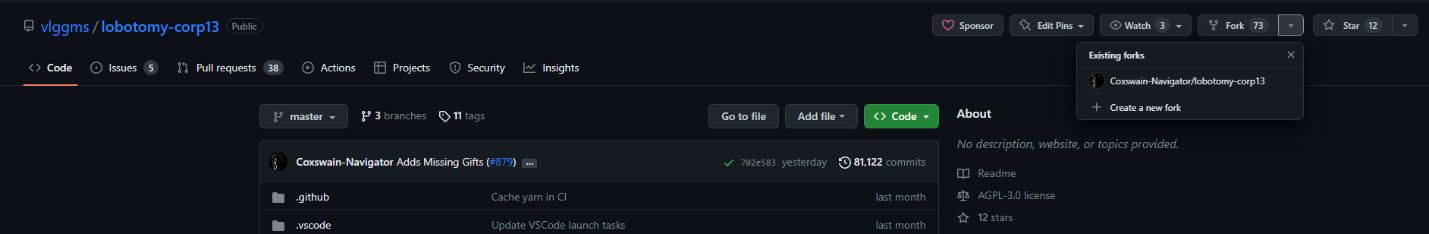
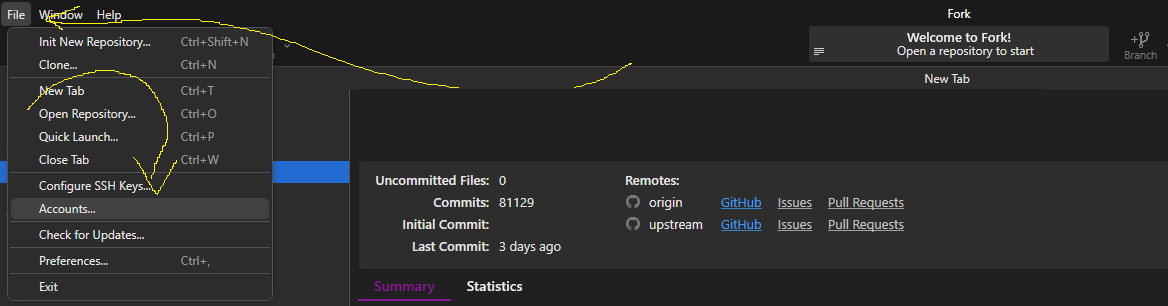
Written by Coxswain#4638

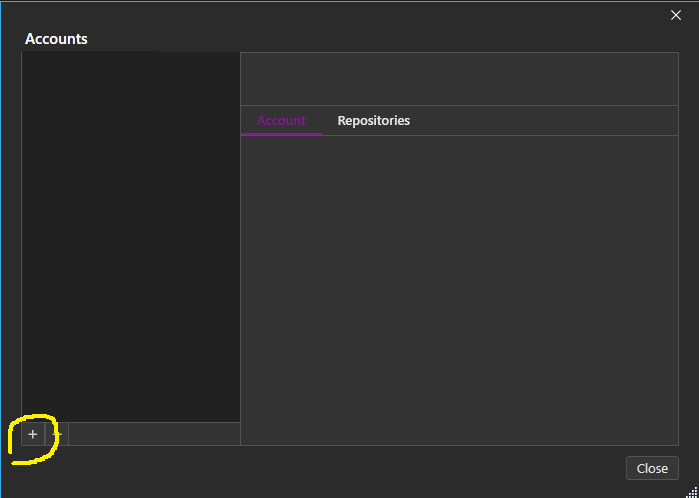
**So, you want to make a pull request?**

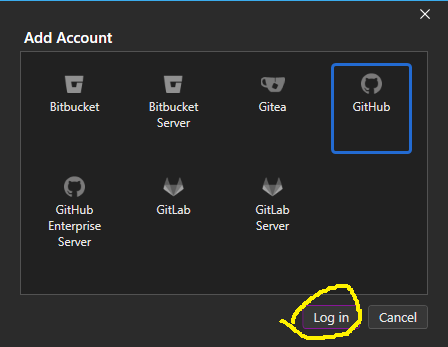
1. First you need to create an account on GitHub. This should be self-explanatory.
2. Navigate to https://github.com/Coxswain-Navigator/capstone-B, which is our repository.
3. Create your own “fork” of the repository. It should look something like this.



