A complex network graph composed of numerous small black dots connected by thin black lines, forming a dense web-like structure that spans the entire background.

Data Science in
Bioinformatics
ws 22/23

Bioinformatics basics and tooling

24.10.2022

High level overview

- This session: Tips and tricks from experience
- The goal: Lay a common groundwork to base the rest of the course on
- Topics we will cover:
 - git basics
 - Conda / Mamba / environments and package managers
 - Software development setups or “How do I get productive on the cluster”
 - Containers and VMs

Git basics

- Distributed source code version control software
- Built in 2005 by Linus Torvalds for the Linux kernel
- Very powerful and widely used around the globe
- Mainly built for source code, not large binary files (there are extensions for this)
- Not to be confused with „GitHub“ and „GitLab“.

Git basics

- Most important commands and what they do:

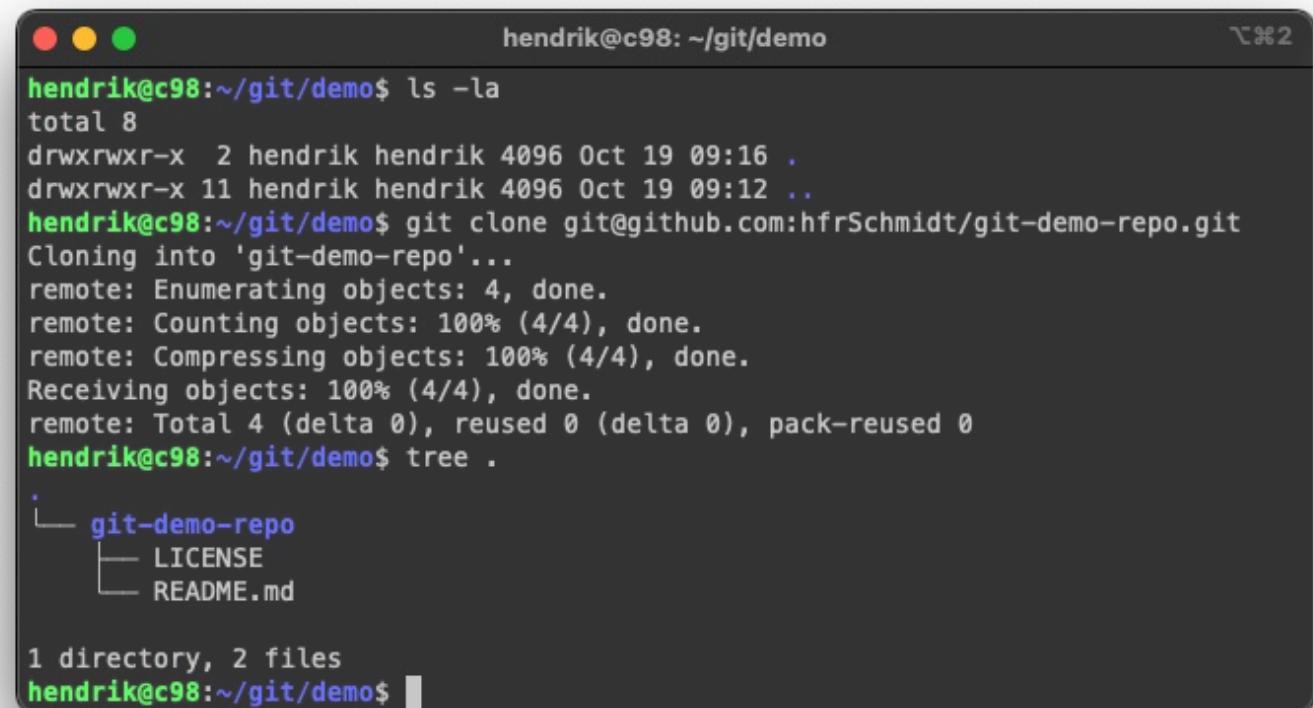
git clone	Initially copy new repository to your local machine
git pull	Update the current branch of your local copy from remote
git add	Add a file or directory to the staging area
git commit	Sum up all changes from the staging area to a change / commit
git push	Upload local commits to remote repository
git branch -a	Show all references to branches (local + remote)
git checkout	Open a new/existing branch

Git as a service providers

- Github vs Gitlab vs Bitbucket vs ...
 - This course: Github
- Not to be confused with git itself
- Issues + Pull requests
- Continuous integration / Continuous delivery (CI/CD)
 - Automatic builds / tests
 - Automatic deployment to staging / production systems
 - Not necessarily connected to git

Git in action

- Clone a repository

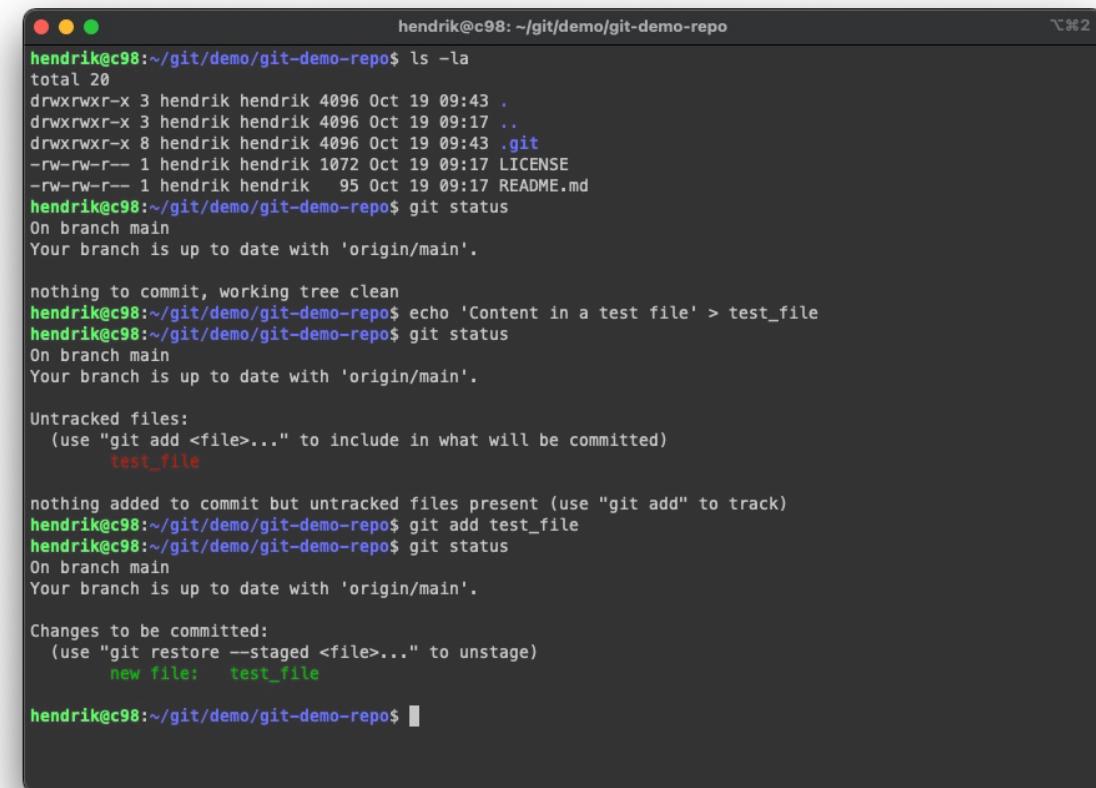


```
hendrik@c98: ~/git/demo$ ls -la
total 8
drwxrwxr-x  2 hendrik hendrik 4096 Oct 19 09:16 .
drwxrwxr-x 11 hendrik hendrik 4096 Oct 19 09:12 ..
hendrik@c98:~/git/demo$ git clone git@github.com:hfrSchmidt/git-demo-repo.git
Cloning into 'git-demo-repo'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (4/4), done.
Receiving objects: 100% (4/4), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
hendrik@c98:~/git/demo$ tree .
.
└── git-demo-repo
    ├── LICENSE
    └── README.md

1 directory, 2 files
hendrik@c98:~/git/demo$
```

