

— Git & GitHub for Teams



By: Kelly Slatery

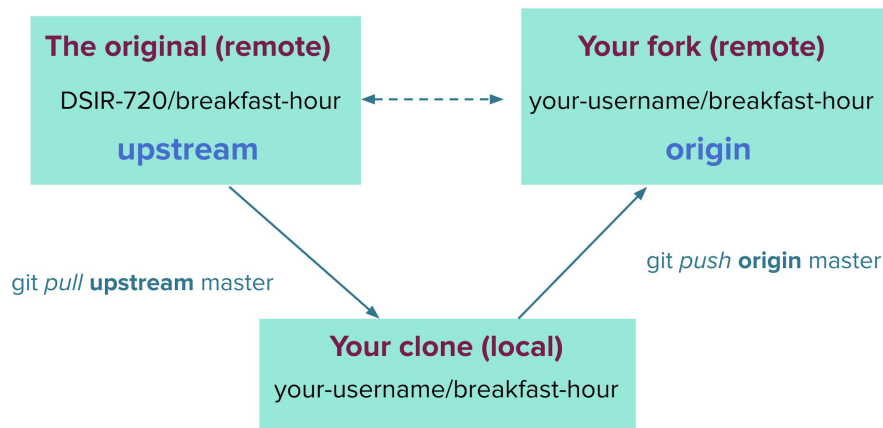
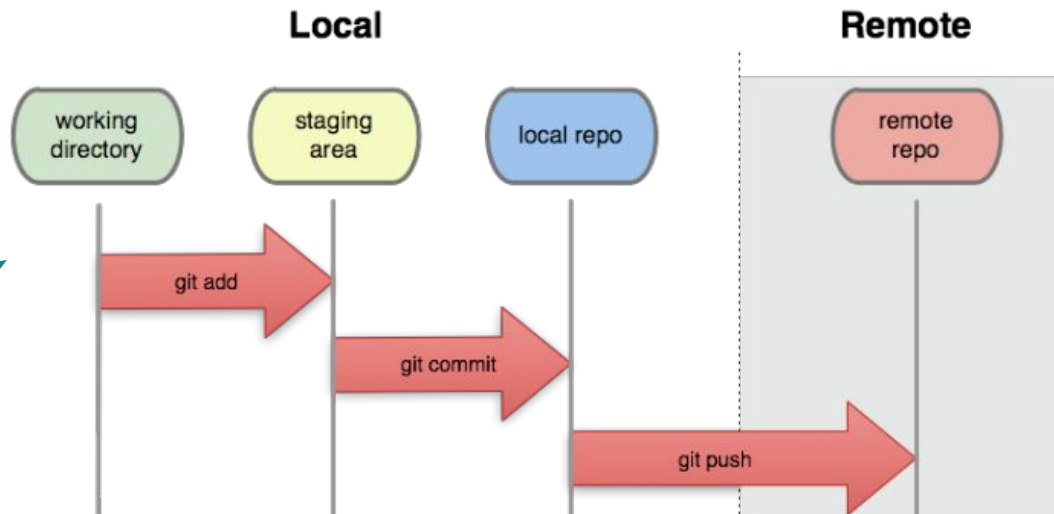
Image credits: DSI London & Aymeric Flaisler, DSI London

—
**How have we been using
git/GitHub so far?**



Git/GitHub so far...

- **Lessons** - Fork & clone
- **Submissions** - Your own repo
- **Breakfast Hour** - Two remote repos (origin & upstream)



Git/GitHub so far...

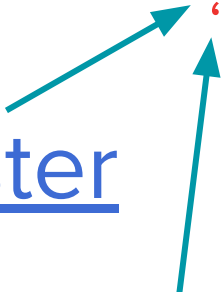
- `git clone`
- `git add {file(s)}`
- `git commit -m "message"`
- `git push origin master → {repository} {branch}`
- `git pull upstream master → {repository} {branch}`
- ...what else??



