Université Paris Cité

Git : Part 3
Collaborate on GitHub

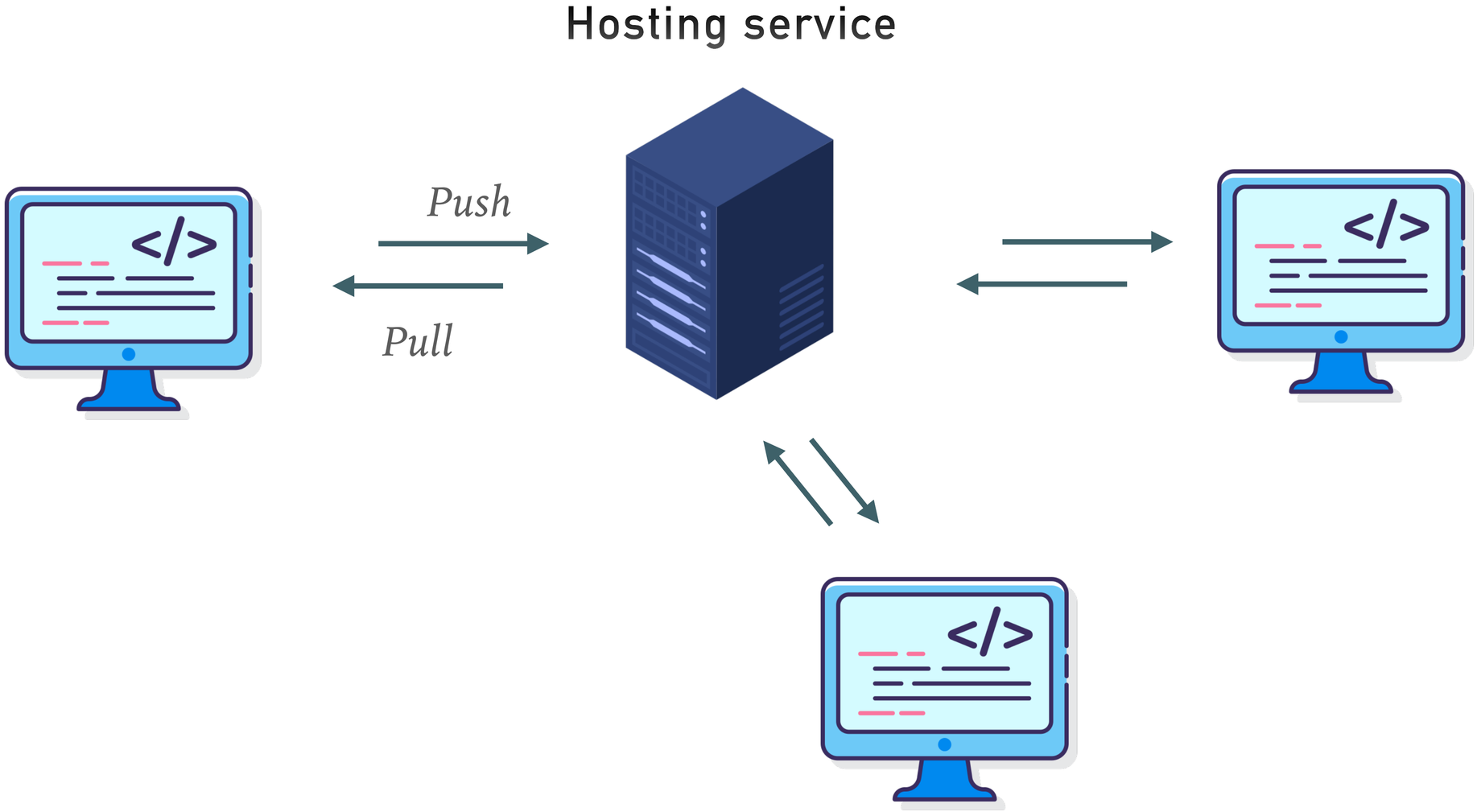
Amine FERDJAOUI
amine.ferdjaoui@sogeti.com

Année universitaire 202/2025

What is Github ?

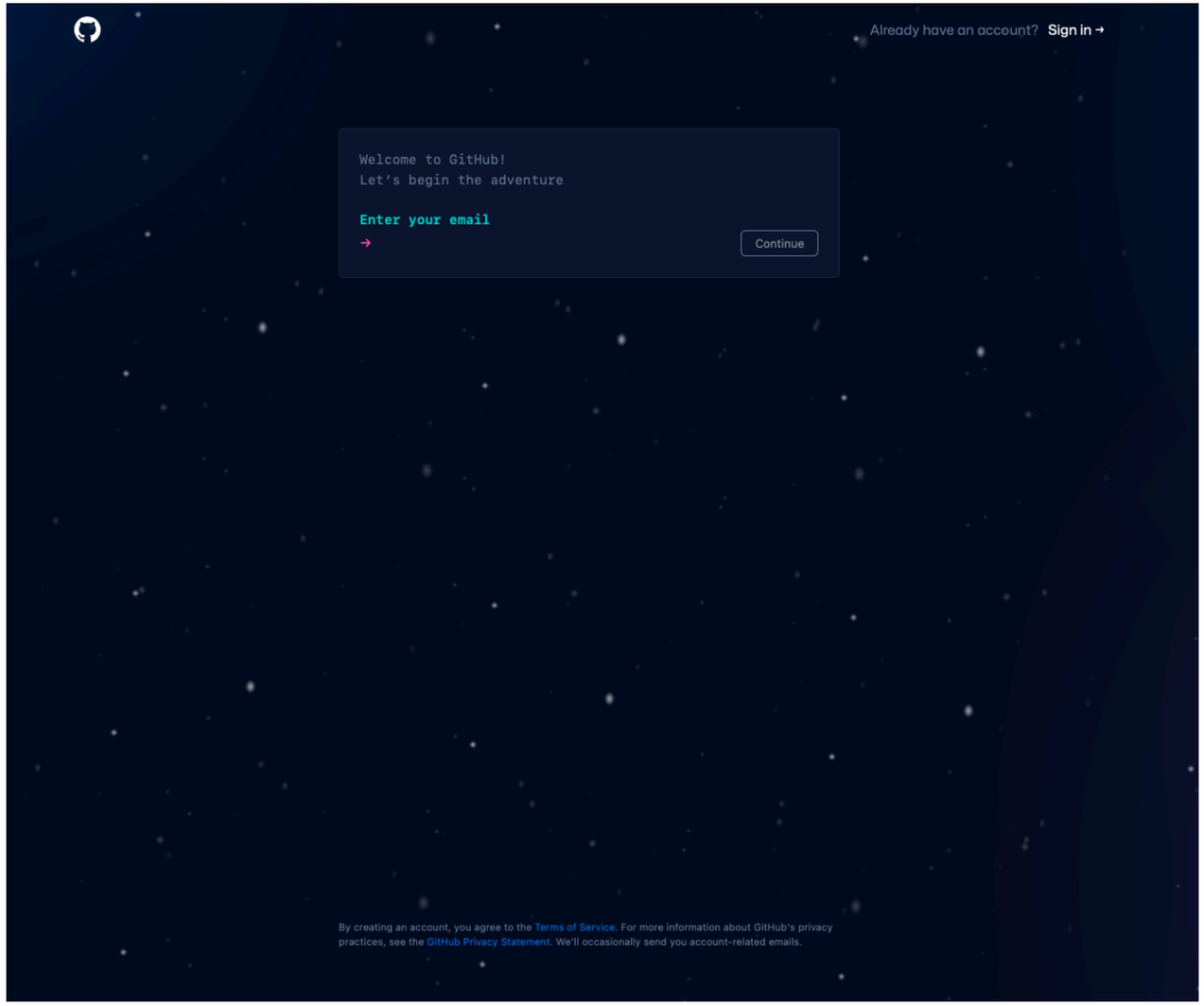
- Web hosting service that hosts git projects on the cloud
- A platform for hosting and collaborating on Git repositories
- The most used ones are : Github, bitbucket and gitlab.
- **Github** is the most popular one, it was bought by Microsoft in 2018

What is Github ?