Git in action

- Add a file to the repo



```
hendrik@c98:~/git/demo/git-demo-repo$ ls -la
total 20
drwxrwxr-x 3 hendrik hendrik 4096 Oct 19 09:43 .
drwxrwxr-x 3 hendrik hendrik 4096 Oct 19 09:17 ..
drwxrwxr-x 8 hendrik hendrik 4096 Oct 19 09:43 .git
-rw-rw-r-- 1 hendrik hendrik 1072 Oct 19 09:17 LICENSE
-rw-rw-r-- 1 hendrik hendrik 95 Oct 19 09:17 README.md
hendrik@c98:~/git/demo/git-demo-repo$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
hendrik@c98:~/git/demo/git-demo-repo$ echo 'Content in a test file' > test_file
hendrik@c98:~/git/demo/git-demo-repo$ git status
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    test_file

nothing added to commit but untracked files present (use "git add" to track)
hendrik@c98:~/git/demo/git-demo-repo$ git add test_file
hendrik@c98:~/git/demo/git-demo-repo$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   test_file

hendrik@c98:~/git/demo/git-demo-repo$
```

Git in action

- Commit & push the file to the remote repo

```
hendrik@c98:~/git/demo/git-demo-repo
hendrik@c98:~/git/demo/git-demo-repo$ git commit -v -m 'Add new test file'
[main 5f4f245] Add new test file
 1 file changed, 1 insertion(+)
  create mode 100644 test_file
hendrik@c98:~/git/demo/git-demo-repo$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

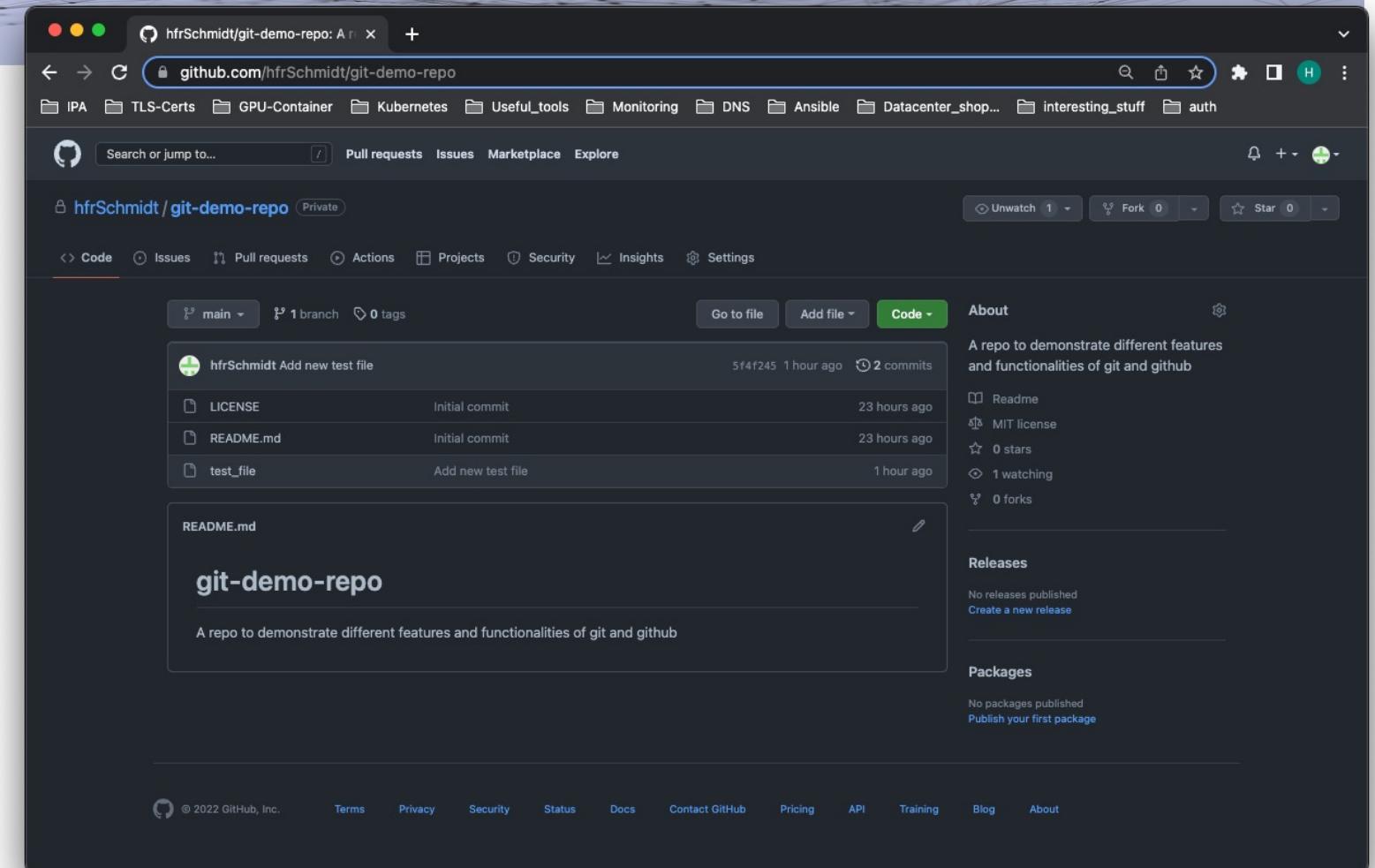
nothing to commit, working tree clean
hendrik@c98:~/git/demo/git-demo-repo$
```

```
hendrik@c98:~/git/demo/git-demo-repo
hendrik@c98:~/git/demo/git-demo-repo$ git push
git@ssh.github.com: Permission denied (publickey).
fatal: Could not read from remote repository.

Please make sure you have the correct access rights
and the repository exists.
hendrik@c98:~/git/demo/git-demo-repo$ git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 64 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 351 bytes | 351.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:hfrSchmidt/git-demo-repo.git
  4169393..5f4f245  main -> main
hendrik@c98:~/git/demo/git-demo-repo$
```