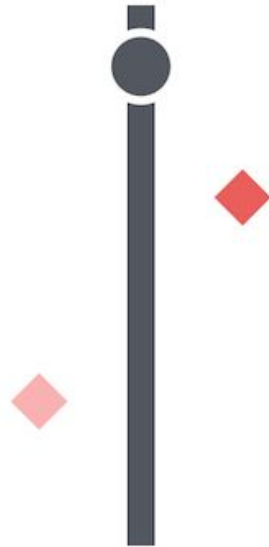
Git/GitHub so far...

git push origin master
git pull upstream master

This is the
“branch”!

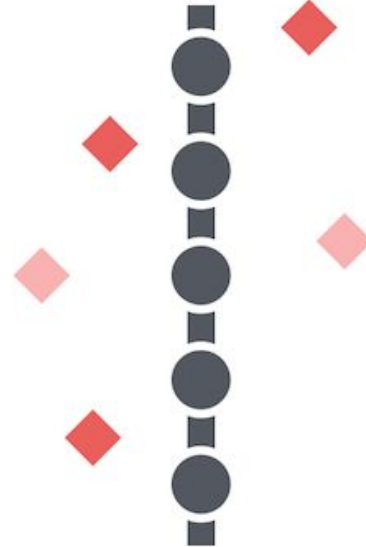


Git/GitHub so far...



first steps

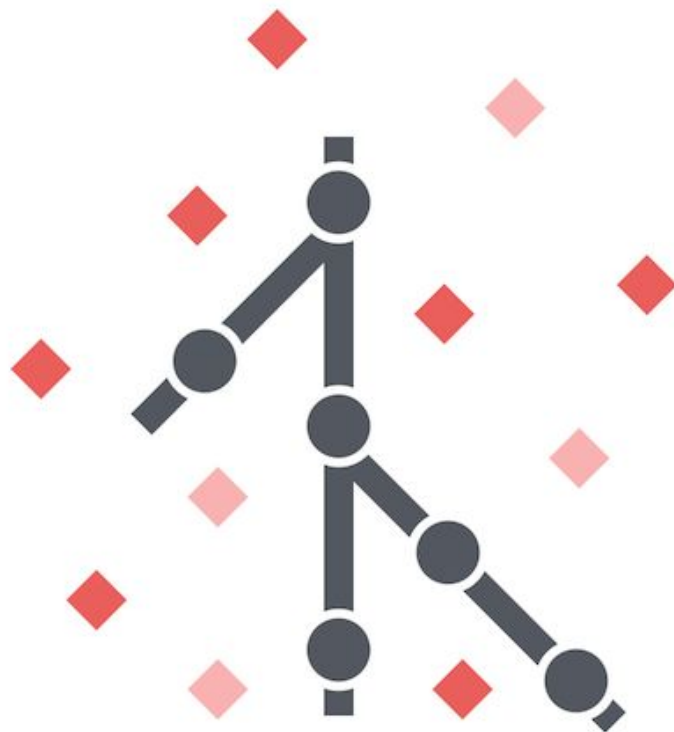
Solo workflow - without branching!



solo workflow

Git/GitHub NOW!

Team workflow - with
branching!