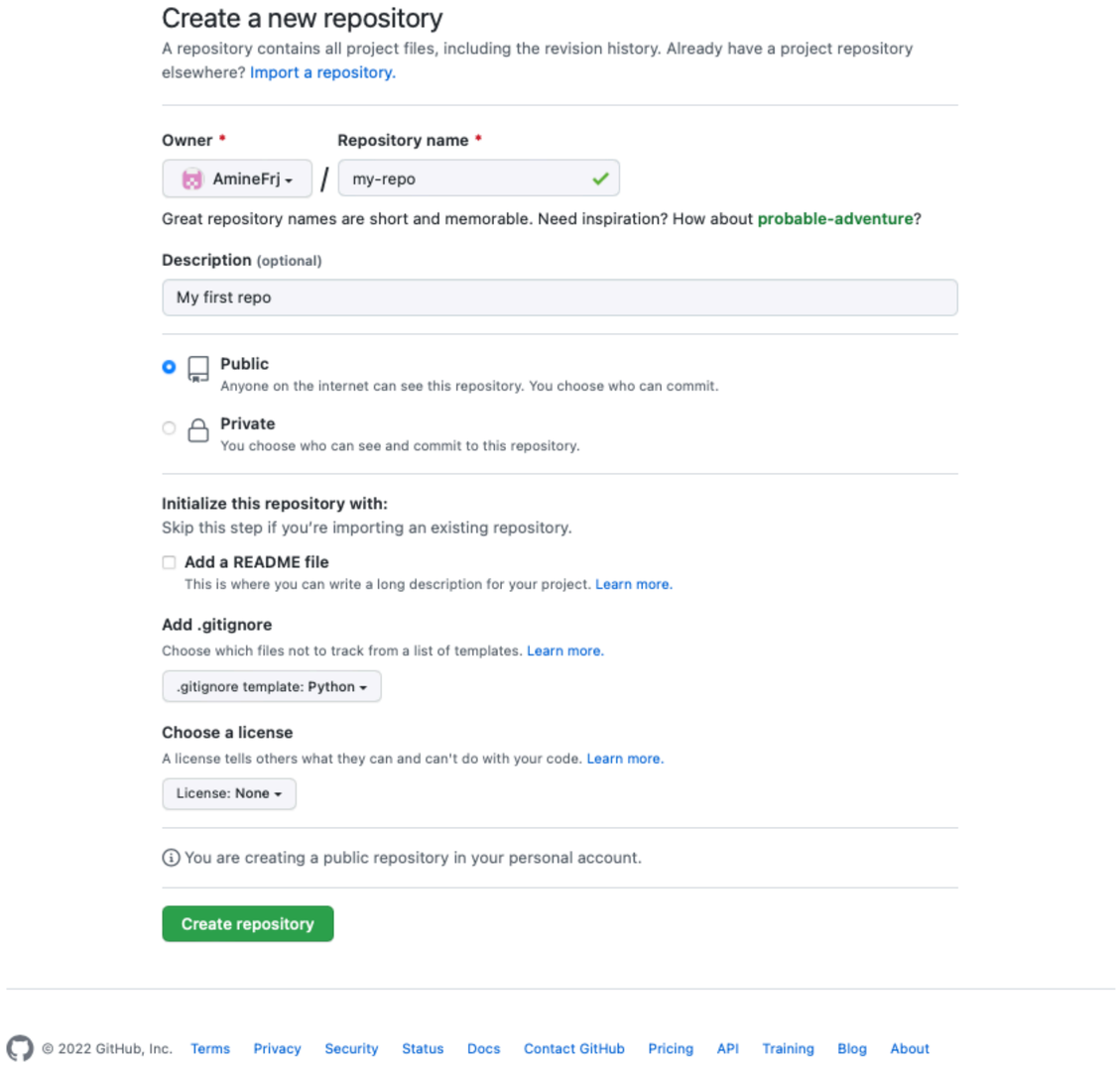


GITHUB SETUP

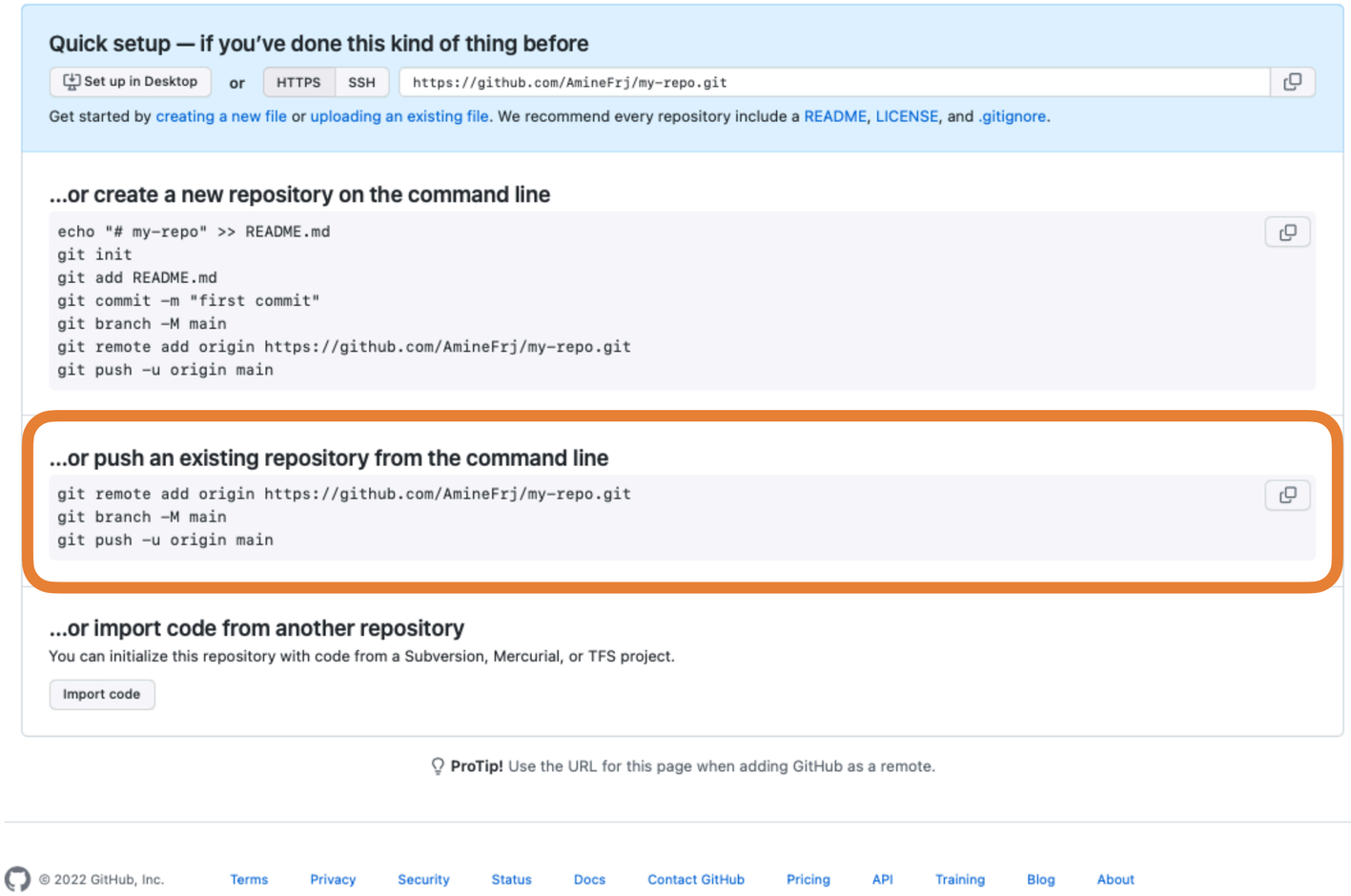
Sign up to GitHub



Create a new repository

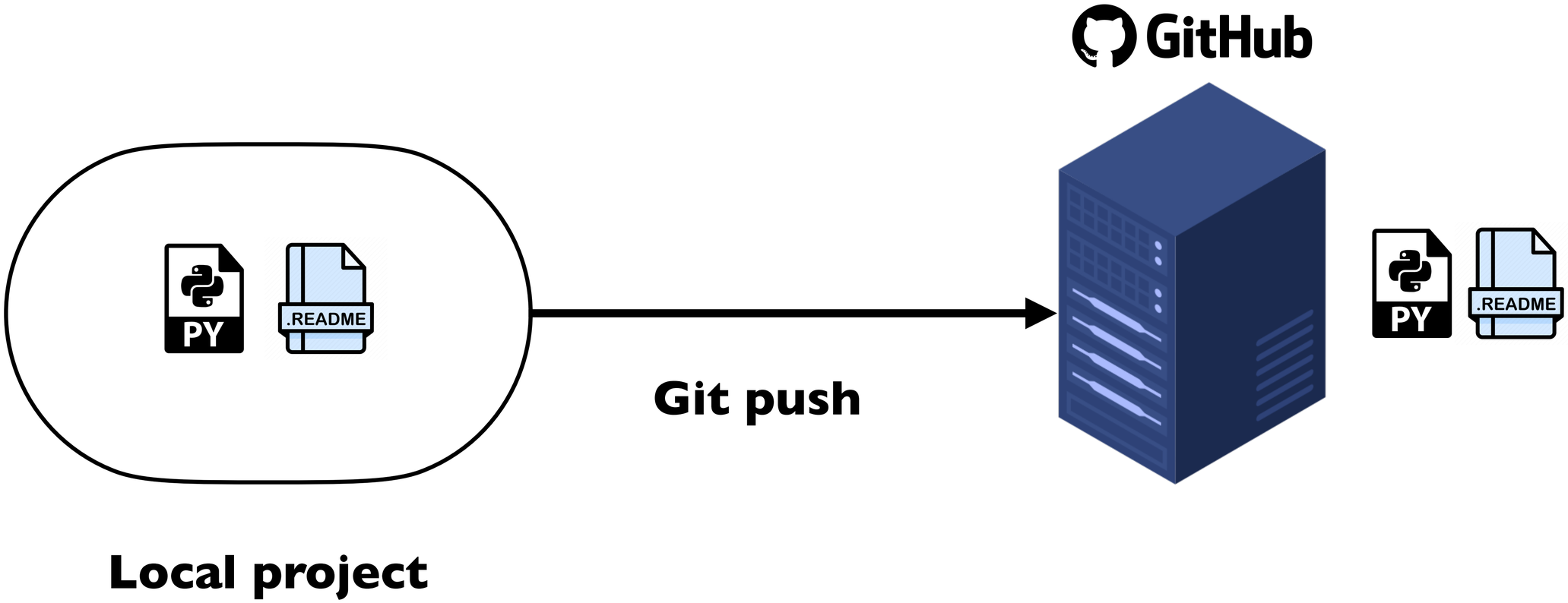


Initial setup



What is a push

Git push sends projects into repository in the cloud

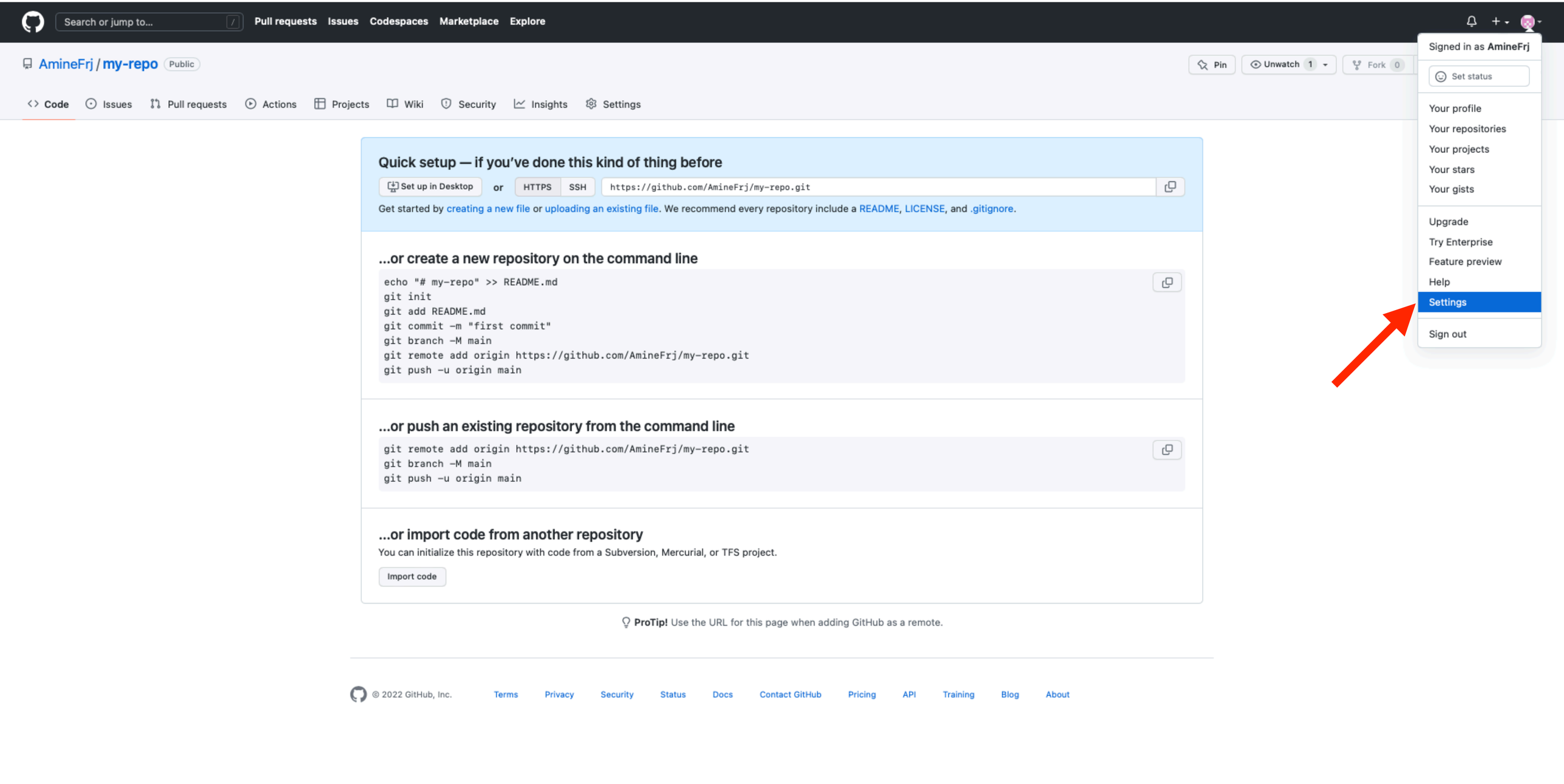


Setup ssh key

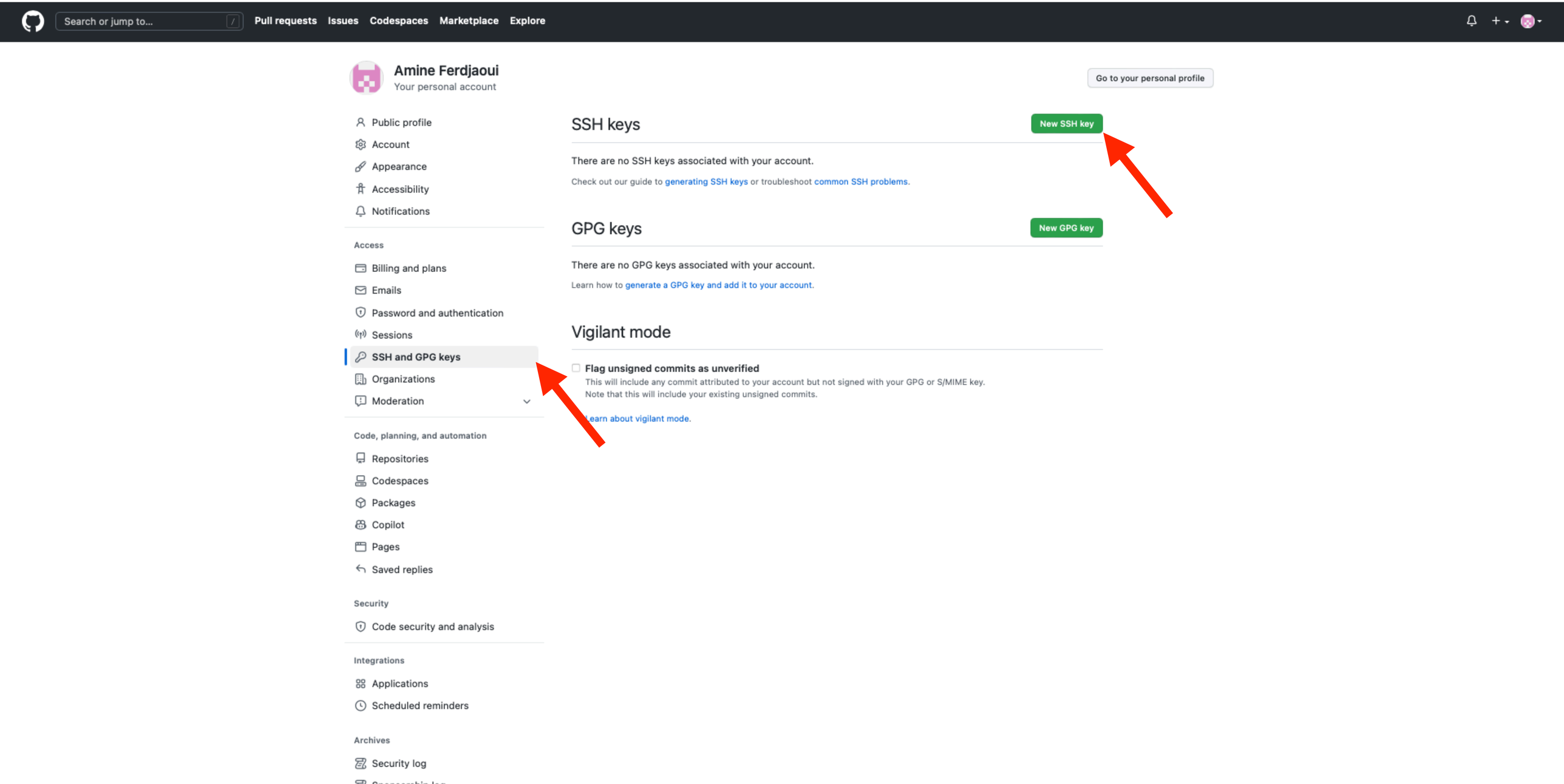
- Since august 2021 GitHub password authentification has been removed and replaced by ssh keys, so you don't have to enter your password multiple times.

```
macbook-pro-de-amine:hello git amine$ git push -u origin main
Username for 'https://github.com': AmineFrj
Password for 'https://AmineFrj@github.com':
remote: Support for password authentication was removed on August 13, 2021.
remote: Please see https://docs.github.com/en/get-started/getting-started-with-git/about-remote-repositories#cloning-with-https-urls for information on currently recommended modes of authentication.
fatal: Authentication failed for 'https://github.com/AmineFrj/my-repo.git/'
```