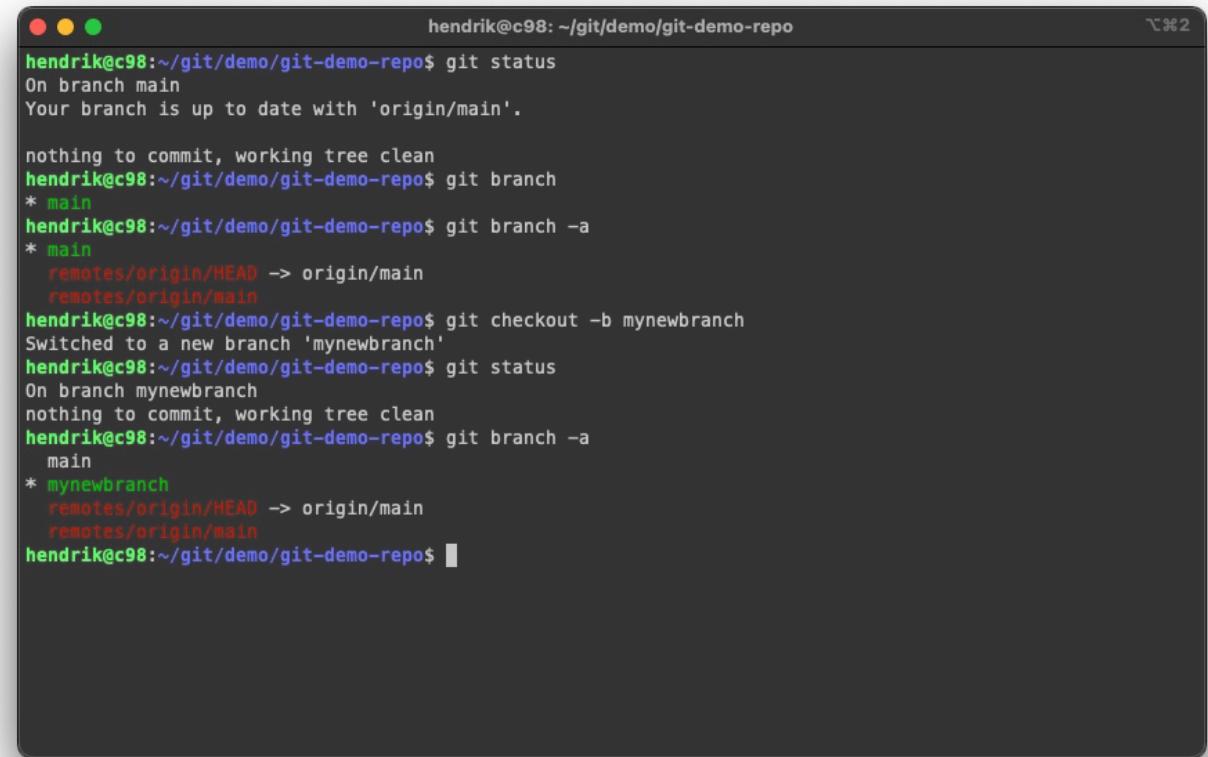
Git in action

- View from the remote side of the repo
- Currently: working on the “main” branch



Git in action

- Create a new branch

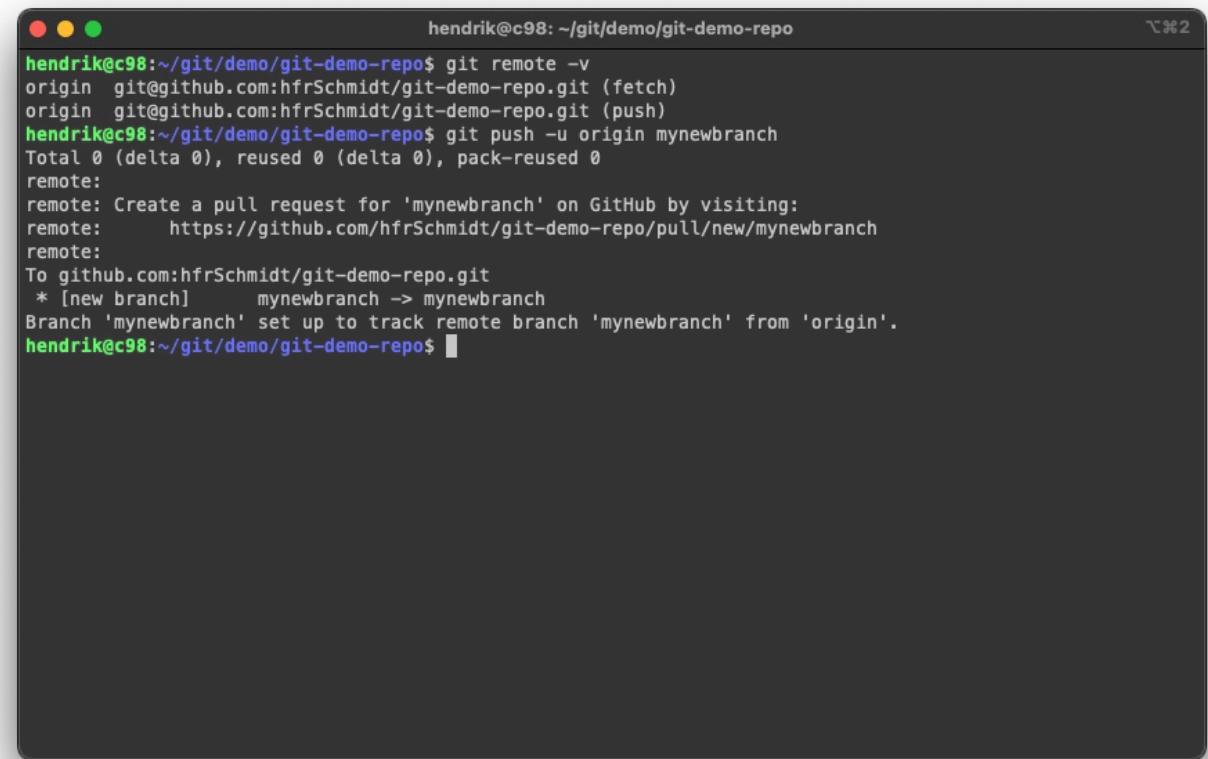


```
hendrik@c98: ~/git/demo/git-demo-repo
hendrik@c98:~/git/demo/git-demo-repo$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean
hendrik@c98:~/git/demo/git-demo-repo$ git branch
* main
hendrik@c98:~/git/demo/git-demo-repo$ git branch -a
* main
  remotes/origin/HEAD -> origin/main
  remotes/origin/main
hendrik@c98:~/git/demo/git-demo-repo$ git checkout -b mynewbranch
Switched to a new branch 'mynewbranch'
hendrik@c98:~/git/demo/git-demo-repo$ git status
On branch mynewbranch
nothing to commit, working tree clean
hendrik@c98:~/git/demo/git-demo-repo$ git branch -a
  main
* mynewbranch
  remotes/origin/HEAD -> origin/main
  remotes/origin/main
hendrik@c98:~/git/demo/git-demo-repo$
```

