Git Repo

For our term project, each enrolled student shall visit https://edu.cs.illinois.edu/create-ghe-repo/cs427-fa18/ by logging in with NetID. Doing so will automatically create an individual repository for each student and also add the student to the GitHub organization. Then our TA will grant students in a team as stated in the team assignment the access to the team repository.

In particular, there will be a team repository for each team for the term project. The link structure of the team repository will be https://github-dev.cs.illinois.edu/cs427-fa18/T999, where T999 will be replaced with the team number (e.g., T801).

If you encounter any issue for setting up or accessing your individual repository or team repository, please contact our TA Peilun Zhang <peilunz2@illinois.edu>.

Instructions of Using the Repo

We use the CS-hosted GitHub for MPs and projects this semester. Followings are instructions on how you should submit your files.

For MPs:

- You should submit files to your private repo (named after your NetID) under the CS427 organization.
  - e.g. https://github-dev.cs.illinois.edu/cs427-fa18/netid
  - For each MPs, please make a directory for it.

For Projects:

- You should submit files to the team repo (named after your team number) under the CS427 organization.
  - e.g. https://github-dev.cs.illinois.edu/cs427-fa18/T***

Please make sure you have git installed on your computer. Most of the *nix system comes with git installed. You may also use this page to help you install git: https://www.atlassian.com/git/tutorials/install-git

Getting Started:

1. Clone your repo via command line. (You can definitely use any GUI programs you like)
   ```bash
   git clone https://github-dev.cs.illinois.edu/cs427-fa18/your_netid
   ```
2. Create an unique directory for each assignment.
   ```bash
   mkdir mp1
   ```
   Remember to put all your files inside this directory.
3. After finishing your assignment, you can use the following commands to show the current status
   ```bash
   git status
   ```
4. Adding edited files for git to track
   ```bash
   git add FILENAME (add FILENAME to the system) or git add . (to add everything)
   ```
5. Commit your changes
   ```bash
   git commit -m "Some meaningful commit message"
   ```
   Writing down meaningful messages is a good practice :) 
6. Push your changes to CS Github (Very Important) git push or git push -u origin master
   ```bash
   ```
   Your assignments cannot be graded if you do not push it to the remote repos
7. If you want to change one of your submitted files, make the changes to the file and redo "add, commit, and push". We will only grade your latest answers before the due date by default. For late submissions, see Grading Policy.

Important Notes:

1. We will be using grading scripts to grade the assignments so please name your files according to descriptions of the MP, e.g. one of the files for MP1 should be named as "CoffeeMaker_Web_NetID.zip" where NetID is replaced with your UIUC NetID.
2. Your submission must be a proper project that we can run, or you may receive 0 points. So please double-check (or triple-check) that you have the correct spelling of the zip/text file names of the submitted files.
3. It is very useful to learn git work flow for team projects. This article may help: https://www.atlassian.com/git/tutorials/comparing-workflows
   /gitflow-workflow
4. The website above also provides a good Git tutorial. If you would like read it, it's https://www.atlassian.com/git/tutorials