Configure SSH keys



Configure SSH keys



Setup ssh key

- First check if you have no SSH keys on your computer

```
$ ls -al ~/.ssh
```

- SSH keys supported by GitHub are :
  - id_rsa.pub
  - id_ecdsa.pub
  - id_ed25519.pub
- If you already have one just copy it into your GitHub settings, otherwise follow the instructions : https://docs.github.com/en/authentication/connecting-to-github-with-ssh/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent

Setup multiple ssh keys

- If you have multiple accounts, make sure to add specific key for each one in `~/.ssh/config`

then add ssh private keys to your agent with : `ssh-add ~/.ssh/<key-name>`

- Cf. https://gist.github.com/oanhnn/80a89405ab9023894df7

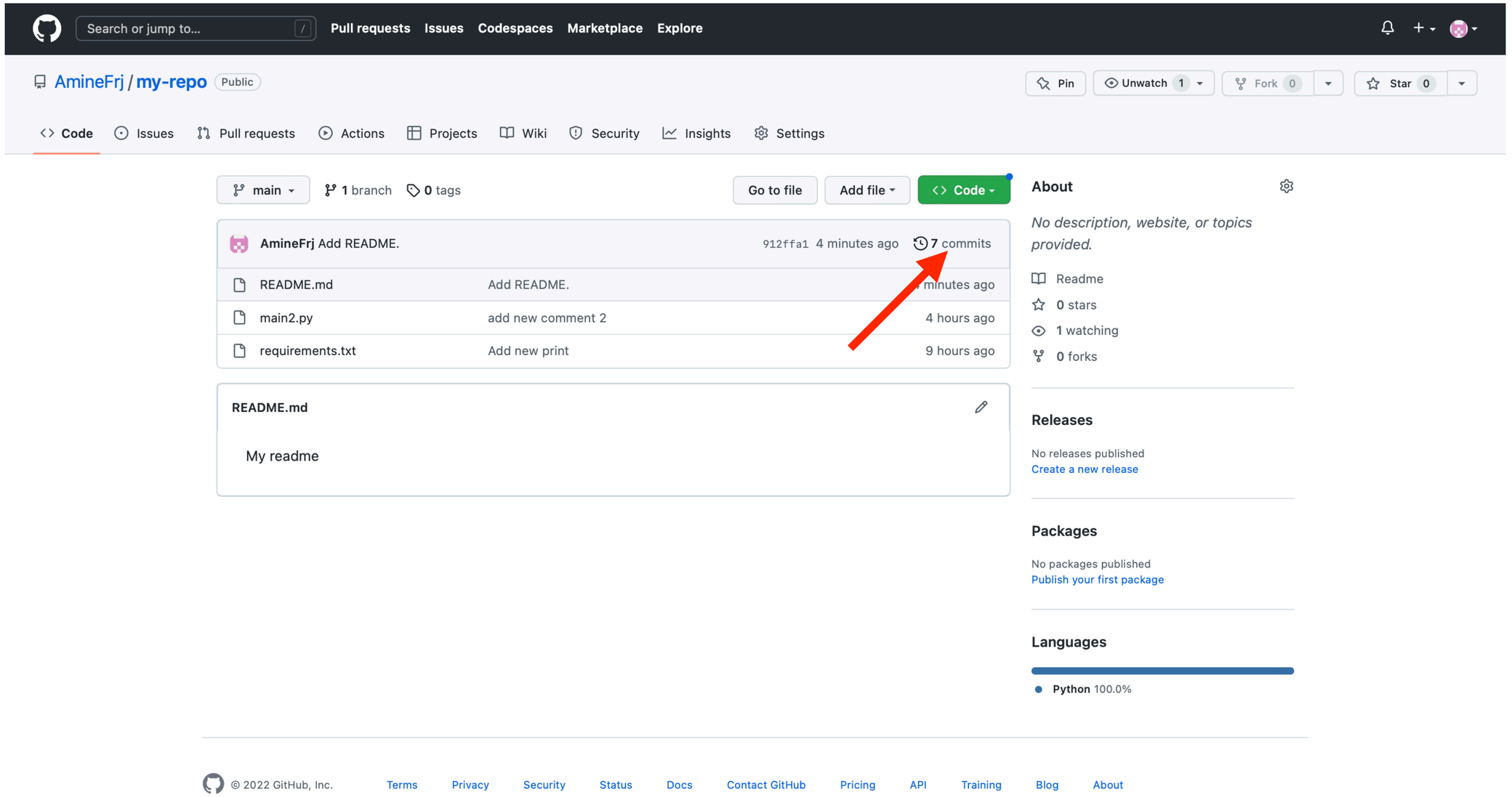
Initial push

```
$ git remote add origin git@<repo_url>
$ git branch -M main
$ git push -u origin main
```

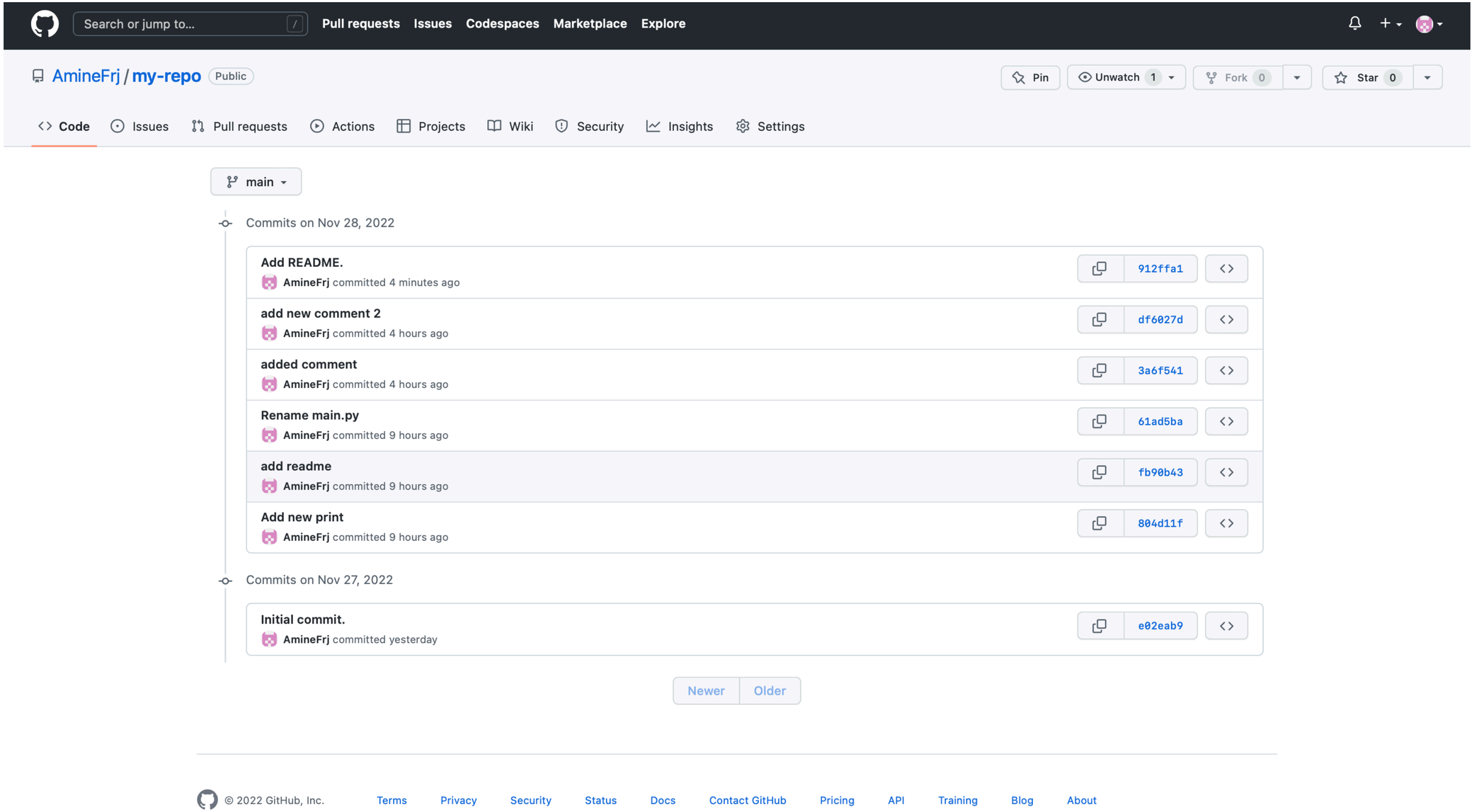
Nb. Use the ssh url in *git remote add origin*

```
$ echo "This repo is a python test on GitHub" > README.md
$ git add README.md
$ git commit -m 'Add a README.'
$ git push
```

Initial push



Initial push



Clone project

- Once you pushed the initial commit, you can clone it using this clone command

```
$ git clone <repo_url>
```

- You can see that git added the remote branches under origin

```
$ git branch -a
```