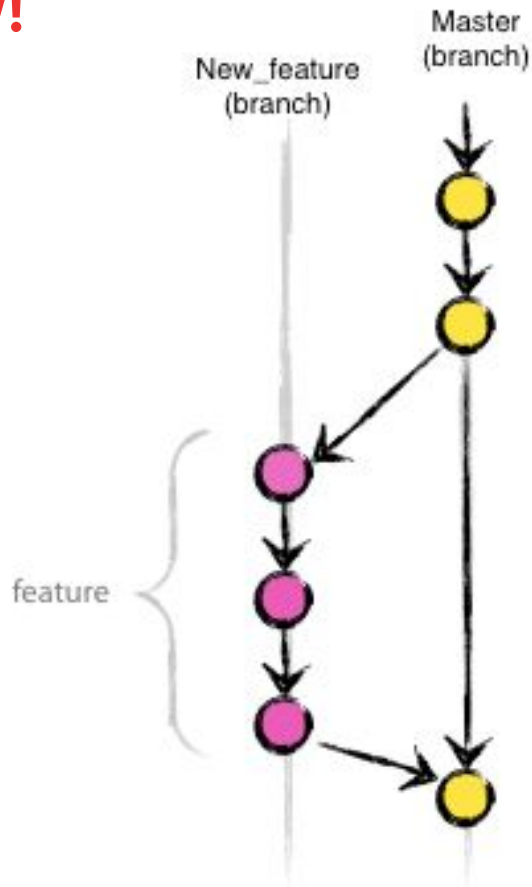


collaboration

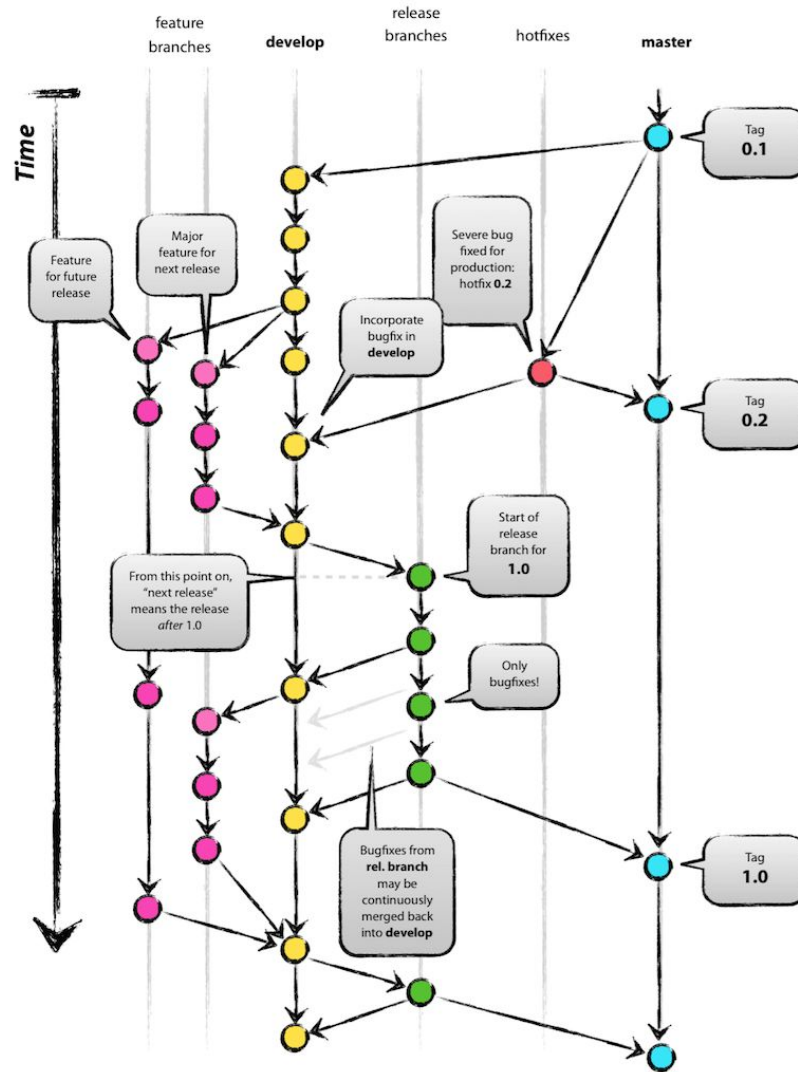


Git/GitHub NOW!

Merging a branch



Git/GitHub... — later!



Branching on *large teams* in industry!!

Git/GitHub Team Workflow Options

1. One main repository, everyone creates a branch for the feature they're working on and opens pull requests between branches (all within one shared repository)
2. One main repository, everyone creates a fork and clones their individual fork, then opens pull requests between the fork and the main original repository
3. One main repository, everyone creates a fork, everyone uses branches on their forks to keep track of the various features they're working on and maximize version control!! Pull requests are opened between the fork and the main original repository



Git/GitHub Team Workflow Options

1. One main repository, everyone creates a branch for the feature they're working on and opens pull requests between branches (all within one shared repository)
2. One main repository, everyone creates a fork and clones their individual fork, then opens pull requests between the fork and the main original repository
3. One main repository, everyone creates a fork, everyone uses branches on their forks to keep track of the various features they're working on and maximize version control!! Pull requests are opened between the fork and the main original repository