Git in action

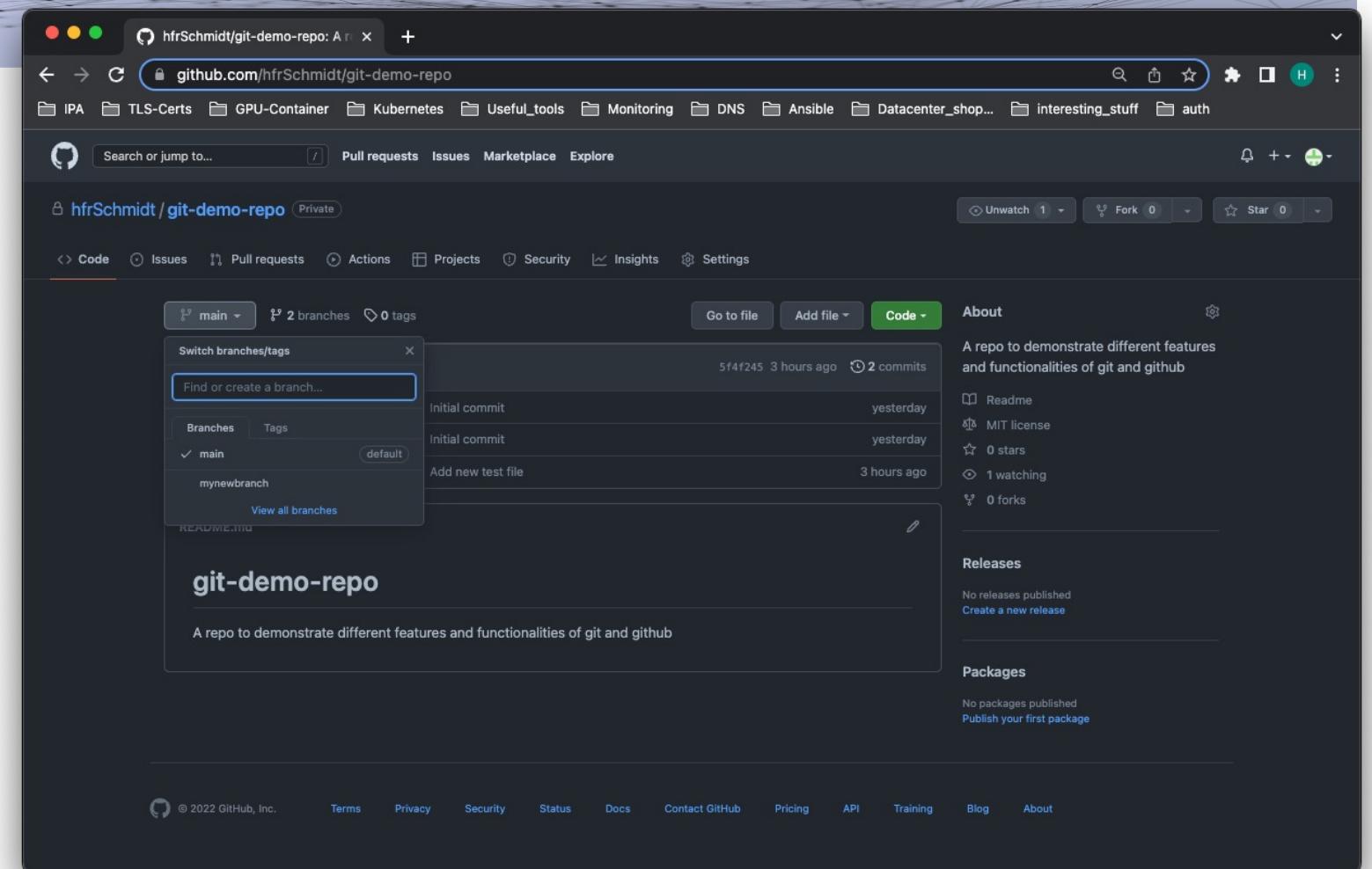
- Push the new branch to the remote



```
hendrik@c98: ~/git/demo/git-demo-repo
hendrik@c98:~/git/demo/git-demo-repo$ git remote -v
origin  git@github.com:hfrSchmidt/git-demo-repo.git (fetch)
origin  git@github.com:hfrSchmidt/git-demo-repo.git (push)
hendrik@c98:~/git/demo/git-demo-repo$ git push -u origin mynewbranch
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'mynewbranch' on GitHub by visiting:
remote:     https://github.com/hfrSchmidt/git-demo-repo/pull/new/mynewbranch
remote:
To github.com:hfrSchmidt/git-demo-repo.git
 * [new branch]      mynewbranch -> mynewbranch
Branch 'mynewbranch' set up to track remote branch 'mynewbranch' from 'origin'.
hendrik@c98:~/git/demo/git-demo-repo$
```

