

Git

Saturday, September 9, 2023 12:17 PM

1. Create a Git repository on git hub.
2. Clone the repository using:
 - a. `git clone <url>`
3. Create new files in terminal using:
 - a. `touch <filename>`
4. Edit file in text editor.
5. Commits: save points
6. Add files to the commit using:
 - a. `git add <filename>`
 - b. this allows you to commit only the files that you are happy with your edits to.
7. How to commit:
 - a. `git commit -m "message"`
 - b. commit all files
 - i. `git commit -am "message"`
8. Track the status of your local repository using:
 - a. `git status`
 - b. This will tell you what branch you are on and how many commits you have past the repository stored on git.
9. Push the commits on your local repository to git
 - a. `git push`
10. Pull latest versions of files off of git hub using:
 - a. `git pull`
11. Merge conflict
 - a. Your partner edits a line of code and uploads it to the git repository.
 - b. You also edit that line of code without having pulled your partners edit.
 - c. That will cause a conflict.
 - d. You will then get a message saying conflict and you can just go to the file and determine how you want to fix it. Then commit the fix and push.
 - e. The top part of the merge message is your change
 - f. The bottom part of the merge message are the remote changes
12. Track all your commits and edits using
 - a. `git log`
 - b. To get out use "q"
 - c. See the edits that you made in a commit:
 - i. `git show <commit_hash>`
 - d. Compare commit to prev version
 - i. `git diff <commit_hash>^ <commit_hash>`
13. Make a change that you didn't mean to make you can go back with:
 - a. `git reset`
 - i. `git reset --hard <commit hash>`
 - 1) go back to particular commit
 - ii. `git reset --hard origin/main`
 - 1) go back to version that is on github.
14. Branching:
 - a. Look at all branches and see what you are currently on:
 - i. `git branch`
 - b. Switch branch

- i. `git checkout <name>`
 - c. Make new branch and switch to it
 - i. `git checkout -b <name>`
 - d. Merge branches
 - i. be on the branch that you want to merge into
 - 1) `git merge <name>`
 - ii. This merges `<name>` branch into the current branch we are on.
 - e. Merge conflict: editing the same line on two branches
 - i. Fix same way as before
 - f. Then commit changes.
 - g. Reset a branch back to main:
 - i. Be on the branch that you want to reset.
 - ii. `git reset --hard main`
15. Forking:
- a. copy a repository to your account where you can then edit it on your own git.
 - b. Then when you edit it and you want to update the original repository you can start a pull request, that would ask the repository owner to accept or deny your changes to the repository.
16. Git hub Pages:
- a. create repository called:
 - i. `<username>.github.io`
 - b. Then you can clone it
 - c. This allows us to update a website instantly while developing and pulling and pushing edits.
17. Track all actions:
- a. `git reflog`
 - i. shows you the hash for every action you took
 - b. reset to an action
 - i. `git reset --hard <action_hashcode>`