

Git Workflow

Note: The steps below are a basic git workflow for our project that is intended to get you started using git. If we all follow this workflow, our lives will be much easier. This seems like a lot of steps, but I stretched it out to make each step as simple as possible for you. Remember, you can always ask me a question if you have an issue or are unsure about something.

- 1) Open a terminal or cmd prompt from within your local git repository and then go to step 2.
- 2) Check to see what branch you are on by typing the command: **git branch** and go to step 3.
- 3) Are you currently on the master branch? If yes, continue to step 4. If no, go to step 7.
- 4) Have you already created a branch before, and you want to go to that branch? If yes, type the command: **git checkout <BRANCH NAME>** and then jump to step 8 . If no, go to step 5.
- 5) You need to create a new branch, but first, ensure that you have the latest version of master. Type the command: **git pull origin master** and then go to step 6.
- 6) Create a new branch by using the command:
git checkout -b <BRANCH NAME> and go to step 8.
- 7) If this is the branch you want to be on, go to step 8. If not, run the command:
git checkout master and go to step 4.
- 8) Do some work on your branch. When you are done, go to step 9.
- 9) Are you done your work and ready to commit your changes? Look at the files you have changed by typing: **git status** and then go to step 10.
- 10) Select the files you want to commit by using the command: **git add <FILE NAME>** for each file you want to commit. If you want to commit all of the changed files, you can use the command: **git add .** (dot is a part of this command) Go to step 11 when you're done.
- 11) Now that you have added your changes, it is time to commit those changes. Type the command: **git commit** and some form of a text editor will open. Go to step 12.
- 12) In the editor, write the ticket # from Trello that your worked on (if there is one) and then write a message briefly explaining what changes were made. Save the file and go to step 13.

13) Now that you have committed the changes, you want to push them to master, but first, you must rebase on master. This is very important that you always do this. To do this type the command: **git checkout master** and then go to step 14. Don't worry, your changes were committed and are safe on your branch.

14) Now that you are back on master, you should pull the latest in case others have made changes to master while you were busy working. Run the command: **git pull origin master** and go to step 15.

15) Now that you have ensured that master is up to date. Switch back to the branch that you made changes in, by using the command: **git checkout <BRANCH NAME>** and then go to step 16.

16) You should be back on the branch where you did work and committed your changes. Now type the command: **git rebase master** and go to step 17.

17) Assuming that you have no merge conflicts, you are ready to push to master! Write the command: **git push origin master** and go to step 18.

18) You did it! Your changes were pushed to master, and now you are free to work on something else. Go back to step 1.

To find more useful git commands, check out the [git cheat sheet](#).

For an interactive browser tutorial on git, try [codecademy](#).

For video tutorials, try one of these out:

- [Up and Running with Git and GitHub](#) (Lynda.com)
- [Git Essential Training](#) (Lynda.com)
- [YouTube tutorial](#)

[Google Slides presentation on Git](#)