you need to accept the assignment via GitHub Classroom using the following link:

`https://classroom.github.com/a/nMEEuL7F`

If you already have a github account, you can link it. Otherwise, you need to create one. Once you accept the link, you have to select your name from the roster. Your repository will be automatically created.

Step 1: Connect git repository to the cluster.

You only need do this once per assignment. It downloads the repository from GitHub to your local machine or, if the directory already exists in your local machine, it links it to the github remote repository.

Option 1: Clone

Use this option if you are starting fresh and you don't have an existing local repository. You already have a folder in the USC server so we don't need to do this.

```
# Navigate to where you want to keep your coursework
cd /scratch1/username

# Clone your specific repository (replace URL with your classroom repo link)
git clone https://github.com/trgn-515-2026/student-repo-USERNAME.git

# Enter the directory
cd student-repo-YOURNAME
```

Option 2: Initiate existing Folder

Since you already have a folder for your user on the cluster, we don't need to "clode" one from github. However, you need to turn your normal folder into a git repository.

```
# 1. Enter your folder
cd /scratch1/username

# 2. Initialize Git
git init
git branch -M main

# 3. Link to GitHub (Replace URL with your repo link)
git remote add origin
 ↪  https://github.com/TRGN-515-2026/student-repo-USERNAME.git

# 4. Pull any existing files (like the README) from GitHub
git pull origin main --allow-unrelated-histories
```

Step 2: Edit code

Create files, write code, and save your work as you normally would. For this class, you will create a new folder for each assignment within your user repository.

Step 3: Stage and Commit (Save)

Git has a two-step saving process.

- Add (Stage): Select which files you want to save.
- Commit: Actually save them with a message describing what you did.

```
# Check which files have changed
git status

# Add a specific file...
git add my_script.py

# ...OR add everything in the current folder
git add .

# Save the snapshot with a message
git commit -m "Completed exercise 1"
```

Step 4: Push (Upload)

Your commits currently live only on your computer. To submit your assignment, you must upload them to GitHub.

```
git push origin main
```

Setting Up Assignment 1

For your first task, you will practice this workflow by setting up the directory structure for Assignment 1.

Task 1: Create the Directory Structure

Note

Inside your local user repository (eg: /scratch1/username/trg515, or whatever you called it), create a folder named 1_sequencing_techs.

```
mkdir -p 1_sequencing_techs
```

Task 2: Copy the Template

Note

Download the assignment_template.md file from:

`https://github.com/trgn-515/trgn-515.github.io/blob/main/modules/0_intro_git/assignment_te`

Save it inside `1_sequencing_techs`

Rename the file to 1_sequencing_techs.md.

Task 3: Render and Submit

Note

Open 1_sequencing_techs.md in RStudio or VS Code.

Change the author field to your name.

Render the file to PDF.

Use Git to push both the qmd file and the generated .pdf to GitHub.

```
# 1. Check that Git sees your new files
git status

# 2. Add the new folder and files
git add modules/1_sequencing_techs/

# 3. Commit the changes
git commit -m "Setup assignment 1 structure"

# 4. Submit
git push origin main
Once you have pushed, go to your GitHub repository URL in your
 ↪  browser. If you see the modules folder and your PDF, you
 ↪  have successfully submitted Assignment 0!
```