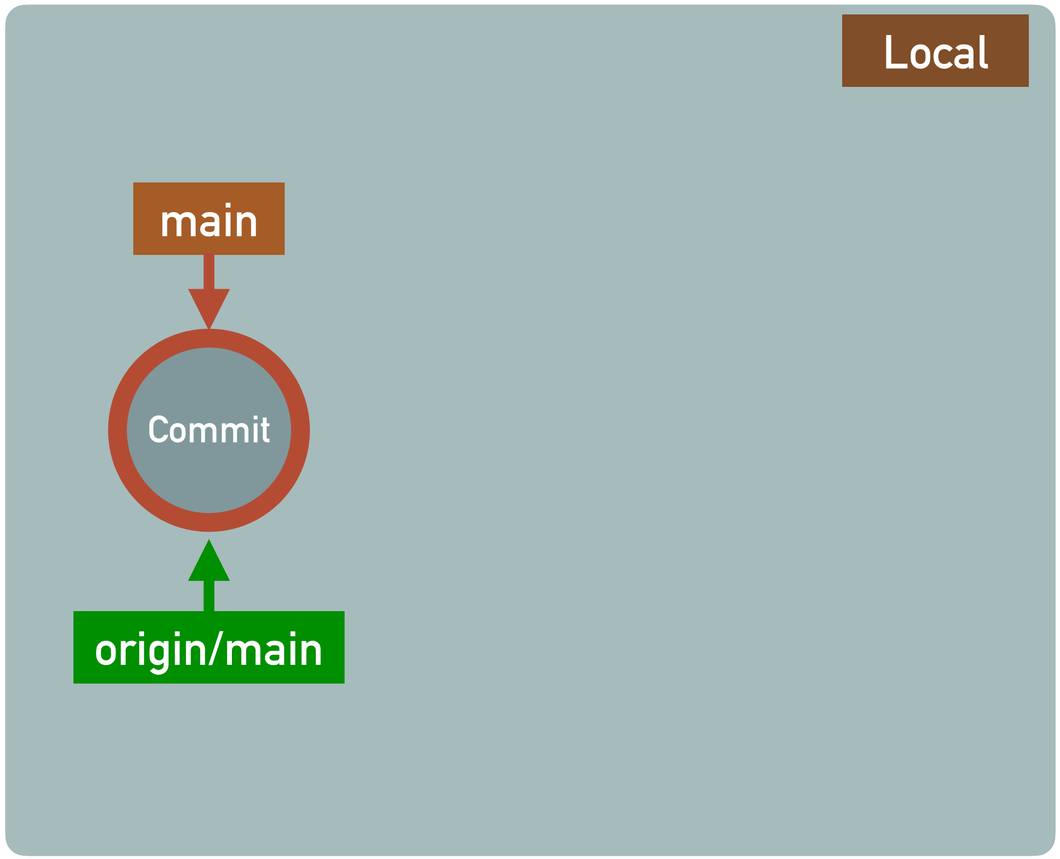
Git visualization

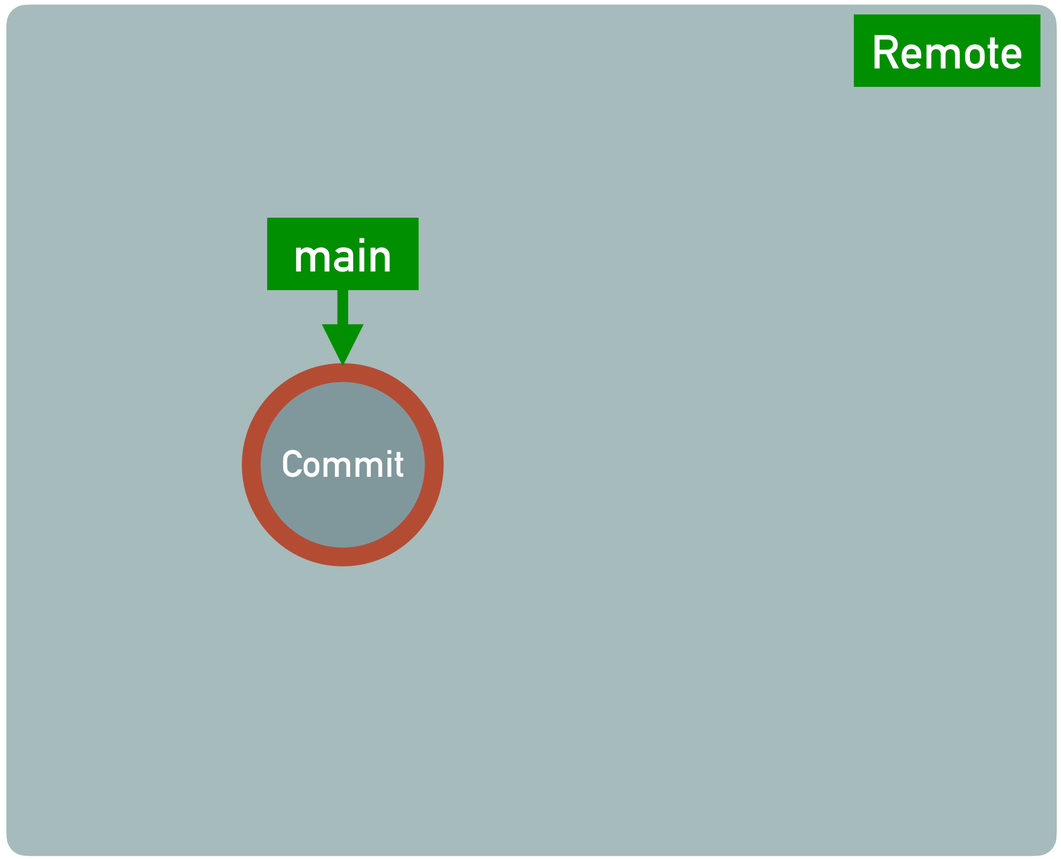
https://git-school.github.io/visualizing-git/

GIT FETCH & GIT PULL

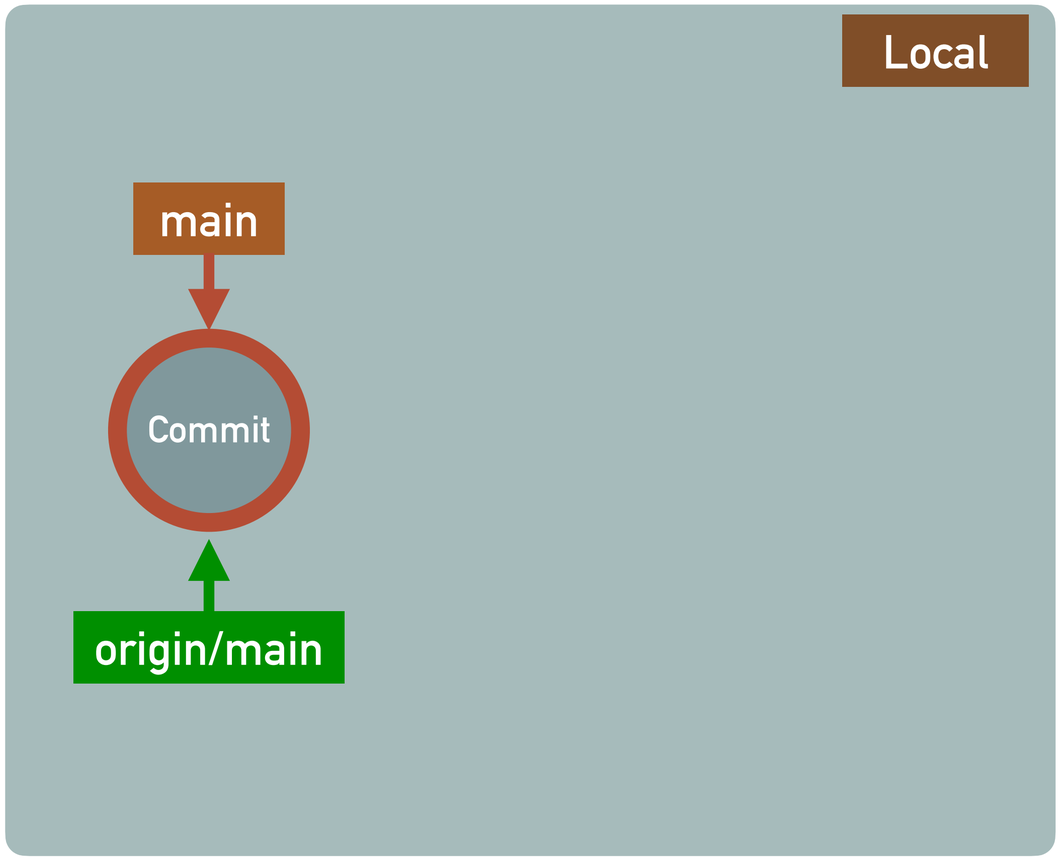
Fetching

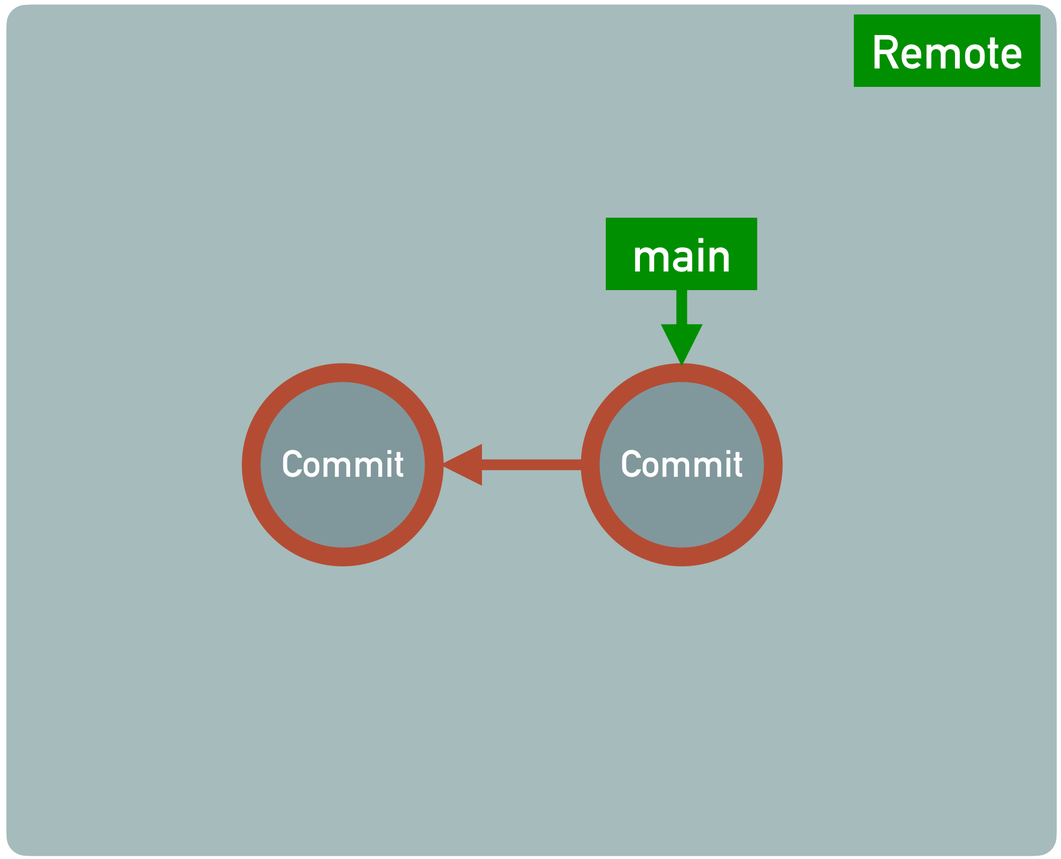
Your local repository is not automatically synchronised with the remote, you have to update the remote changes locally to have the latest version





