

Week 1

Discussion Session (Please make sure you are muted.)

MPCS 51042: Python Programming

University of Chicago

Discussion Session Agenda

1. Zoom meeting guidelines
2. Assignment distribution
3. Office hours overview
4. Working with Gitlab
5. Coding Exercises
6. Open discussion: Week 1 material and course

Zoom Meeting Guidelines

- Please make sure you are muted when entering the discussion.
- Do not un-mute yourself unless you are called out by name.
- Sometimes I will open the floor up for questions. In that case, please use Zoom's "Raise Hand" and I will call on you.
- If you have a (non-Zoom) question at any point, please type it in the chat (I will periodically pause to look at these questions)
- If you are encountering issues related to Zoom, please try shutting down your machine and signing back in. These meetings are recorded so if you miss them then you'll be able to watch it at a later time.

Assignment Distribution

- Assignments will be submitted via your Gitlab repository:
<https://mit.cs.uchicago.edu/mpcs51042-aut-20/CNETID.git>
- All assignments are out the following day (in the evening) after our weekly discussion sessions.
 - Reason: Gives you the opportunity to ask me general questions during the discussion session.
- Assignment descriptions will be posted on the course website. All starter code will be given to you inside your repository before the assignment is posted.
- Full instructions on how to submit and your homework will be given in a video that will come tomorrow.

Office Hours Overview

- All office hours will be done via Zoom meetings. I will create links inside our Canvas page this weekend.
- The TA (Collin Olander) will hold his office hours on **Monday @ 6:00pm-8:00pm CT.**
- Lamont's office hours: **Thursdays @ 1:30pm-3:30pm CT.**
- Office hours will be one-on-one time with me. Similar to office hours in person.
 1. You will sign up on Piazza in the office hours post.
 2. More details about office hours will come tomorrow before the first session.

Working with Gitlab: Homework Uploading Process

Git: A Version Control System

Git is a DVCS (Distributed Version Control System). It is a system that allows for tracking file changes in a project. It has this notion of a **repository** that tracks the status for files in a project.

In Git, there are two kinds of repositories:

- **Local Repository**: This repository acts as the working copy of your repository on your local machine (laptop, desktop, etc.). Changes are **committed** (saved) to your local repository.
- **Remote Repository**: This repository is a repository that lives on a machine somewhere other than your local machine (e.g., web server). After saving changes to your local repository you need to **push**(upload) them to your remote repository. If remote repository is updated by someone else (e.g., Me, TA, grader), you can **pull** (download) those changed to your local repository.

Accessing your Remote Repository

- For this course, your remote repositories are hosted on the CS department's GitLab server:
<https://mit.cs.uchicago.edu>. You can also find this link on the course website.
- Your login credentials is your CNET id and password.
- When you login you will be presented with all repositories created for you on the CS's Gitlab server.
 - One should be a private repository for this class:
<https://mit.cs.uchicago.edu/mpcs51042-aut-20/cnet-id>

Cloning your Remote Repository

Cloning is the process of creating your local repository from your remote repository. You *may* need to use SSH authentication to be able to clone your remote repository.

1. Generate or locate your SSH public key:

`https://mit.cs.uchicago.edu/help/ssh/README`

2. Add your public key to GitLab:

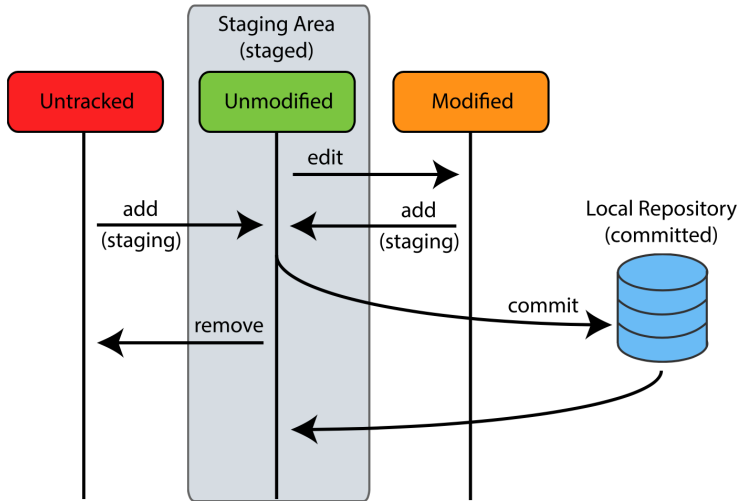
`https://mit.cs.uchicago.edu/profile/keys`

File Tracking States

When working with files in your local repository it will go through the following different states:

- **Untracked**: Files that are not part of the repository (Git doesn't know about these files).
- **Unmodified**: Files that have not been modified since the last commit. They will be included in the next commit.
- **Modified**: Files that have been modified since the last commit and will not be committed.

File Tracking States, cont.



Coding Exercise

Open discussion: Week 1 material
and course
