{coding_ academy

Hello GIT!

Nice to git ya

Basic Setup

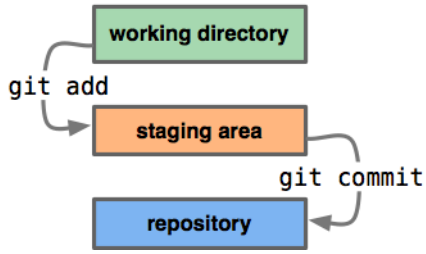
1. Start by yourself
2. download git http://git-scm.com/download/win
3. run and install, put git in your path
4. git --version
   (you can run git from anywhere now)

1. `git config --global user.name "Your Name"`
2. `git config --global user.email youremail@example.com`

Note

We are not using branches yet, so when you google, ignore all checkout commands
I've **marked** the first time we use each git command here for easy tracking

Lesson 1 - Local Repo



3. In your dev folder, start with the basic project
4. `git init`
5. `Git status`
6. `git add .`
7. `Git status`
8. `git commit –m "My first commit!"`
9. `Git status`
10. `Git log`
11. in utils.js add a function isPrimeNum(num)
12. `Git status`
13. `Git add .`
14. `Git status`
15. `git commit –m "added isPrimary function"`
16. `Git log`

Lesson 2 - Remote Repo

Now we want to synch our local repo with a remote repo (such as: github, bitbucket)

1. login or signup to github
2. create a new repo: git-ex
   (its common to use the folder name (use lower-case dash-mode))
3. `git remote add origin https://github.com/...`

4. `git remote -v`
5. `git push origin master`
6. See your files in github

Lesson 3 - git clone, add, commit and push

1. (Caution!) Delete your local project
2. In your dev folder run git clone <repo url>
3. `Cd git-ex`
4. `Git status`
5. Fix a bug in the file utils.js
6. `git status`
7. See that the file is modified and is not (yet) staged for commit
8. Add all to staging area:
   `git add -A`
9. `git status`
10. Lets commit our changes:
    `git commit -m "..."`
11. `git status`
12. Lets push it to the remote repo:
    `git push`
13. `git status`
14. Check to see your changes and commit in github
15. Add a new file:
    getRandomColor()
16. Delete the tempfile.txt
17. `git status`
18. `git add -A`
19. `Git commit -m "Cleanup the project and added getRandomColor"`
20. `Git status`
21. `Git push`
22. Check in github

Lesson 4 – Add a README markdown file

1. Inside your project, create a README.md file
   See the syntax here: https://en.support.wordpress.com/markdown-quick-reference/
2. git add README.md
3. git commit –m "…"
4. git push
5. see in github

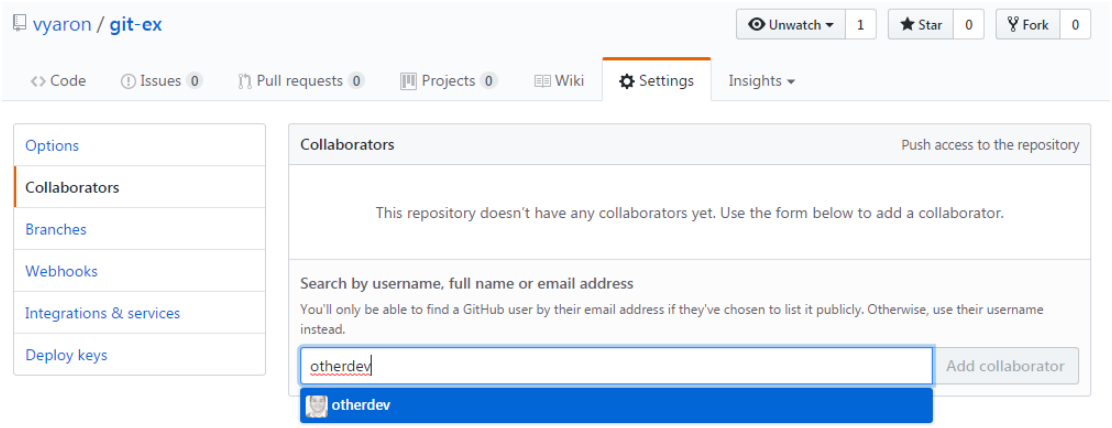
Lesson 5 – git directive files

1. read about .gitkeep and try it out
2. read about .gitignore and try it out

Lesson 6 – work together

Now, let's work with another developer on the same repo.

So select one of your repo, add the other team member as a collaborator:



1. Now (gently), try to work together:
    a. add a new file
    b. changed the README.md
2. To see your team mate changes use:
   git **pull**
   (git pull === git fetch && git merge)

Lesson 7 – merging a conflict

1. Change the same line (can be a comment line with the name of the last author)
2. Manually merge the changes
3. Read a bit about it to see different strategies:
   https://stackoverflow.com/questions/161813/how-to-resolve-merge-conflicts-in-git

Lesson 8 – github pages

Making your project accessible from github is easy:

1. Under settings look for github pages
2. Set it up on the master branch
3. You're done.

Putting it all together

Work on the given touchTheNums, push it to one of your github accounts and work together on improving it, try to pull and push as often as possible to practice git.

coding_
academy

1. git commit

2. git push

3. leave building