Git cheatsheet exercise

In this exercise you will work on a simple HTML project in a **Team of 2 or more people**. If you never had to chance to write an HTML document don't worry! You're not required to know HTML, in this exercise you only have to do minor modifications to template documents (however, if you are interested in building web pages check out this link: http://www.w3schools.com/html/).

A starting web project is already provided in the "git-cheatsheet" folder. This simple web site is a cheatsheet containing a list of the most commonly used git commands. From the home page the user can select a command and navigate to a specific web page containing more details about it. Two examples (*git init* and *git log*) are already implemented, feel free to add the commands you wish!

NOTE: Before starting, ensure that you have installed a text editor on your computer. You can find a good one here: http://www.sublimetext.com/

Part 1

Initialize your project:

1. Form a team of 2 or more people and nominate a <u>team master</u>
2. The team master should do the following:
    a. Copy "git-cheatsheet" in a different location (e.g. his/her desktop)
    b. Initialize a git repository within the **COPIED** "git-cheatsheet" folder (do not initialize a repository in the original folder!)
    c. Do an initial commit on branch master adding all the files
    d. Create a remote repository on GitHub
    e. Invite his/her teammates to the repository giving them write access
    f. Link his/her local repository to the remote repository
    g. Do a first push of the project

3. Then, each teammate should do the following:
    a. Clone the repository on his/her computer
    b. Verify that all the files pushed by the team master are found within his/her computer

Part 2

Working on the project (no branches, just use master for now!)

Each person in the team should do the following:
1. Think about 2 git commands to add within the list
2. For each command do:
    a. Open index.html with a text editor and insert the command in the homepage index.html (see the other listed commands for hints)
    b. Create a new html page within folder "commands" by copying an existing one (e.g. git-init.html or git-log.html)
    c. Insert some text about the command
    d. Commit the work on master
    e. Push the work on the remote branch master (If conflicts arise you may need to do git fetch and git merge – or – git pull)
3. Fetch updates from the other teammates
4. Merge the remote modification in your local branch

Part 3

Working on the project (with branches)

Each person in the team should to the following:
1. Create a local branch with his name or username (e.g. branch "marcor")
2. Switch to the branch just created locally
3. Push the branch to the remote repo (inform remote of a new branch)
4. Think about other 2 git commands to add within the list
5. For each command do:
    a. Add the command to index.html
    b. Create a new html page with the command info
    c. Commit the work on your local branch and push it to your remote branch
6. Finally, when your work is complete, switch to master and merge the modification done in your branch
7. Push your local branch master to the remote branch master (If conflicts arise you may need to do git fetch and git merge – or – git pull)
8. Switch to your branch and merge the modification done in master