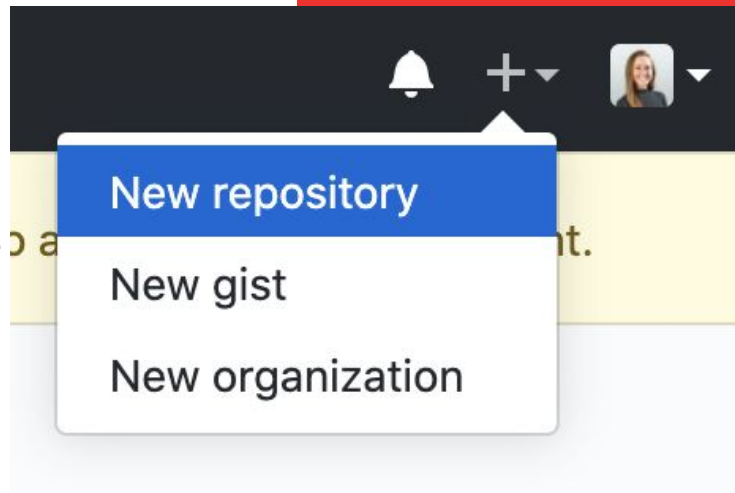
Let's make a branch!

Open GitHub Enterprise in your browser



Steps to Make & Use a Branch

1. Create a new repository on GitHub Enterprise
2. Clone your repository
3. **git branch your-branch-name**
4. **git checkout your-branch-name**
5. Make your changes!
6. **git add .**
7. **git commit -m "v informative message"**
8. **git push origin your-branch-name**
9. Go to the browser to create a...*pull request!*



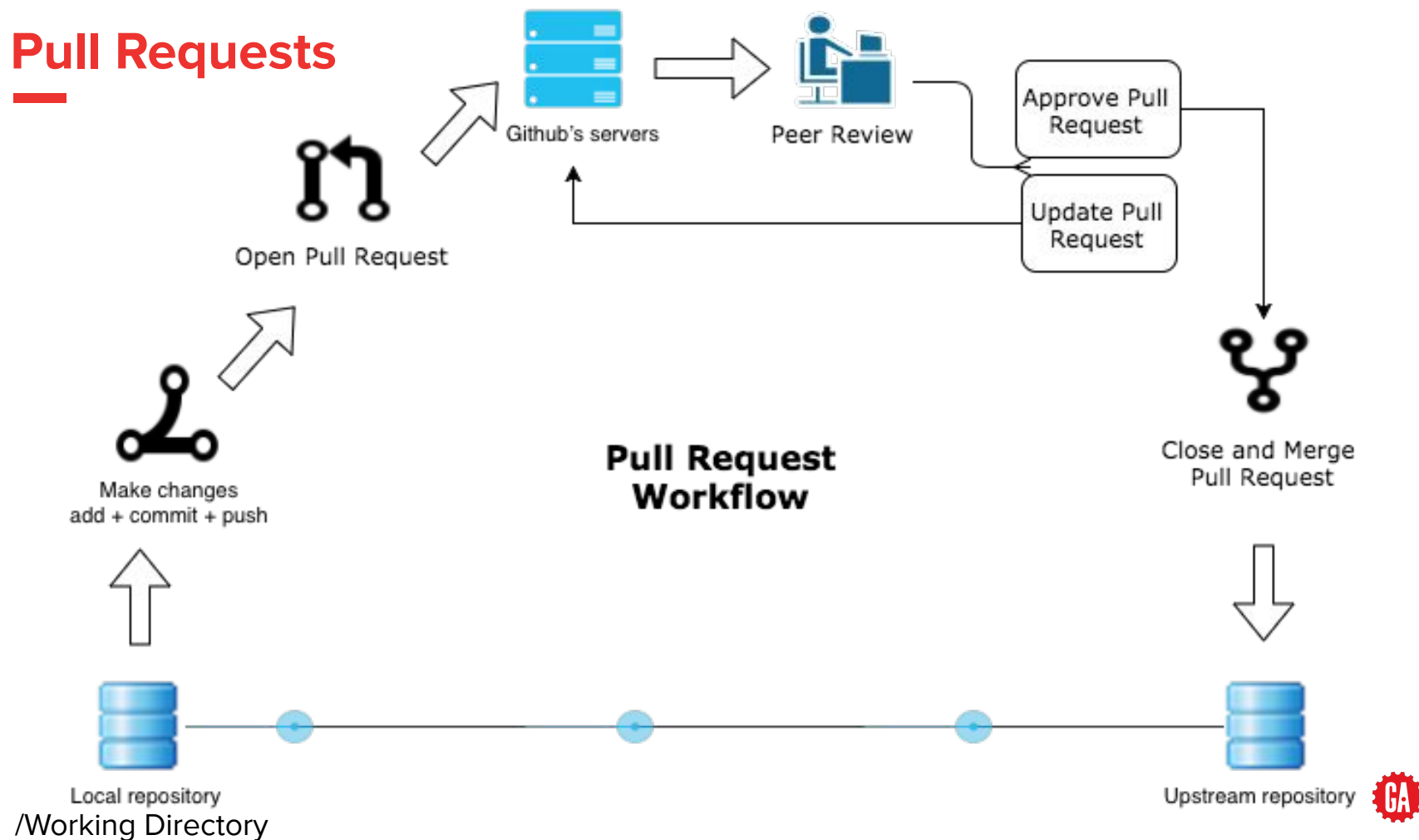
Pull Requests: Do we already do them???