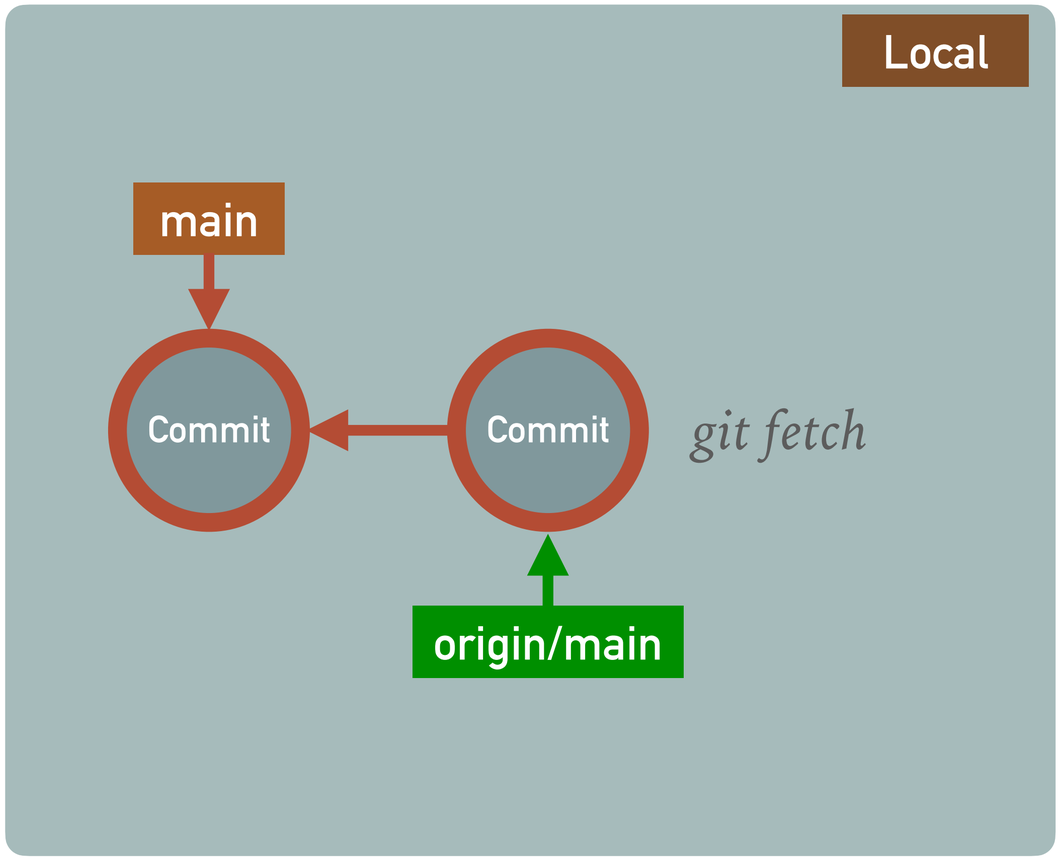
Fetching

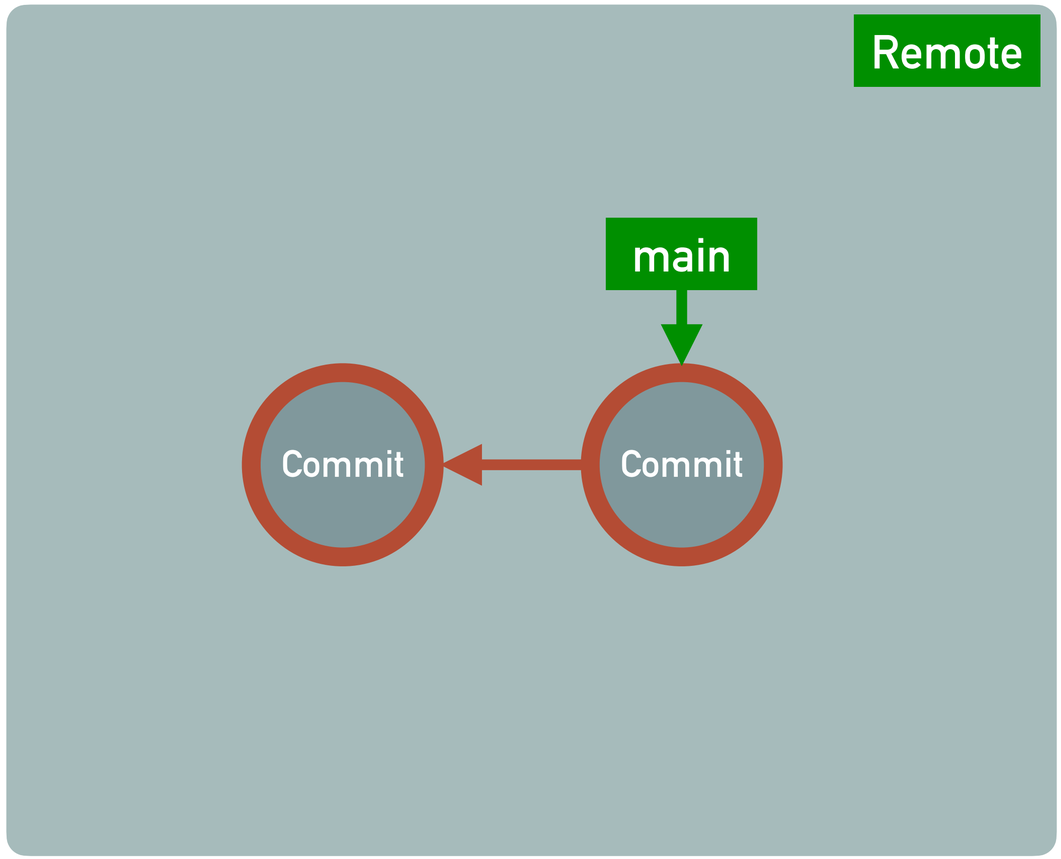




Fetching

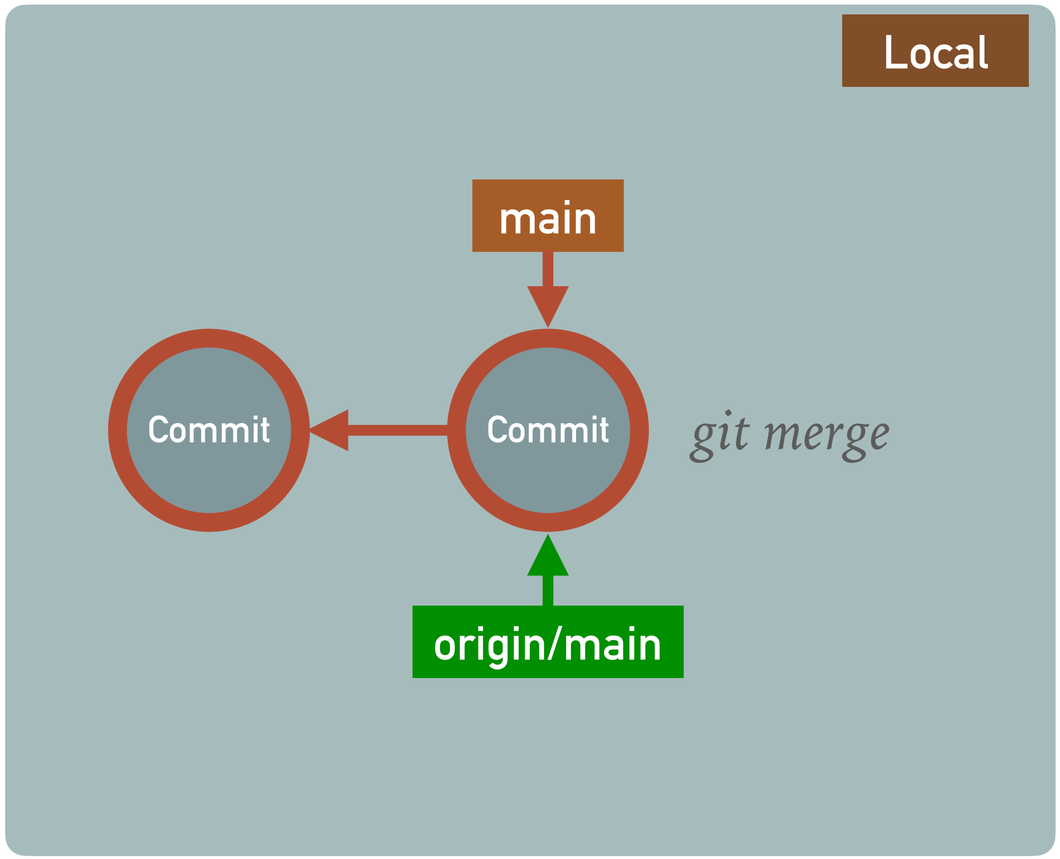
```
$ git fetch <branch-name> #get the remote changes locally
```

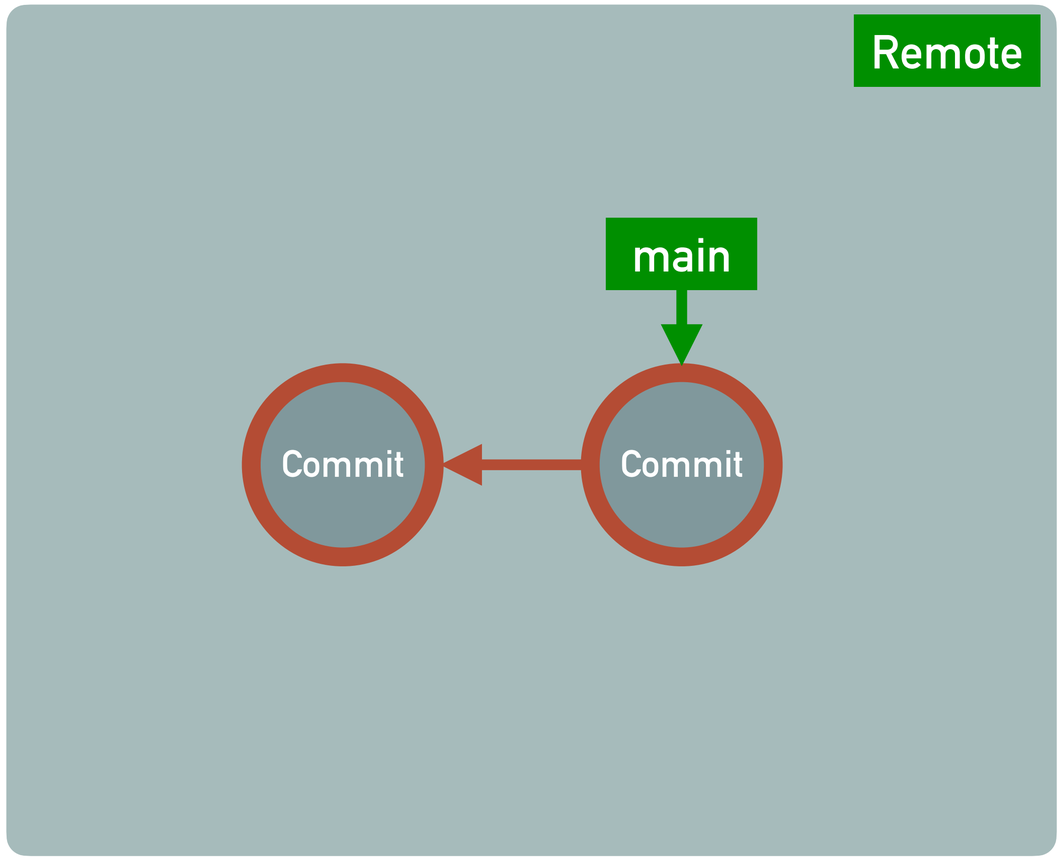




Fetching

```
$ git merge origin/main
```





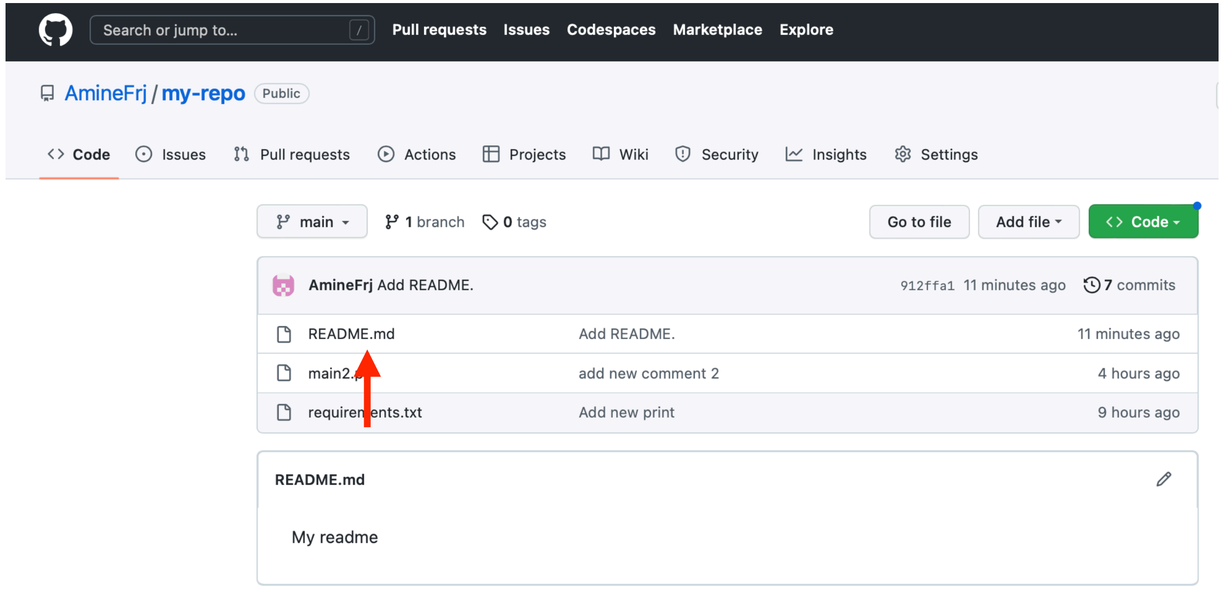
Git pull

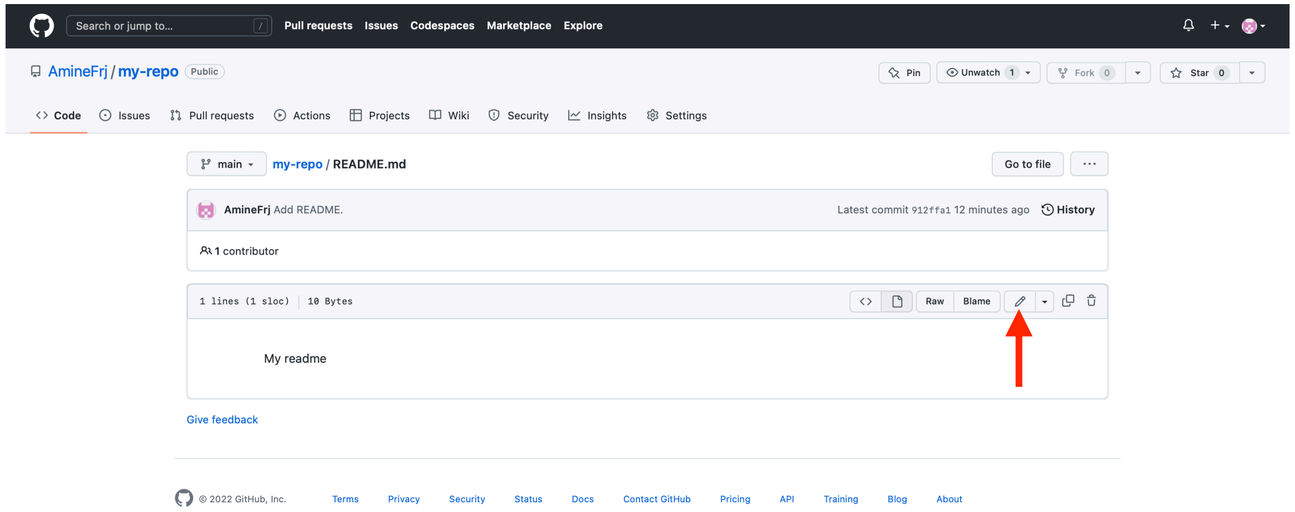
We can do both in a single command :
git pull = git fetch + get merge + create a new commit