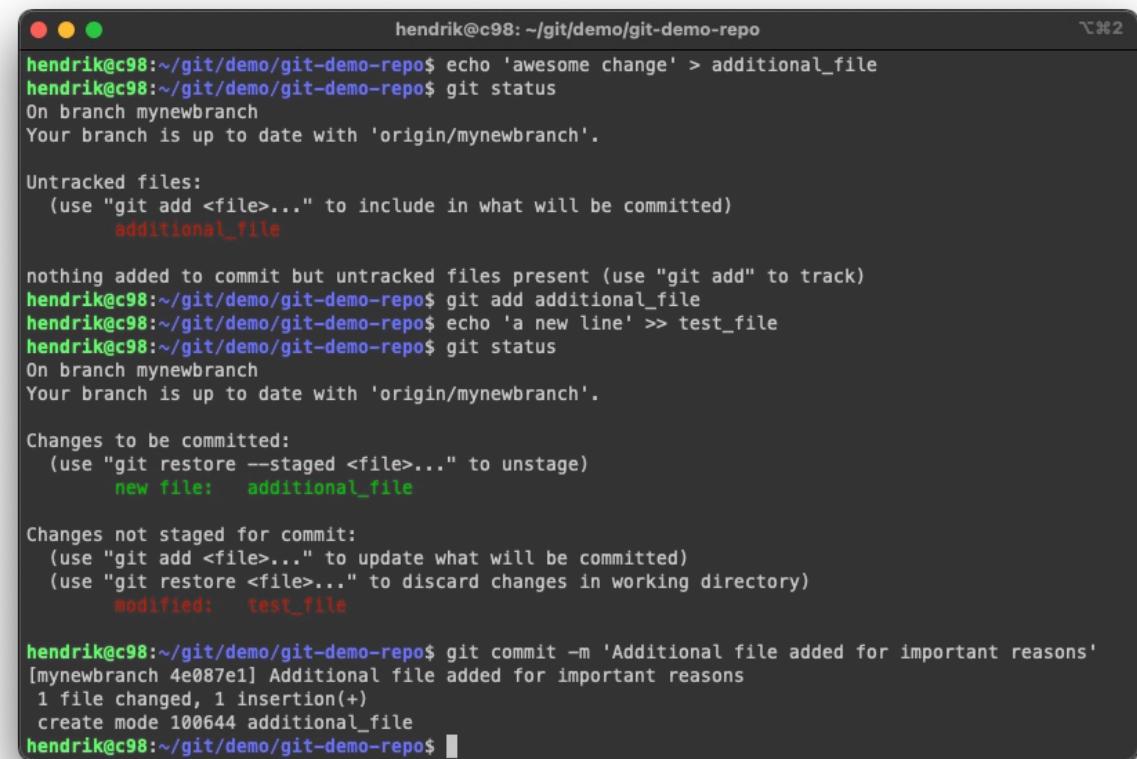
Git in action

- View from the remote side of the repo
- The new branch is there



Git in action

- Do some work on the new branch



```
hendrik@c98: ~/git/demo/git-demo-repo
hendrik@c98:~/git/demo/git-demo-repo$ echo 'awesome change' > additional_file
hendrik@c98:~/git/demo/git-demo-repo$ git status
On branch mynewbranch
Your branch is up to date with 'origin/mynewbranch'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    additional_file

nothing added to commit but untracked files present (use "git add" to track)
hendrik@c98:~/git/demo/git-demo-repo$ git add additional_file
hendrik@c98:~/git/demo/git-demo-repo$ echo 'a new line' >> test_file
hendrik@c98:~/git/demo/git-demo-repo$ git status
On branch mynewbranch
Your branch is up to date with 'origin/mynewbranch'.

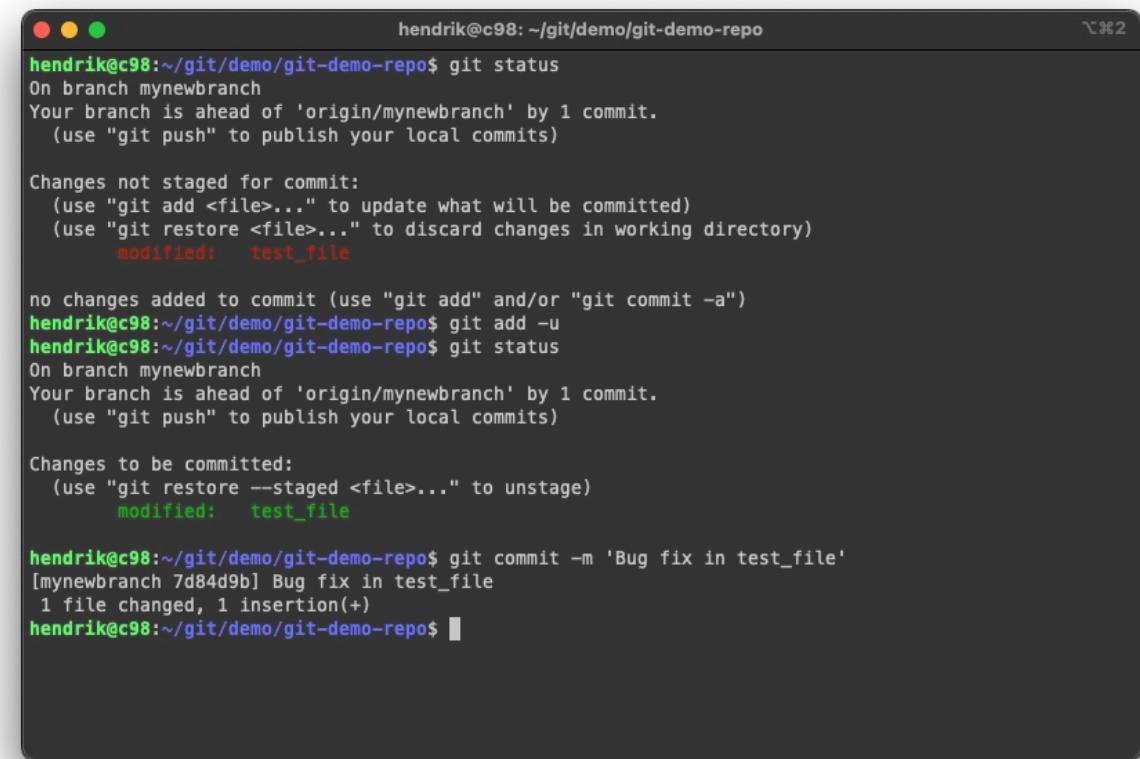
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   additional_file

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:  test_file

hendrik@c98:~/git/demo/git-demo-repo$ git commit -m 'Additional file added for important reasons'
[mynewbranch 4e087e1] Additional file added for important reasons
 1 file changed, 1 insertion(+)
 create mode 100644 additional_file
hendrik@c98:~/git/demo/git-demo-repo$
```