Yes! “git pull upstream master” → *pull!*

...but it's a little different.



Pull Requests



Pull Requests

What is a pull request ?

After adding, committing, and pushing, if you're collaborating on a project, one of the common ways to **share your code is to do a pull request (PR)**.

At that point, whoever you're working with **will look over the code you've issued a pull request on**, and make any comments/suggestions they have, or ask any questions.

If hosted through Github (or some similar system), these typically **take place through the browser**. In it, you can have a discussion about pieces of the code and even view the changes made (see the pictures in the following slides).

Once your code is ready to merge, the other person (or you if you have the correct rights) will merge via the browser (ie: "accept" the Pull Request, and then close it).

No description, website, or topics provided.

Edit

[Add topics](#)

12 commits

1 branch

0 releases

2 contributors

Branch: master ▾

New pull request

Create new file

Upload files

Find file

Clone or download ▾

This branch is 1 commit ahead, 1 commit behind DSI-LDN-4:master.

[Pull request](#) [Compare](#)



aflaisler added git add shortcuts and warning

Latest commit 5e3c41f 42 seconds ago



week01/day1_intro_github_and_command_line

added git add shortcuts and warning

40 seconds ago



.gitignore

week01

an hour ago



README.md

monday lessons

an hour ago



README.md

lessons-repo

Week 1: Programming Fundamentals

Session Time	Day 1	Day 2	Day 3	Day 4	Day 5
9:00 - 10:30	Welcome to Data Science				
10:45 - 12:30	Command line lesson				



Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#).



base fork: DSI-LDN-4/lessons-repo ▾

base: master ▾

...

head fork: aflaisler/lessons-repo ▾

compare: master ▾

✓ **Able to merge.** These branches can be automatically merged.



Create pull request

Discuss and review the changes in this comparison with others.



↩ 1 commit

📄 2 files changed

💬 0 commit comments

👤 1 contributor



Commits on Oct 12, 2017



aflaisler

added git add shortcuts and warning

5e3c41f

📄 Showing 2 changed files with 121 additions and 1 deletion.