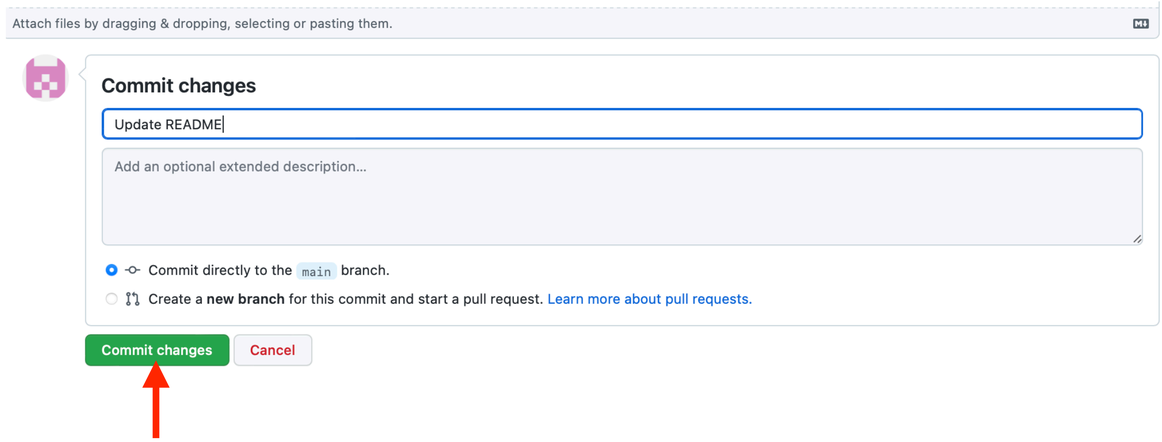
```
$ git pull
```

Git pull

- Simuler un push dans github en modifiant sur l'interface github comme suit :
- Faire un pull ensuite en local



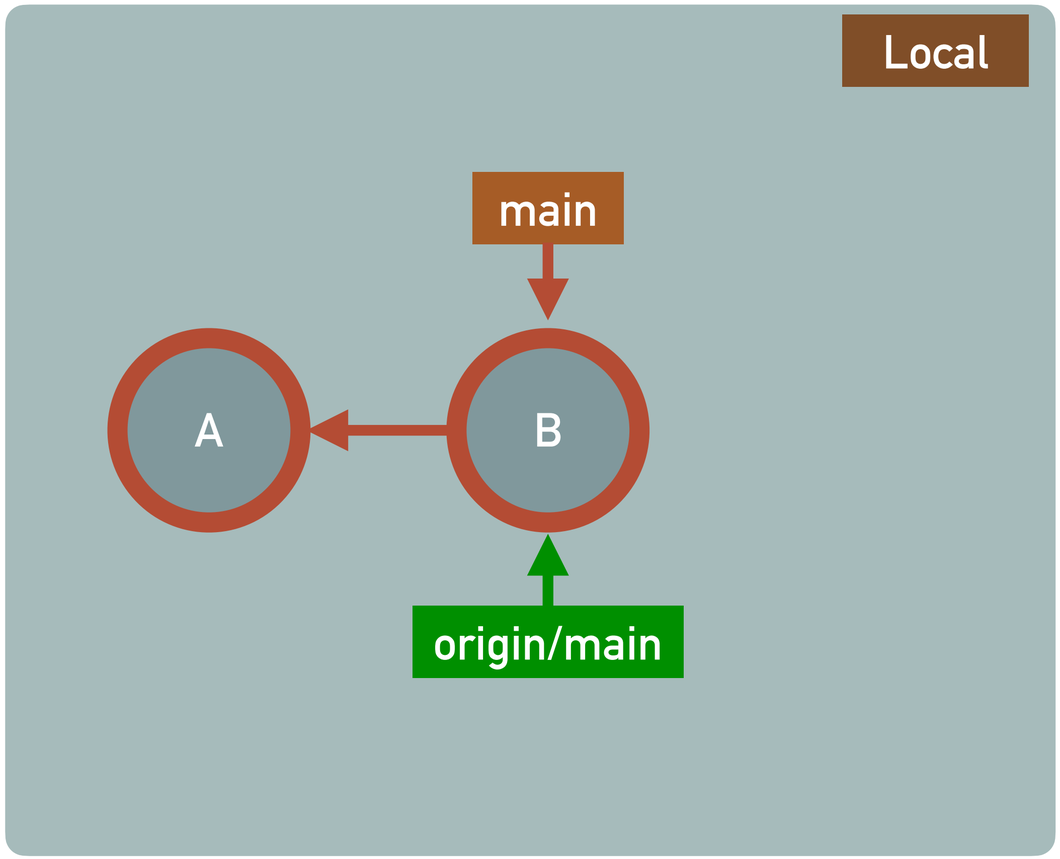


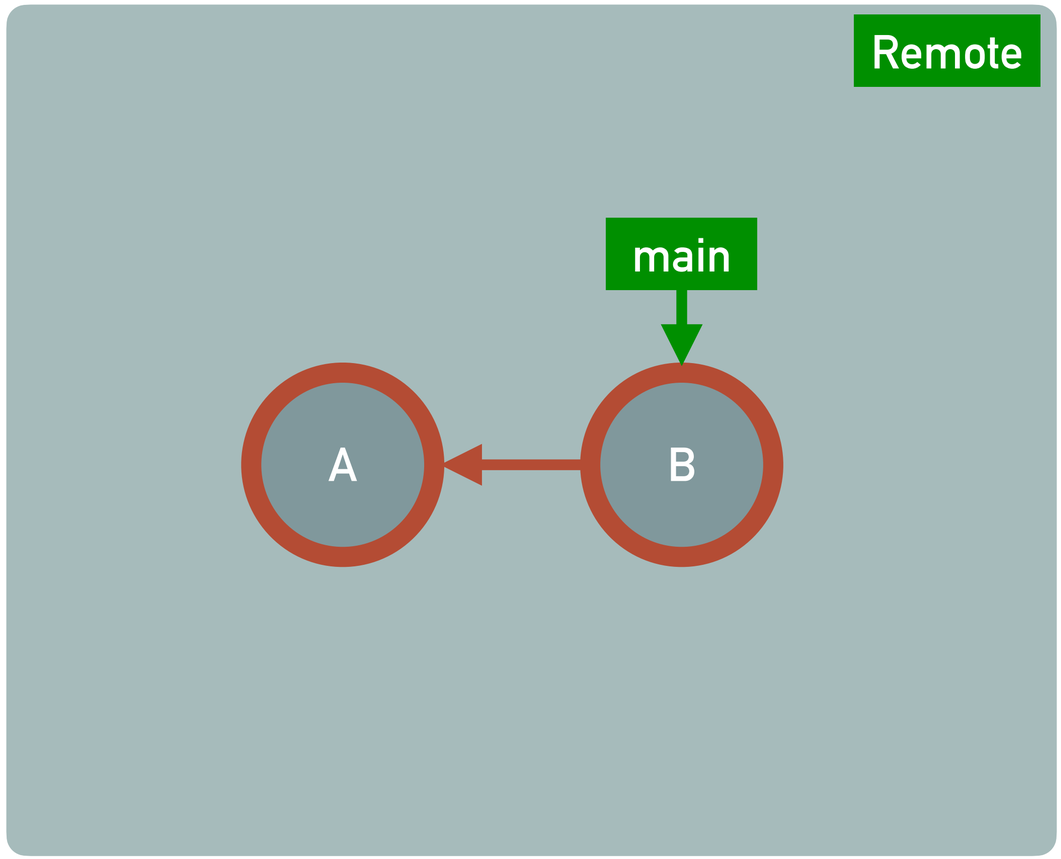


GIT PUSH

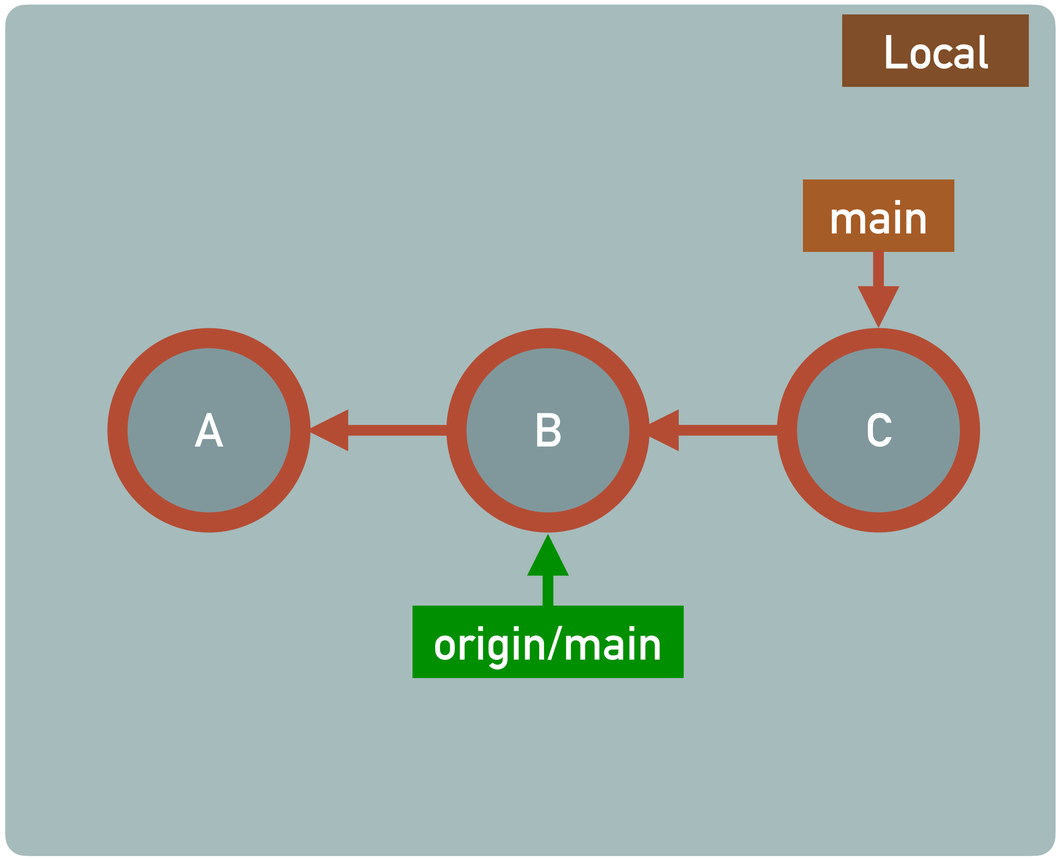
Push

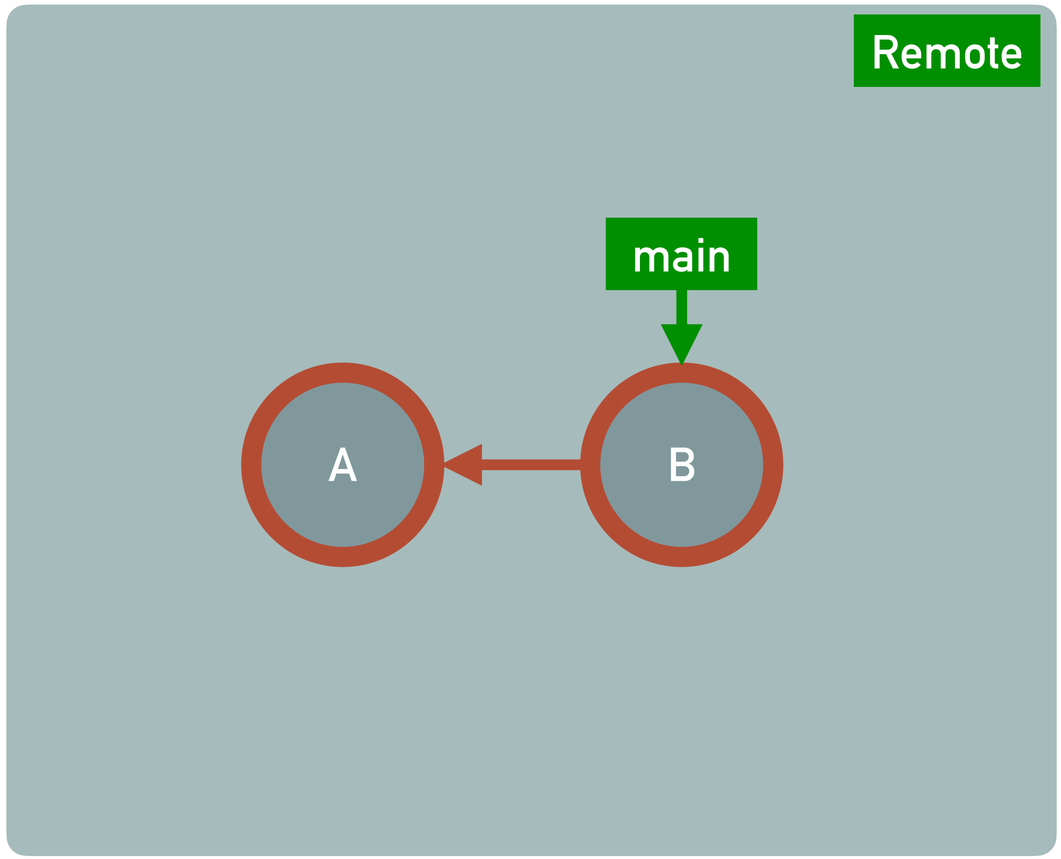
- Origin/main is considered as a separate branch
- Since it shows the remote state, it can only be modified with a *push*





