

---

# Lab 0: Learning to use Git

---



*Due Monday, 11 January 2021 11:59 PM*

## Minimum Submission Requirements

- Create 6 folders in your repository labeled: Lab0, Lab1, Lab2, Lab3, Lab4, and Lab5 (**note the capitalization convention**)
- Add README.txt files in each of the 6 folders
- Commit and push your repo to the GitLab server

## Objective

This lab will introduce you to version control and how to use it within the context of this class. In addition to setting up your repository directory structure, you will read the syllabus and [personal responsibility document](#) and write a statement confirming that you have read the documents.

Version control is an extremely important part of engineering. Git is an industry-standard tool for version control, and you should know how to use it as part of your engineering education (for example, Microsoft Windows development is version controlled using Git).

## Preparation

### Register

Before the course staff can create a repository for you, you must register for a [Git @ UCSC](#) account. Note that this is different from a GitHub account. Repository creation is a manual process, so it will take some time before your repo is created.

### Command Line

Familiarize yourself with the command line first.

#### Videos

[Intro to the Command Line](#)

[The Most Important Thing You'll Learn in the Command Line](#)

[Terminal Basics for Beginners](#) (turn your volume down before pressing play)

#### Read

[Introduction to Git and Shell Commands](#)

### Git

Go through these resources after you are comfortable navigating the command line.

#### Tutorials

[Git Immersion](#)

[Learn Git Branching](#)

## Videos

[Git & GitHub Crash Course for Beginners](#)

[Git Tutorial for Beginners](#)

## Procedure

Once you have your GitLab account and a repo for the class, you are going to verify that you have read both the class syllabus and personal responsibility documents. You will be “digitally signing” the documents by creating a README.txt that includes a sentence stating you have read and understood these documents.

This and other labs will be machine graded. That is, a program will run that will check the files and generate your grade. It is extremely important to **match the names and directories exactly (NOTE THE CAPITALIZATION CONVENTION STATED ABOVE!!!)**; the naming convention is necessary for the grading program to find your work. As in all engineering, in this course, **attention to detail is paramount**. Stories abound where small errors lead to catastrophic failures; in this class it could only lead to a failing grade, but you might find that catastrophic. Pay attention!

## Steps

1. Register for an git UCSC account: [https://git.ucsc.edu/users/sign\\_in](https://git.ucsc.edu/users/sign_in)
2. Review the command line and Git resources listed in the preparation section of the lab.
3. Clone your repository. (If you are using a lab computer, make sure to push all changes you’ve made to Gitlab. This way when you come into the lab next time you can re-clone it into any lab computer and pick up where you left off.)
4. Create the following directories in your repository: Lab0, Lab1, Lab2, Lab3, Lab4, and Lab5. **Note the capitalization convention.**
5. Use the “touch” command to create a README.txt file in each of the 6 folders.
6. **Carefully** read the syllabus (on Canvas) and [personal responsibility document](#).
7. Modify README.txt in the Lab0 directory to include a statement that you have read and understood the syllabus and personal responsibility document. Use this format exactly (replace Jane Doe with your first and last name, jdoe with your CruzID, and the date with the current date).

*I, Jane Doe, have read and understood the Spring CSE12 syllabus and Personal Responsibility Document. jdoe 3 April 2020*

Be careful not to make any spelling errors.

8. Add all the README.txt files to your Git repository.
9. Commit your repository.
10. Push your commit to the server.
11. Go to the GitLab web interface and verify that the files have the right names and are in the right folders – i.e., verify that you have met the minimum submission requirements for this lab.

## SSH Keys

To avoid having to enter your Git credentials every time you add a file or commit, you may set up an SSH Key on your own personal computer. Instructions on how to do this are posted [here](#).

On the lab computers, the directory where you would need to save the ssh key gets wiped when you log out, so these instructions are only relevant to using your own personal computer.

## Notes

Be familiar with adding one file at a time to Git, in addition to adding multiple files at a time. Students sometimes accidentally add files that they don't mean to (like adding files for previous labs) which end up costing them late hours.

This lab is quite short, but if you are unfamiliar with Git it can take some time. Start early and there should be no issues with finishing it well before the due date.

This is how you will submit **ALL** labs in this class, and they will all have a similar set of minimum submission requirements. It is important that you learn to do this right at this point in time.

No further work in the class will be graded unless this lab is completed; this is our verification that you have been informed of our expectations of you. Following instructions carefully may seem arbitrary (to you) but is absolutely necessary both from an engineering and pedagogical viewpoint.

You should **commit early and often (at the very least once per day** your work on any lab). This ensures you lose minimal work in case of a malfunctioning computer or other unfortunate event. For this reason, extensions will **never** be issued due to a malfunctioning personal computer.

In addition, having many commits demonstrates your incremental progress and is your best defense against allegations of cheating.

Every time you make a change, add a file, modify a file, go ahead and commit and push. You will always be able to return to that place and never lose your work.

## Grading Rubric

- 2 pt created all Lab{0,1,2,3,4,5} directories
- 2 pt created all Lab{0,1,2,3,4,5}/README.txt files
- 2 pt correct statement in Lab0/README.txt (check your spelling!)
- 2 pt included CruzID
- 2 pt date in the correct format