Git in action

- Do some more work on the new branch



```
hendrik@c98:~/git/demo/git-demo-repo$ git status
On branch mynewbranch
Your branch is ahead of 'origin/mynewbranch' by 1 commit.
  (use "git push" to publish your local commits)

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   test_file

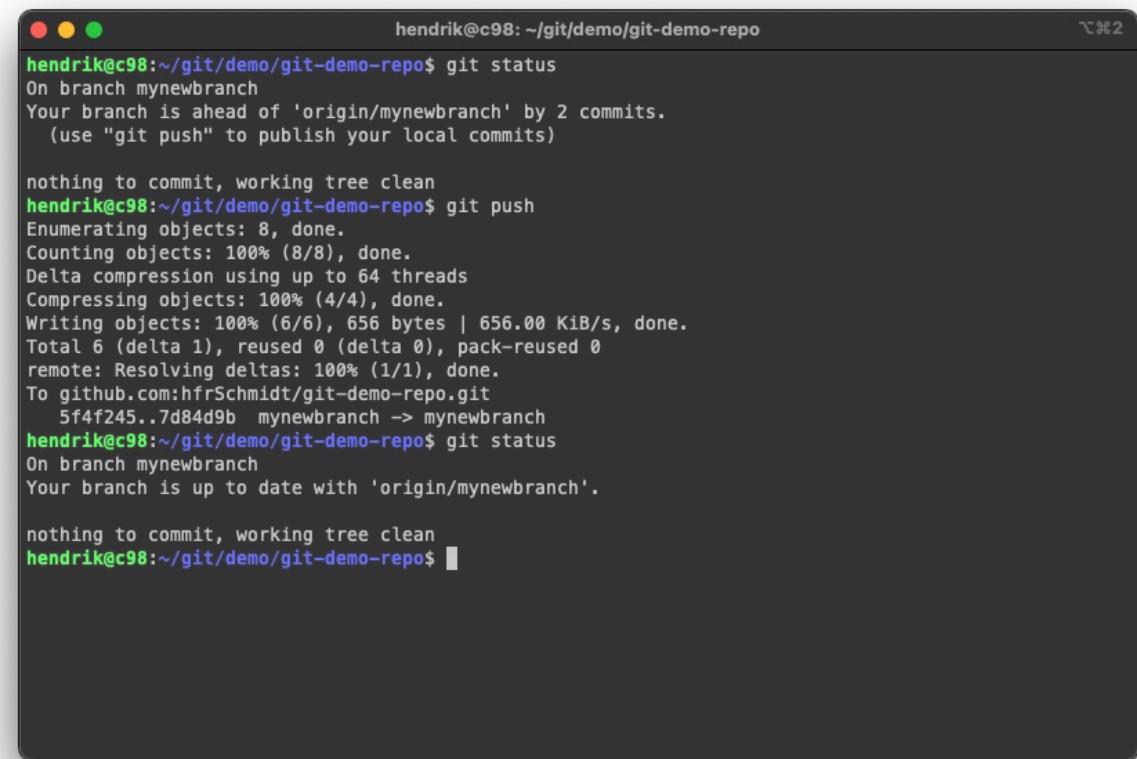
no changes added to commit (use "git add" and/or "git commit -a")
hendrik@c98:~/git/demo/git-demo-repo$ git add -
hendrik@c98:~/git/demo/git-demo-repo$ git status
On branch mynewbranch
Your branch is ahead of 'origin/mynewbranch' by 1 commit.
  (use "git push" to publish your local commits)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   test_file

hendrik@c98:~/git/demo/git-demo-repo$ git commit -m 'Bug fix in test_file'
[mynewbranch 7d84d9b] Bug fix in test_file
 1 file changed, 1 insertion(+)
hendrik@c98:~/git/demo/git-demo-repo$
```

Git in action

- Push the changes to the remote branch



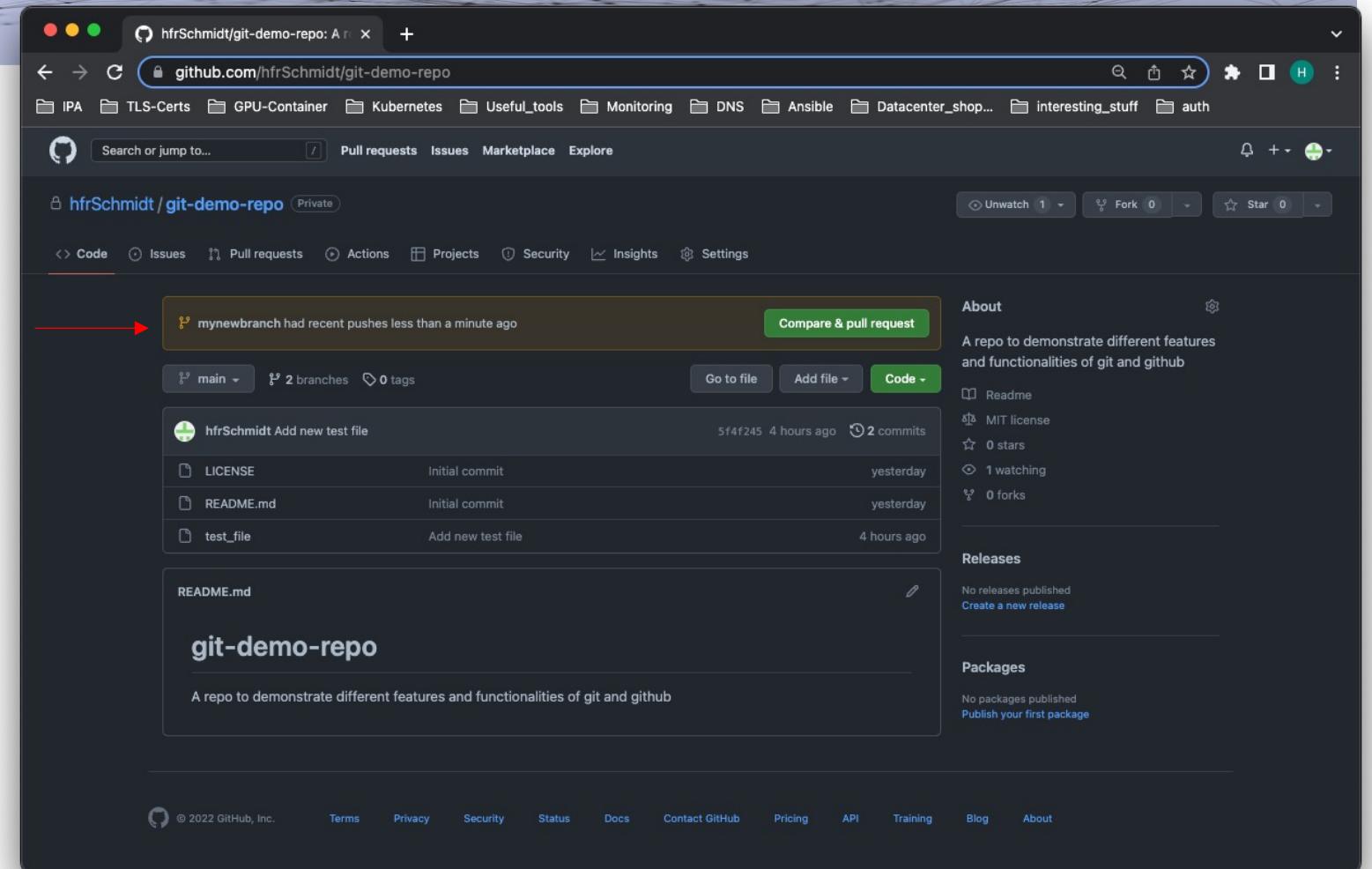
```
hendrik@c98: ~/git/demo/git-demo-repo$ git status
On branch mynewbranch
Your branch is ahead of 'origin/mynewbranch' by 2 commits.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
hendrik@c98:~/git/demo/git-demo-repo$ git push
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 64 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (6/6), 656 bytes | 656.00 KiB/s, done.
Total 6 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
To github.com:hfrSchmidt/git-demo-repo.git
  5f4f245..7d84d9b mynewbranch -> mynewbranch
hendrik@c98:~/git/demo/git-demo-repo$ git status
On branch mynewbranch
Your branch is up to date with 'origin/mynewbranch'.

nothing to commit, working tree clean
hendrik@c98:~/git/demo/git-demo-repo$
```