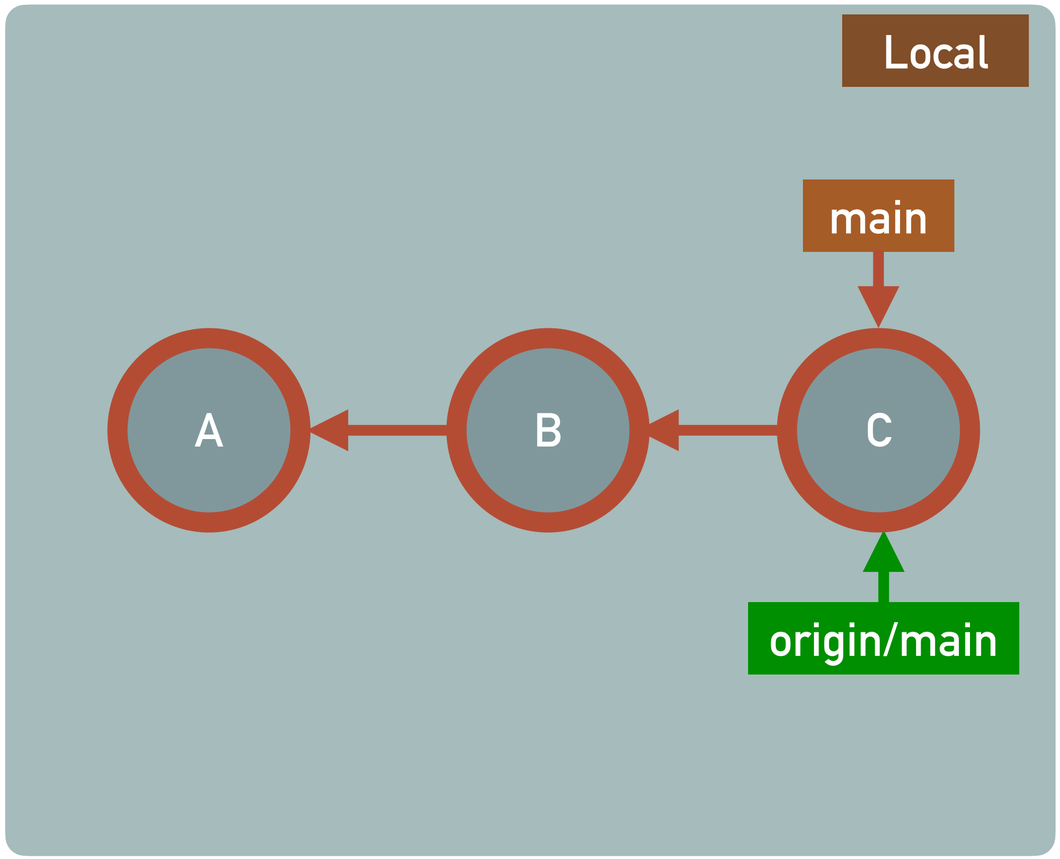
Push

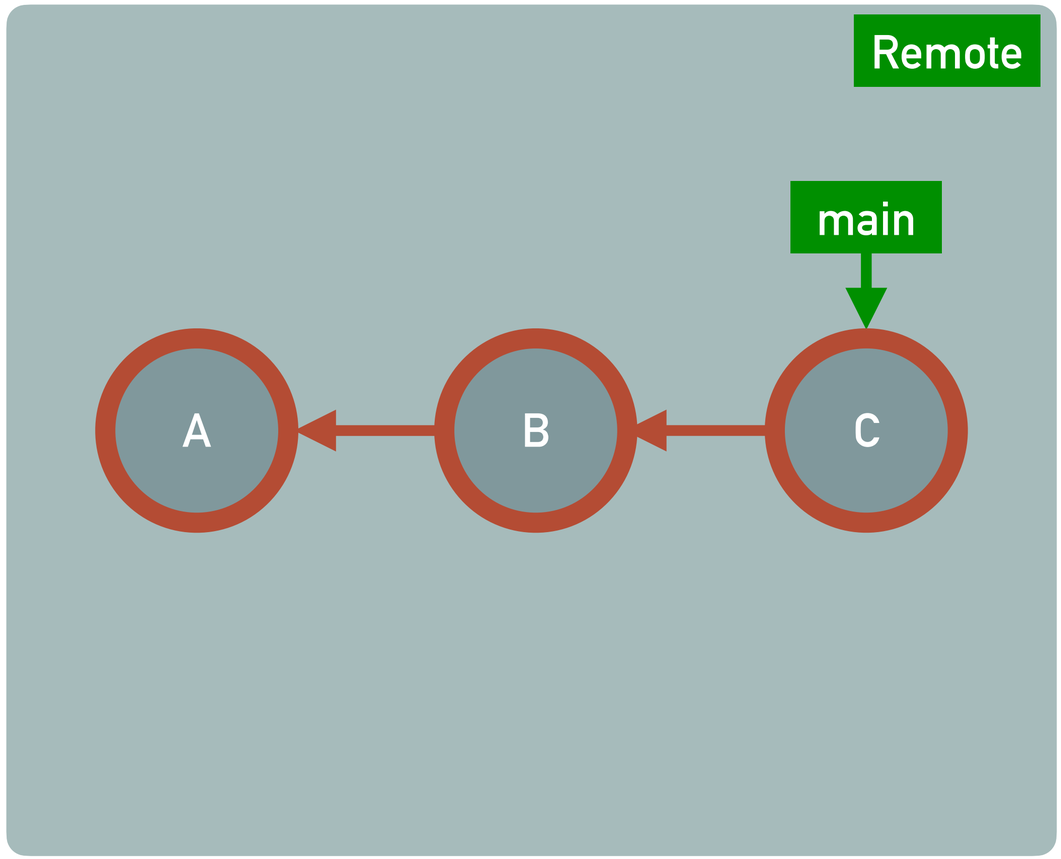




Push

```
$ git push origin <branch> #By default push into origin and your current branch
```





PUSH BRANCH TO REMOTE

Push a branch to remote

When creating a branch locally you have to push it to remote

```
$ git branch -C <branch-name>
$ git push -u origin <branch-name> #push the branch to remote
$ git branch -vv #check if local branches are synchronised with remote
```

Delete a branch (locally and remotely)

```
$ git push -d origin <branch-name> #delete the remote branch
$ git switch main
$ git branch -d <branch-name>
```

FETCH A REMOTE BRANCH LOCALLY

Setup a track remote branch

- In order to work on a new remote branch you have to fetch it and then create a new one locally and make it track the remote one

```
$ git fetch
$ git switch -C <branch_name> origin/<branch_name>
```

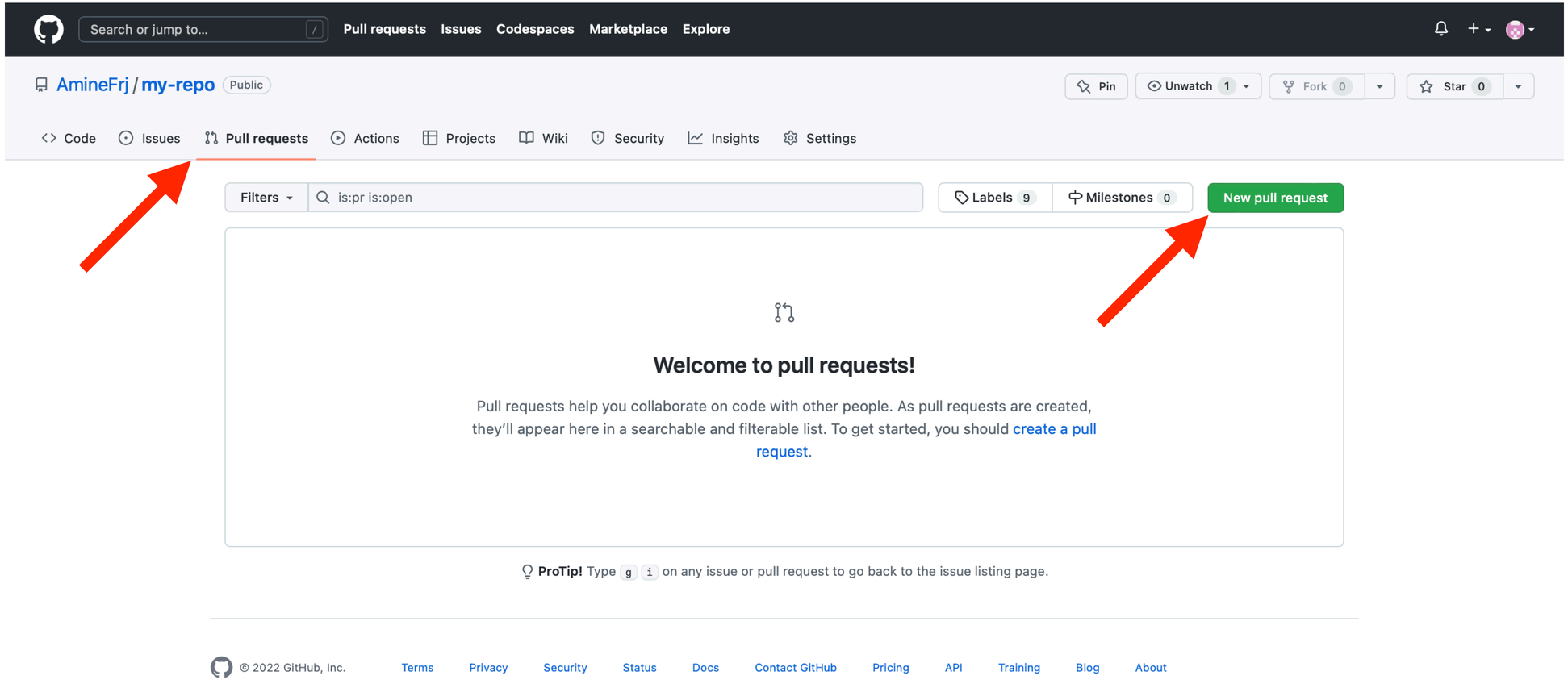
MERGE PULL REQUESTS

Why do we need merge pull requests

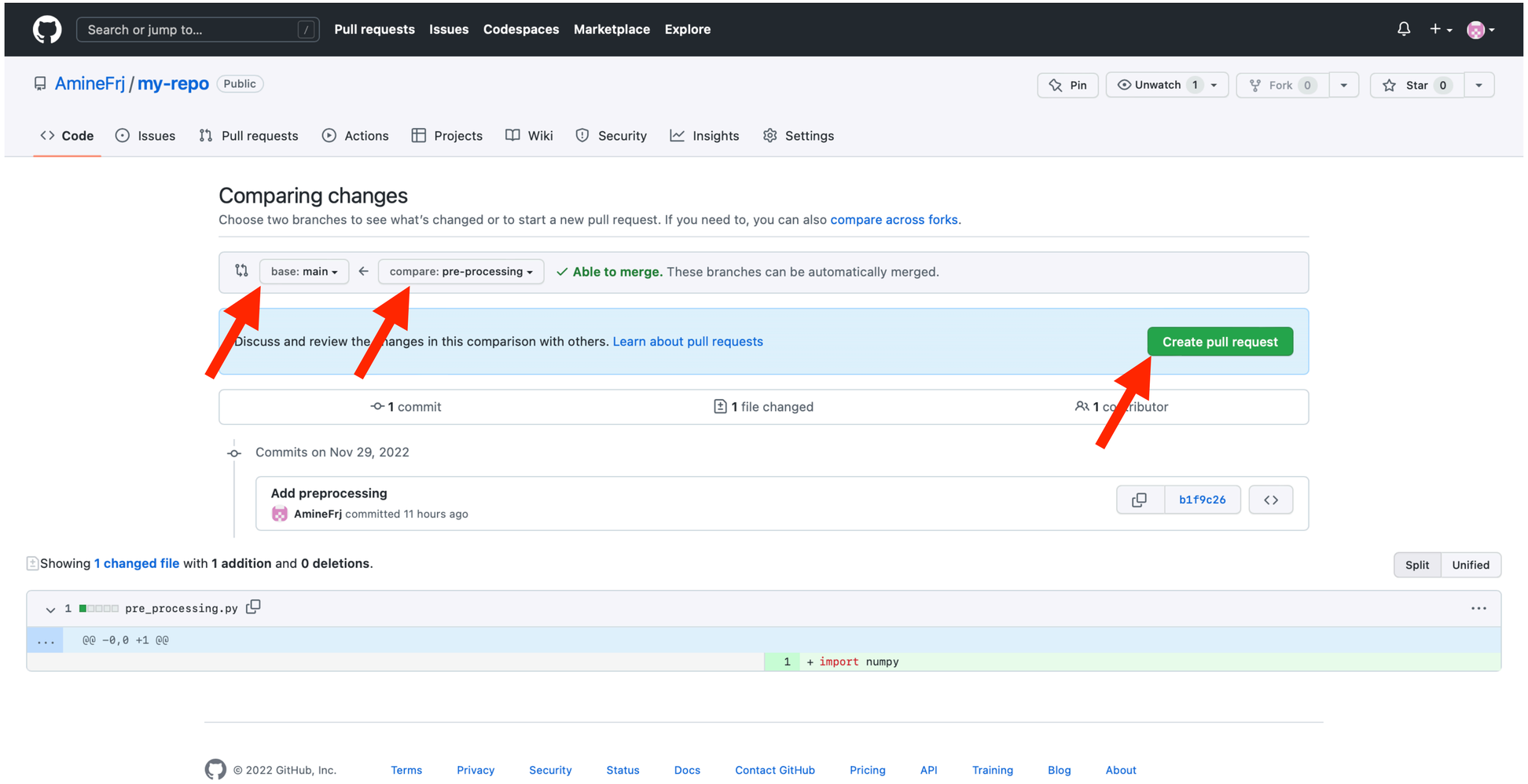
- In real world projects developers don't merge their branches directly to the main branch, otherwise everybody overwrites each other work !

