**CS445 - Lab 1 Collaborative Git**

1. Github Setup – create a github repository for daily lab
2. What you need to submit for this lab
   1. Create a .md file for Lab1
   2. Inside the .md file includes all screenshots when you practice the commands below
3. Markdown study: <https://guides.github.com/features/mastering-markdown/>
4. Markdown for images: <https://www.markdownguide.org/basic-syntax/#images-1>

**[4 points] Exercise 1: Practice with all the local repository commands from the slides:**

|  |
| --- |
| git init  Graphical user interface, text, application  Description automatically generated |
| git status |
| git diff  Graphical user interface, text, application, email  Description automatically generated |
| git add |
| git commit  Text  Description automatically generated |
| git log  Text  Description automatically generated |
| git show  Text  Description automatically generated |
| git restore  Text  Description automatically generated |
| git tag  Text  Description automatically generated |
| git branch  Text  Description automatically generated |
| git checkout  Text  Description automatically generated |
| git merge  Text  Description automatically generated |
| Create a conflict and solve the conflict.  Graphical user interface, text, application, Word  Description automatically generated |

**[2 points] Exercise 2: Practice with all the remote repository commands from the slides:**

1. Create a remote repository
2. Create README.md
3. Add at least 5 markdown syntax (For example: Headers, Bold, Code, Unordered List, Links) [Cheatsheet](https://github.com/adam-p/markdown-here/wiki/Markdown-Cheatsheet" \t "_blank)
4. git remote
5. git clone
6. git push
7. git fetch
8. git pull

**[4 points] Exercise 3: Practice with all the third party repositories commands from the slides:**

Assume we are working with the following upstream: https://github.com/bellaxing/CS445-May2021-Lab1

1. Star the repository
2. Watch the repository for any changes
3. Fork the repository
4. Clone your forked repository into your local machine
5. Create a new topic branch named using studentId-studentName, For example: 610001-JohnSmith
6. Checkout your new branch
7. Add a new file with your name, example: hello.js
8. Commit the changes
9. Merge your studentId-studentName branch with the main branch
10. Push your changes to the remote repo
11. Create a new Pull Request to the upstream repo

**What you need to submit:**

**1. Write a status about your homework, such as how much percentage you finish in this homework, etc**

**2. Copy and paste your github link for this Lab1**