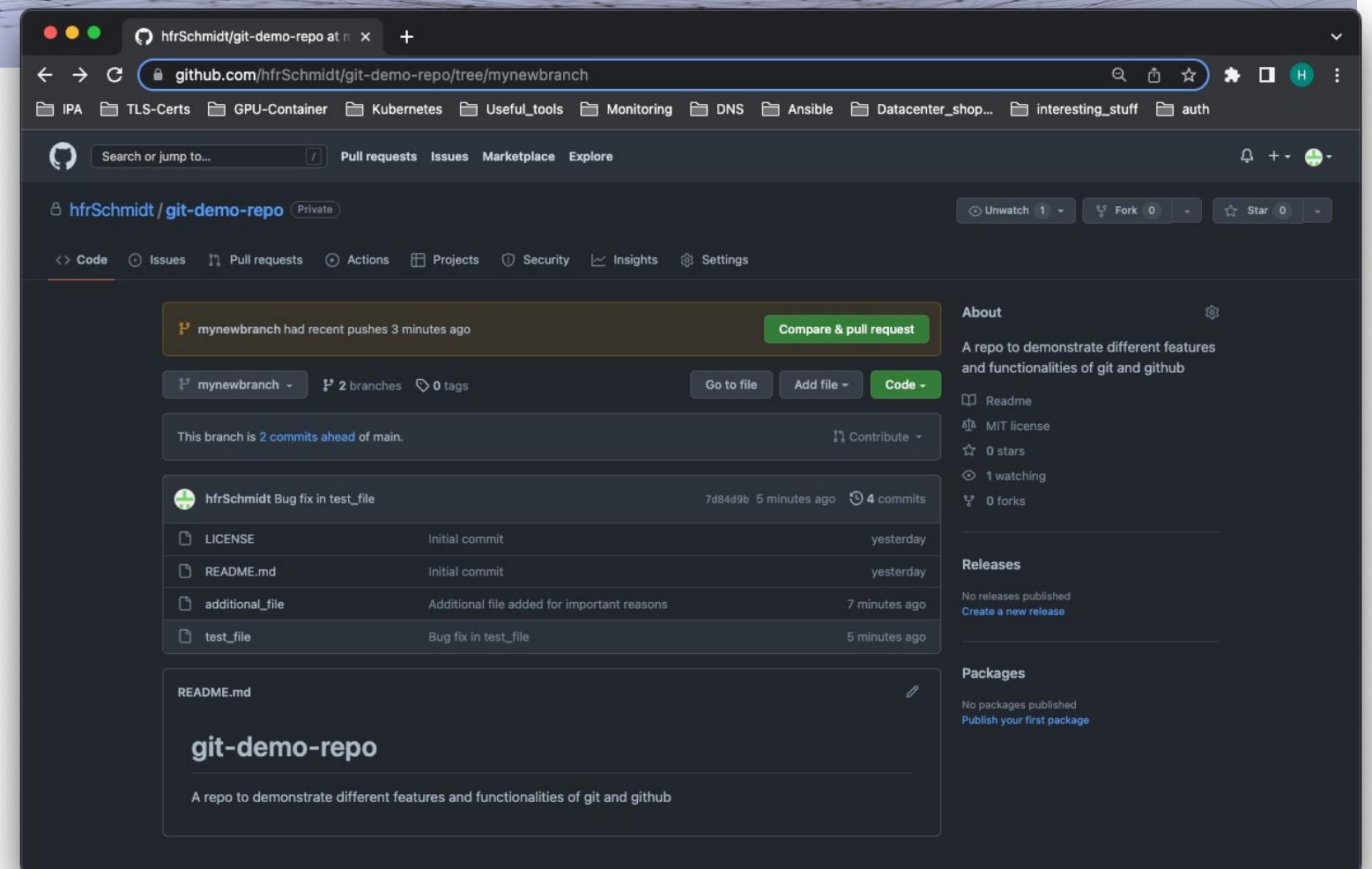
Git in action

- View from the remote side of the repo
- Changes not visible in “main”



