Setting up git

git in CS 527

For our HWs and term project, each enrolled student shall visit https://edu.cs.illinois.edu/create-ghe-repo/cs527-sp19/ 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 (to be released later) 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/cs527-sp19/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 the instructor.

Instructions of Using the Repo

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

For MPs/HWs:

1. You should submit files to your private repo (named after your NetID) under the CS527 organization.
   - e.g. https://github-dev.cs.illinois.edu/cs527-sp19/netid
   - For each MPs/HWs, please make a directory for it.

For Projects:

1. 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/cs527-sp19/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)
   
   `git clone https://github-dev.cs.illinois.edu/cs527-sp19/your_netid`

2. Create an unique directory for each assignment.
   
   `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
   
   `git status`

4. Adding edited files for git to track
   
   `git add FILENAME` (add FILENAME to the system) or `git add .` (to add everything)

5. Commit your changes
   
   `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`

   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.

Important Notes:

1. 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
2. The website above also provides a good Git tutorial. If you would like read it, it's https://www.atlassian.com/git/tutorials