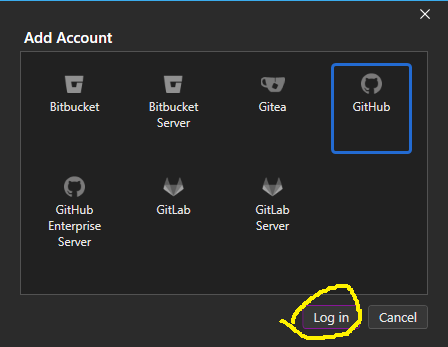
1. Download Fork at <https://git-fork.com/>. It has an infinite free trial.
2. Before you can do anything with fork, you need to connect your github account.

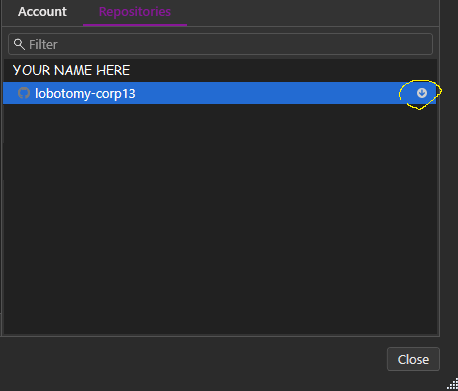




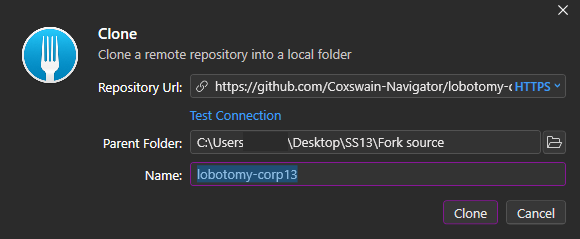


1. Once logged in you will need to open up your repository in fork.

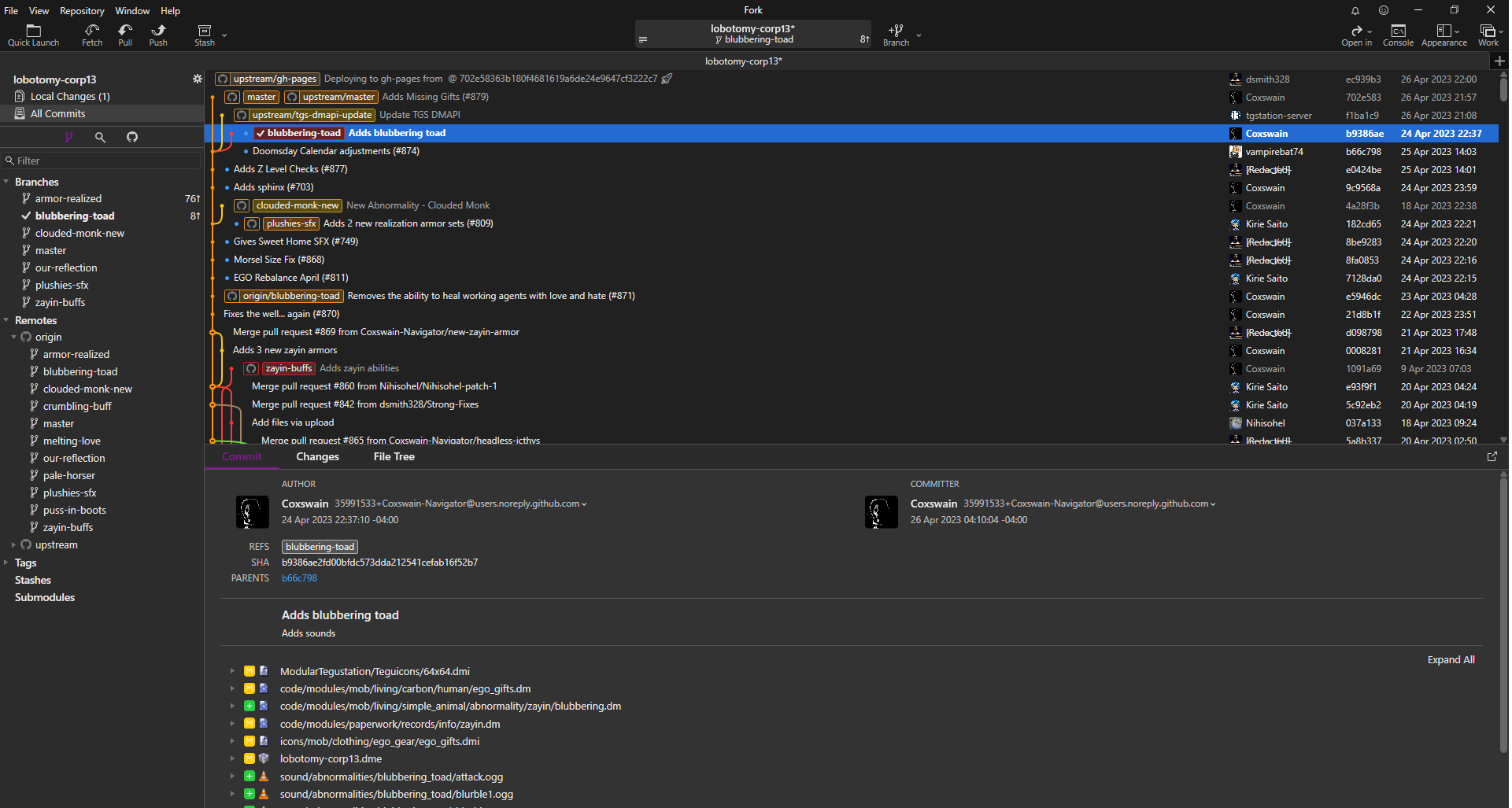




1. From here you can decide how to clone your fork, often the simplest and easiest approach works best. Make sure you choose a file location that is suitable for you. If you are having trouble, it is not very difficult to find solutions on a search engine. For most, hitting clone right away will work just fine.



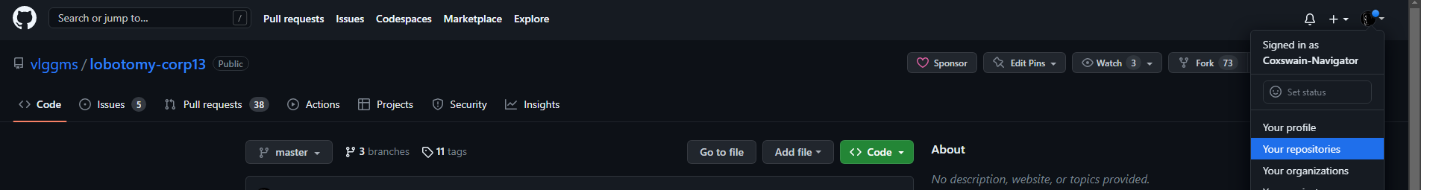
1. From here on out you should now have a functioning repository from which you can write or remove code and make pull requests. You should have the entirety of our repo in the parent folder that you designated in the previous step, and fork should look something like this.