Unified

Split

122 ■■■■ week01/day1_intro_github_and_command_line/lesson-2-intro-to-git/Intro-to-Git.ipynb

View



@@ -104,6 +104,7 @@

```
104 104    " - clone: Makes a copy (clone) of a repository into a newly created directory, with a reference still pointing to
105 105    " - add: Add one or more files to the index (e.g. tell Git to keep track of these files)\n",
106 106    " - commit: Commit your changes, creating a "checkpoint" that can then be referenced or reverted back to later\n",
107 107    + " - status: Displays paths that have differences between the index file and the current HEAD commit (last commit)\n",
107 108    " - push: Updates a remote copy of the repository with local changes\n",
108 109    " - pull: Updates a local copy of the repository with remote changes\n",
109 110    " "
```



@@ -117,6 +118,82 @@

```
117 118    }
118 119    },
119 120    "source": [
121 121    + " - The command **add** tells your local repo to start tracking files.\n",
122 122    + "\n",
123 123    + "`bash\n",
124 124    + "git add week01/lesson-1.py # adding the lesson-1 in the folder \n",
```





aflaisler commented just now



No description provided.



added pull request

4ce375c

Add more commits by pushing to the **master** branch on **aflaisler/lessons-repo**.



This branch has no conflicts with the base branch

Merging can be performed automatically.

Merge pull request



You can also [open this in GitHub Desktop](#) or view [command line instructions](#).



Write

Preview

AA ▾ B i



Leave a comment

Reviewers

No reviews—request

Assignees

No one—assign yourself

Labels

None yet

Projects

None yet

Milestone

No milestone

Notifications

Unsubscribed

You're receiving notifications because you authorized this repository

Merge Conflict?!?!

—



Quick Tips & Tricks - Merges

AVOID THEM!

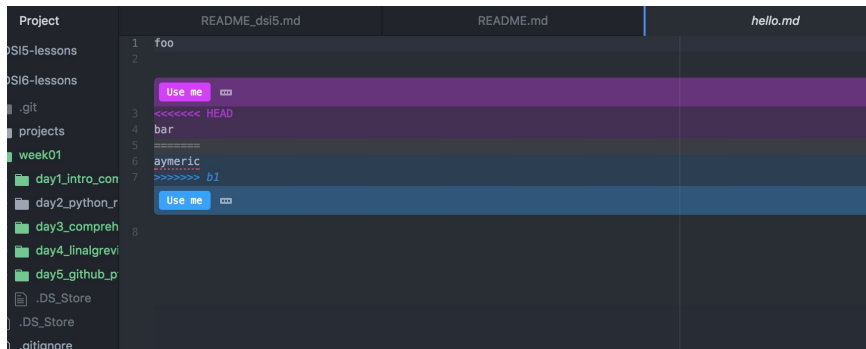
- Work on *separate, uniquely named files*; set up *naming conventions/organization* from the get-go
- Don't push & pull blindly--be aware of what you are pulling/pushing at all times BEFORE doing it & write **good commit messages**
- It is easier to avoid errors than to fix them :-)

Browser

- Review changes & commits, make comments in the GUI

Terminal

- To stop a merge:
 - `git merge --abort`
- You can open the files in an editor atom to view the differences and choose which to keep!



Practice!

1. Create a file on the master branch
2. Add and commit this file
3. Create a new branch
4. Modify the file on both branches to have diverging versions
5. Add and commit the respective modifications on both branches **Note that you now need to specify the branch you are pushing to!*
6. Merge the two branches (in the browser, or try the command line!) - the two versions should be divergent enough to lead to a merge conflict
7. Solve the merge conflict
8. Proceed similarly to create merge conflicts between files in your local and remote repositories

Resources

- Setting up a team repository (part 1):
<https://medium.com/@kihoon.sohn/git-busters-how-to-set-your-team-repository-right-in-github-458bc49fa7cc>
- Working with a team repository (part 2):
<https://medium.com/@kihoon.sohn/git-busters-how-to-make-your-team-repository-errorless-in-github-1142cbe607be>
- Git Branching Explanation: <https://www.javatpoint.com/git-branch>
- GitHub Docs - About Branches:
<https://docs.github.com/en/github/collaborating-with-issues-and-pull-requests/about-branches>
- How Pull Requests Work:
<https://www.atlassian.com/git/tutorials/making-a-pull-request>
- GitHub Docs - About Pull Requests:
<https://docs.github.com/en/github/collaborating-with-issues-and-pull-requests/about-pull-requests>
- GitHub Docs - Merging Pull Request:
<https://docs.github.com/en/github/collaborating-with-issues-and-pull-requests/merging-a-pull-request>
- Git/GitHub Cheat Sheet (in the breakfast-hour repo)