- We do what we call a ***pull request***. It simply means that you ask a ***reviewer*** to review your changes and make sure that your changes passed the ***tests*** and doesn't cause problems to the main project and then ***approves*** your merge.

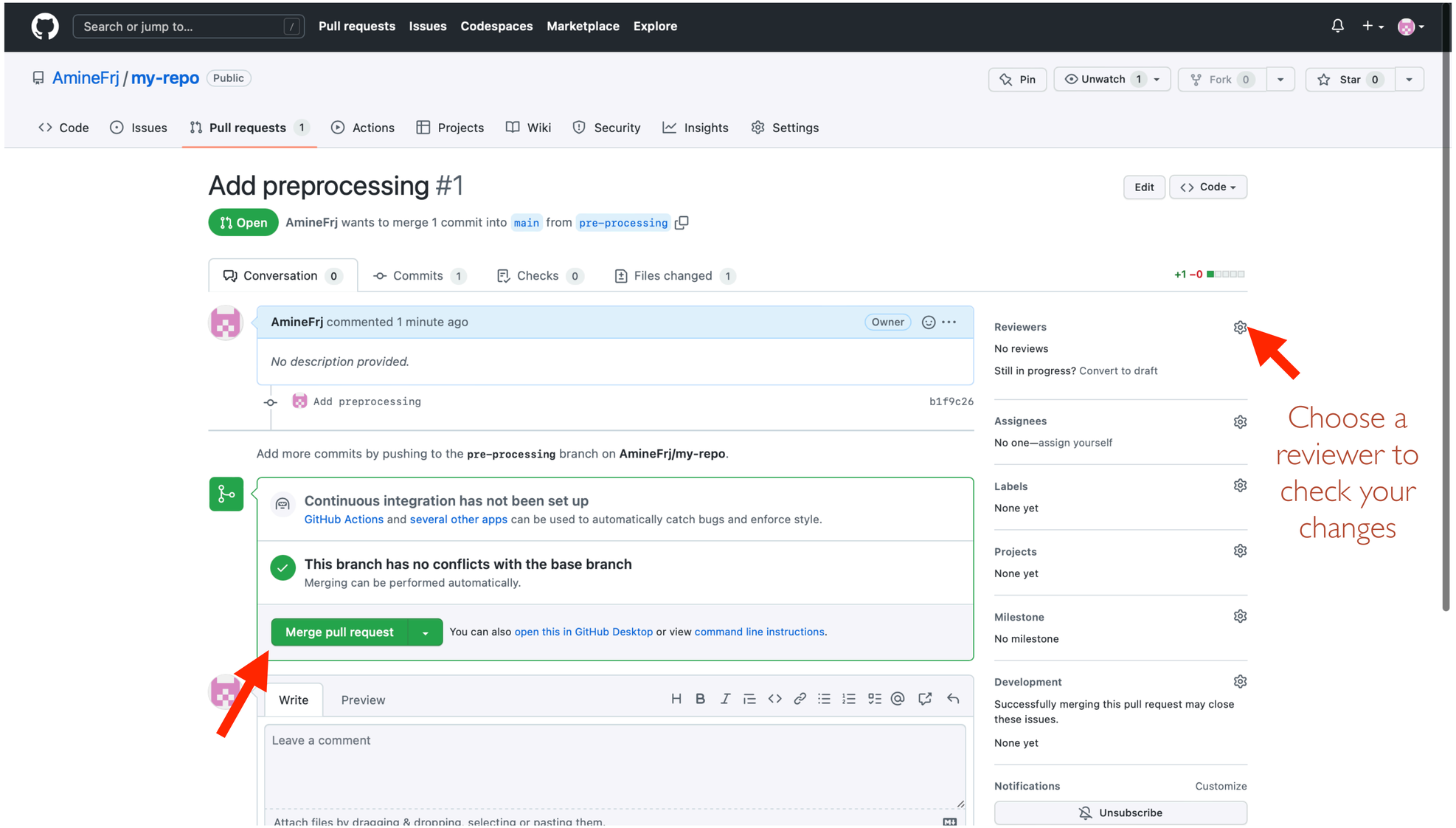
Git merge pull request



Git merge pull request

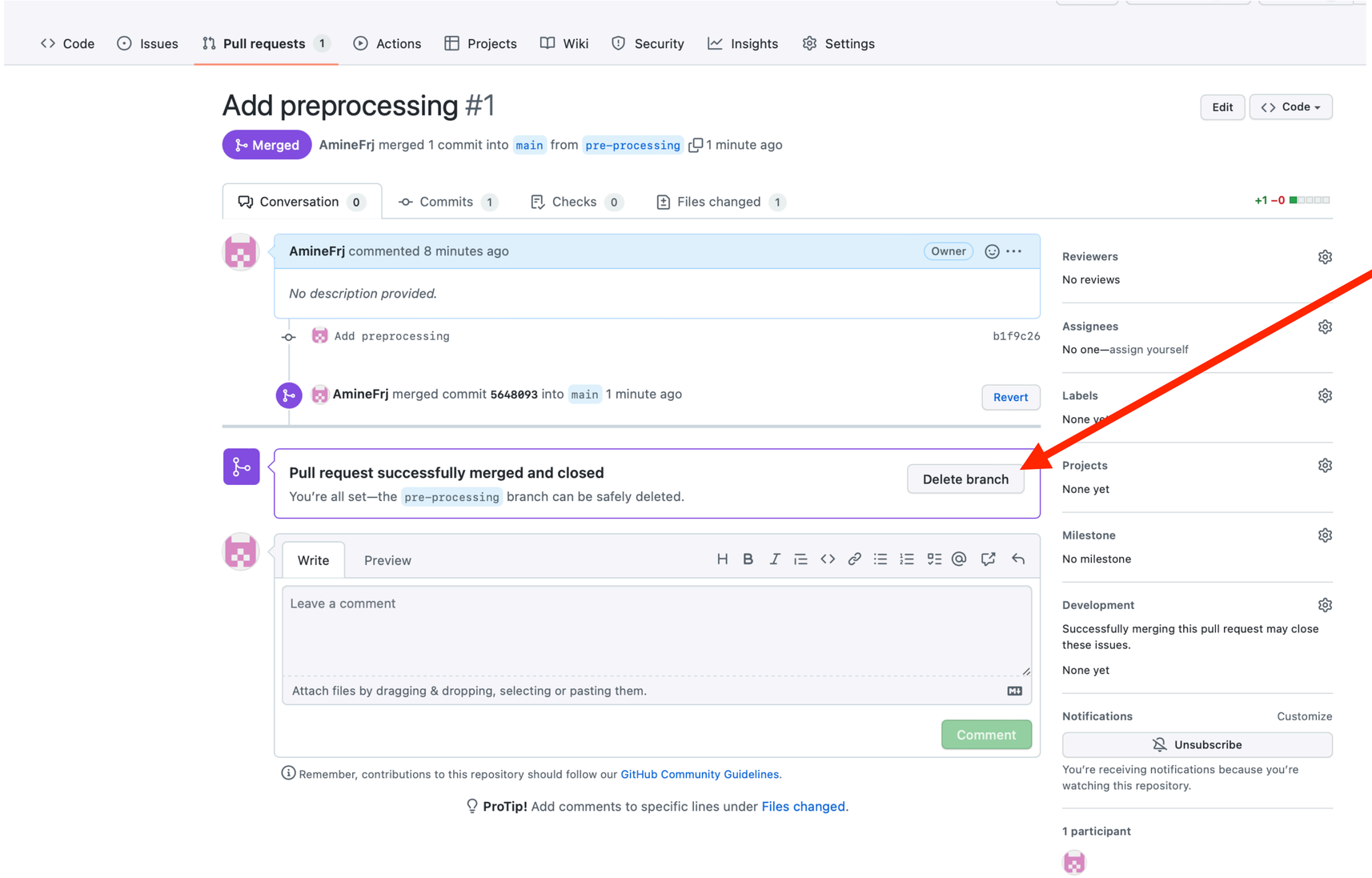


Git merge pull request



Choose a reviewer to check your changes

Git merge pull request



The branch will be removed from the server only

To update the remote branches on local use

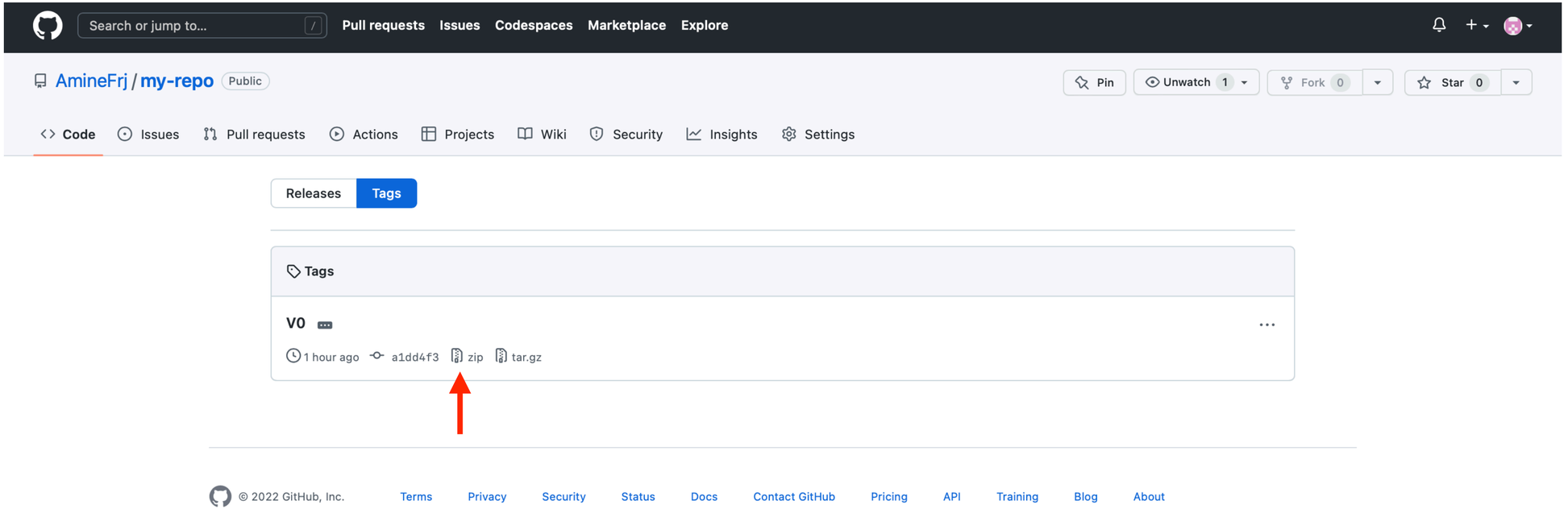
```
$ git remote prune origin
```

TAGS

Tags

Add a tag to the last commit

```
$ git tag <tag name>
$ git push origin <tag name>
$ git tag --delete <tag name>
```



Best practices

- Use pull request instead of merge
- Pull latest changes from remote main and create a branch