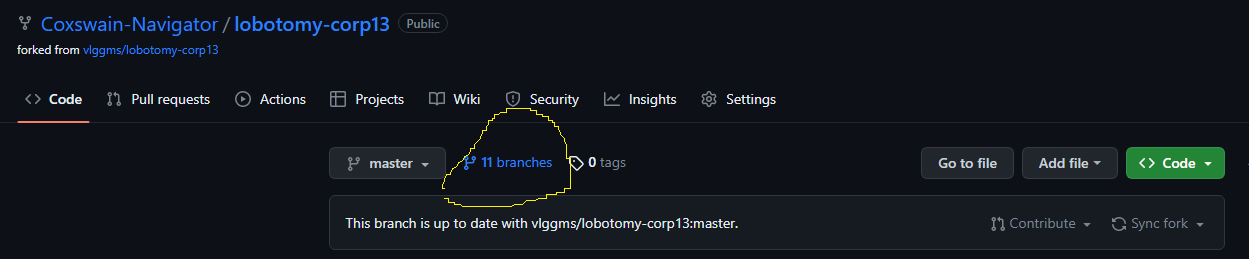
1. The first step in making a pull request is creating a branch. If you know when to make a commit on the master branch, you probably don’t need to be reading this guide. For now, just make sure you never do it. A branch can be created through a variety of means, both GitHub and fork support creating branches. The following steps will display a foolproof method on creating a new branch.

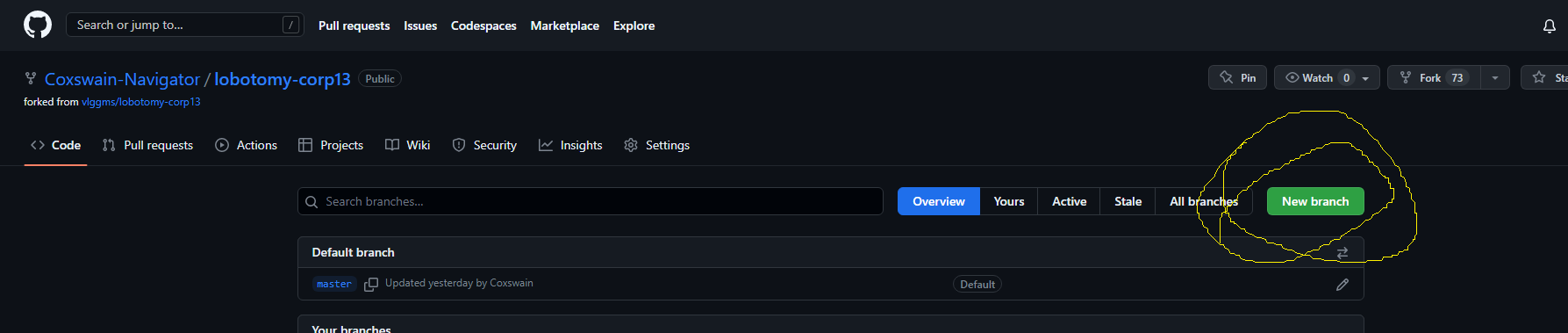
**Creating a new branch through github**

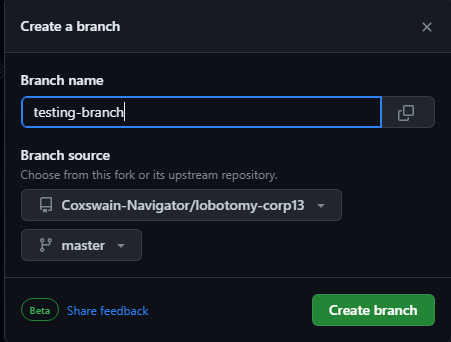
1. Navigate to https://github.com/Coxswain-Navigator/capstone-B
2. Go to your fork of our repo. The following image will show you how to navigate to it.



