Show Submission Credentials

P0. Git Best Practices A document for git best practices

✓ Intro

Intro

Intro to Git Best Practices

These are some suggested `git` best practices for working individually or in teams to make you a master of your code's lifecycle! Git is version control tool that allows you to manage the code you write in a local repository, push to external repositories (on GitHub or hosted elsewhere), experiment on different areas of your code, and much more! Additionally, Git is a tool which startups and large corporations alike expect their engineers to be completely comfortable with and fluent in especially while working in teams.

If this is your first time using `git` , we recommend you to learn by doing with the following hands-on interactive Git tutorials: Git-it (https://github.com/jlord/git-it-electron) and LearnGitBranching (https://learngitbranching.js.org/).

Here is a list of Git and GitHub specific glossary (https://help.github.com/articles/github-glossary/), and you should get familiar with the following concepts before you move on: Branch, Clone, Commit, Fork, Merge, Pull, Push, Repository, and Upstream.

If you don't have `git` installed on your machine, and don't feel comfortable exploring a way to install `git` yourself, please refer to Git's suggestions https://git-scm.com/book/en/v2/Getting-Started-Installing-Git (https://git-scm.com/book/en/v2/Getting-Started-Installing-Git). We have installed git on the student AMI for you in most projects.