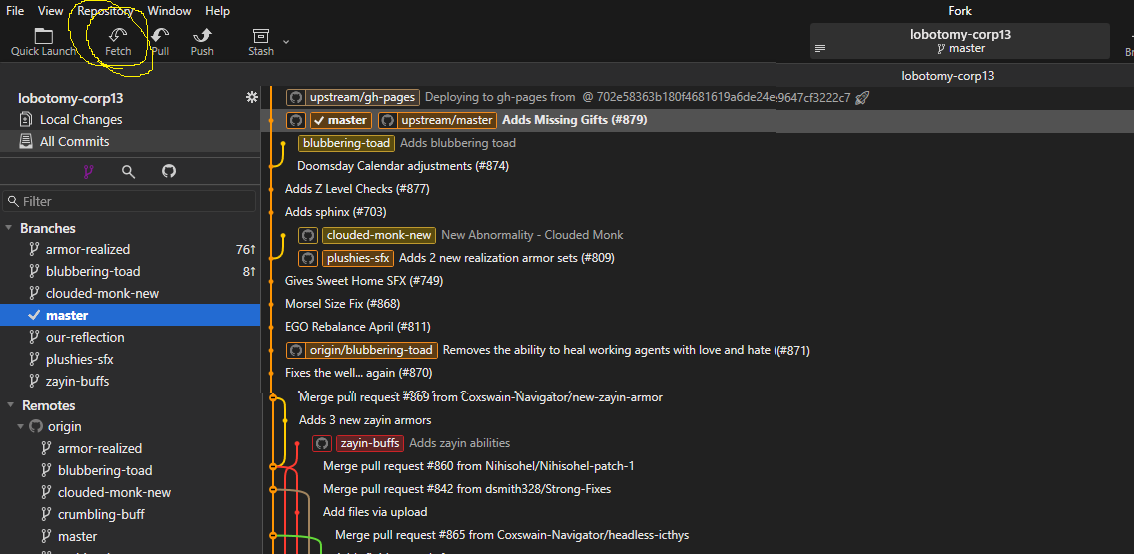
1. Click “branches” and create a new branch. Branches typically following the naming scheme of “new-thing” or similar. It will not show up in fork until you complete the next step.







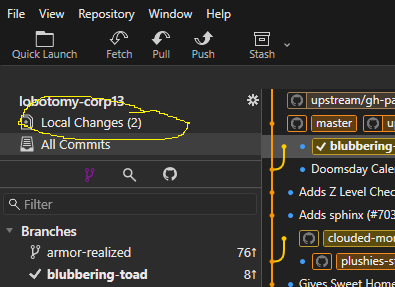
1. Your branch is now created, but why can’t you see it in fork yet? This is because your fork of github branch is a “remote” which is not yet in your file system. It needs to be “Fetched” through fork. Luckily, this is as simple as pressing a button.
2. First, return to Fork. Check out the “master” branch and hit the fetch button as displayed in the following screenshot. “Master” should be your only branch displayed in fork at this time. The new branch will show up under remotes. Make sure that “origin” is open under remotes.

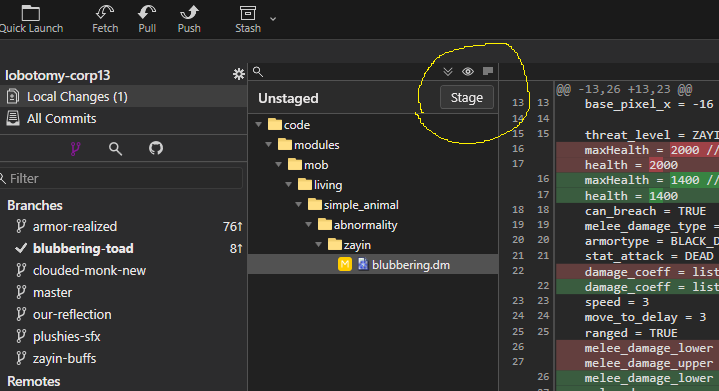


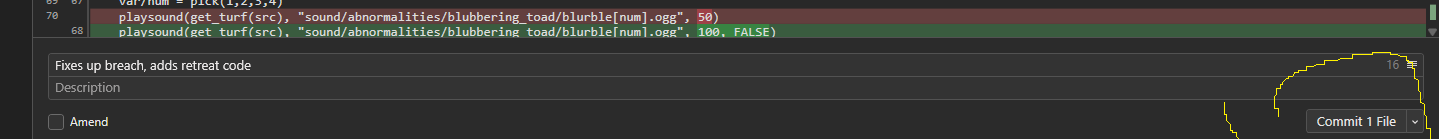
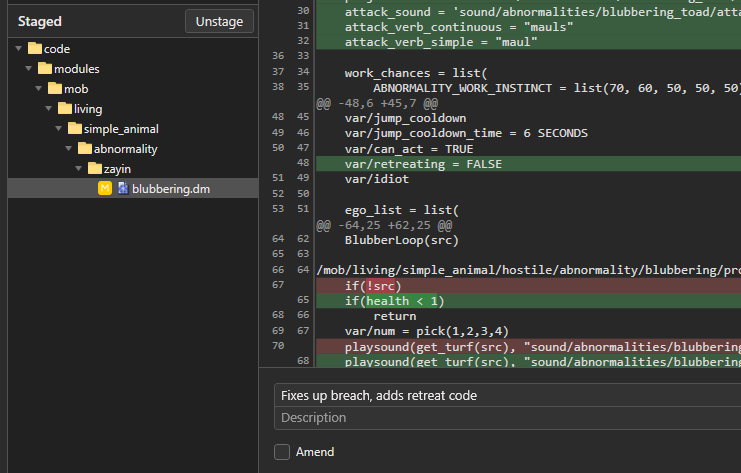
1. You can now check out the branch by simply clicking on it twice, which will allow you to start tinkering with code without worrying about your local changes ending up in your master branch. At this point, you can actually start coding! I **HIGHLY** recommend making a very small first pull request in case anything goes wrong.
2. This is a good time to set up your upstream if you haven’t already. Follow this guide. <https://medium.com/@topspinj/how-to-git-rebase-into-a-forked-repo-c9f05e821c8a> Our repo link is <https://github.com/vlggms/lobotomy-corp13> If you are confused about how to use the command line, look further down the guide.

**I’ve made my code, now what?**

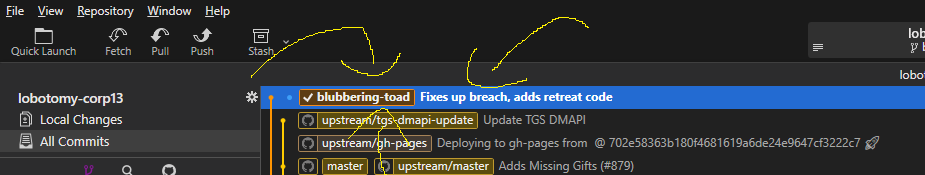
1. When working with fork, there are two major things to concern yourself with – commits and branches. You should have figured out what a branch is by now, so now you will be making a commit. A commit is a snapshot of the code. Usually, you want to keep commits squashed regularly which combines them into one and keeps things organized. By doing this, each branch will more or less be a different snapshot of the code and thus you can manage multiple projects at once with ease. In this section we will go over the management of commits and making a pull request. In the next section, we will go over handling conflicts and I highly recommend reading that over before actually making a pull request.
2. After making your changes to the code, open up Fork. The changes will show up under “local changes” You must stage the changes you want to keep and discard the ones that you don’t want to keep. Discarded changes will be LOST, so keep that in mind. The following screenshots will guide you.





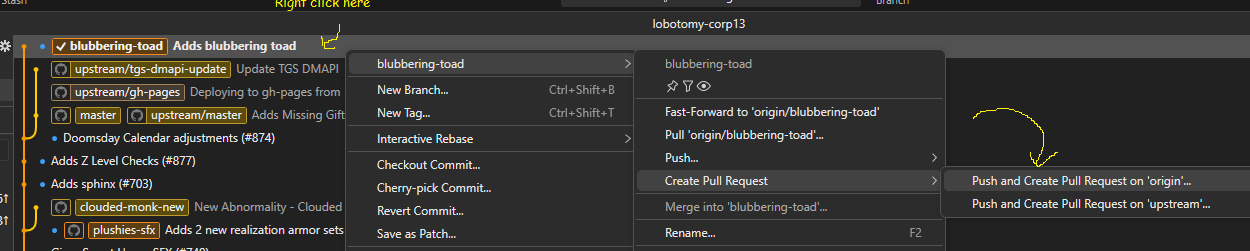


1. After you have sorted out your desired changes, write a commit message detailing commit and commit. Your new commit will show up on the main page.



**How do I make a pull request?**

1. Making a pull request with fork is surprisingly easy, pull request guidelines will not be within the scope of this guide but you can look at any existing pull request to get an idea of the expected standard. First, open up Fork.



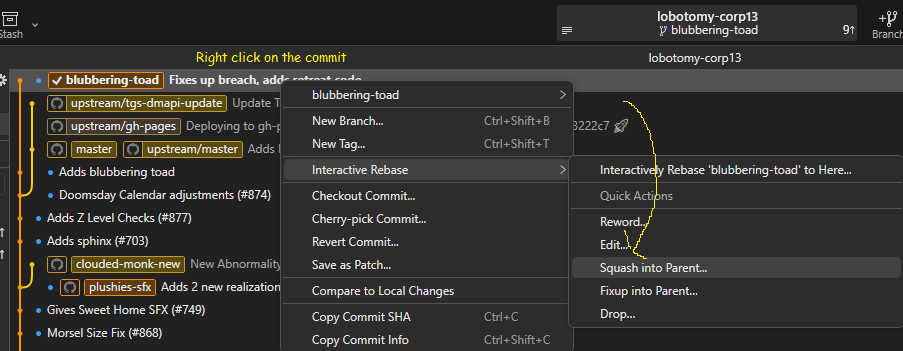
1. A GitHub page with the relevant pull request information should open up right away. If there are issues, check your fork account settings and your GitHub repository for inconsistencies with this guide.

A screenshot of a computer

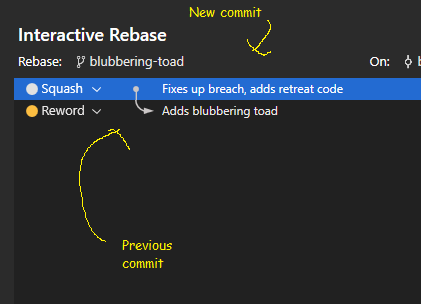
Description automatically generated

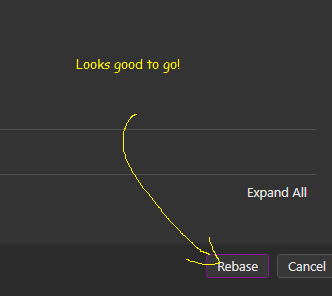
**How do I squash Commits?**

1. You’ve now got your first commit, but what If you want to make some changes to that commit? You simply go through the process again, and you will now need to “squash” your commits. Here’s how.



1. Make sure you check the commits before you squash! If one of those commits is NOT yours, do NOT commit! You may be trying to squash without a previous commit on the same branch, or some other issue has arisen! In this case, my previous commit is here so I am good to squash by hitting rebase!

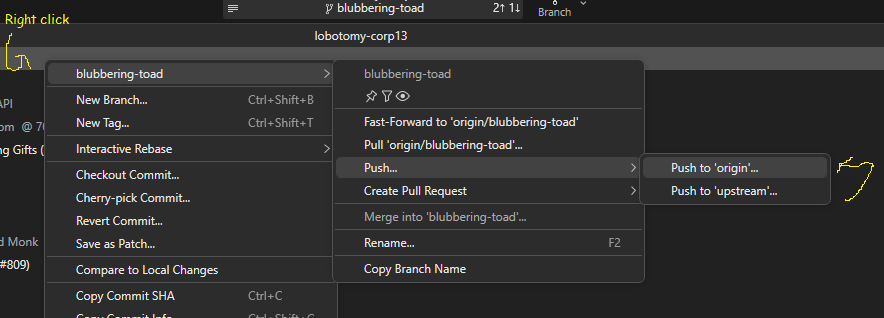




1. If everything worked out well, you should see your commits merged into one, with the name of the initial commit you created.

**How do I update my pull request?**

1. You’ve squashed commits but your pull request is still outdated? Now you need to PUSH your changes. It’s actually pretty simple.

****

**A screenshot of a computer

Description automatically generated with medium confidence**

1. Make sure you check “force push” in order to update your pull request!