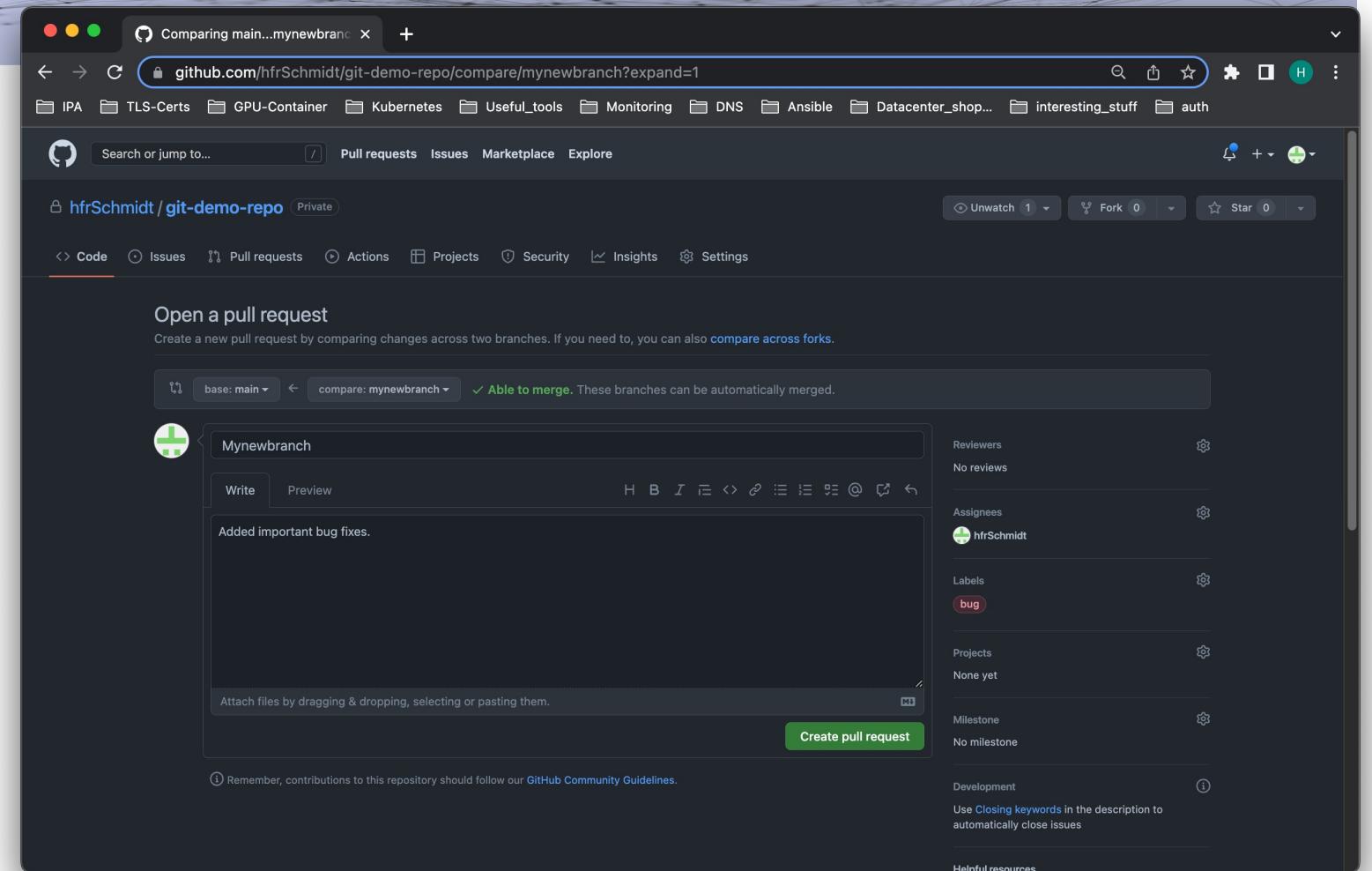
Git in action

- View from the remote side of the repo
- “mynewbranch” has the changes



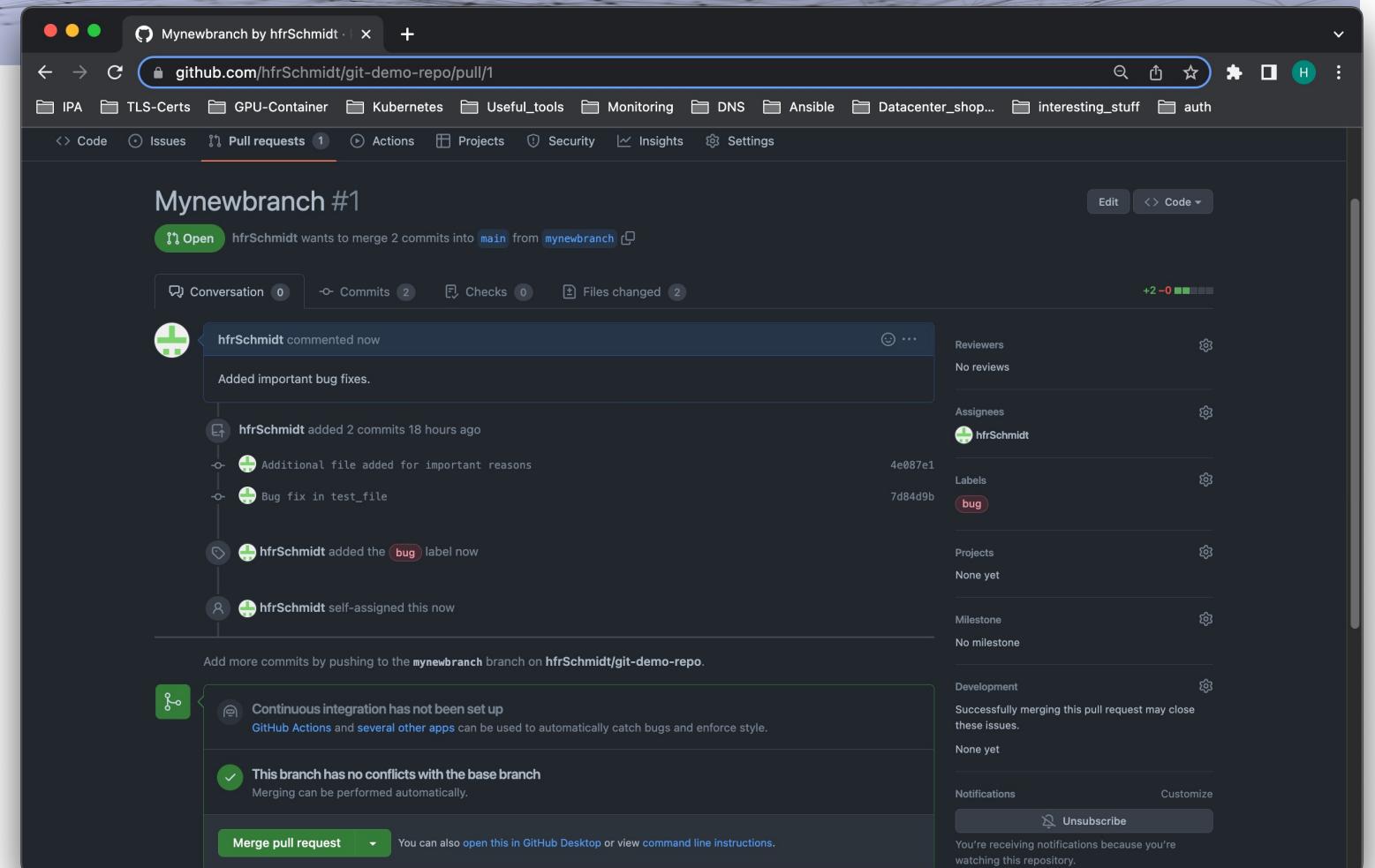
Git in action

- Creating a pull request



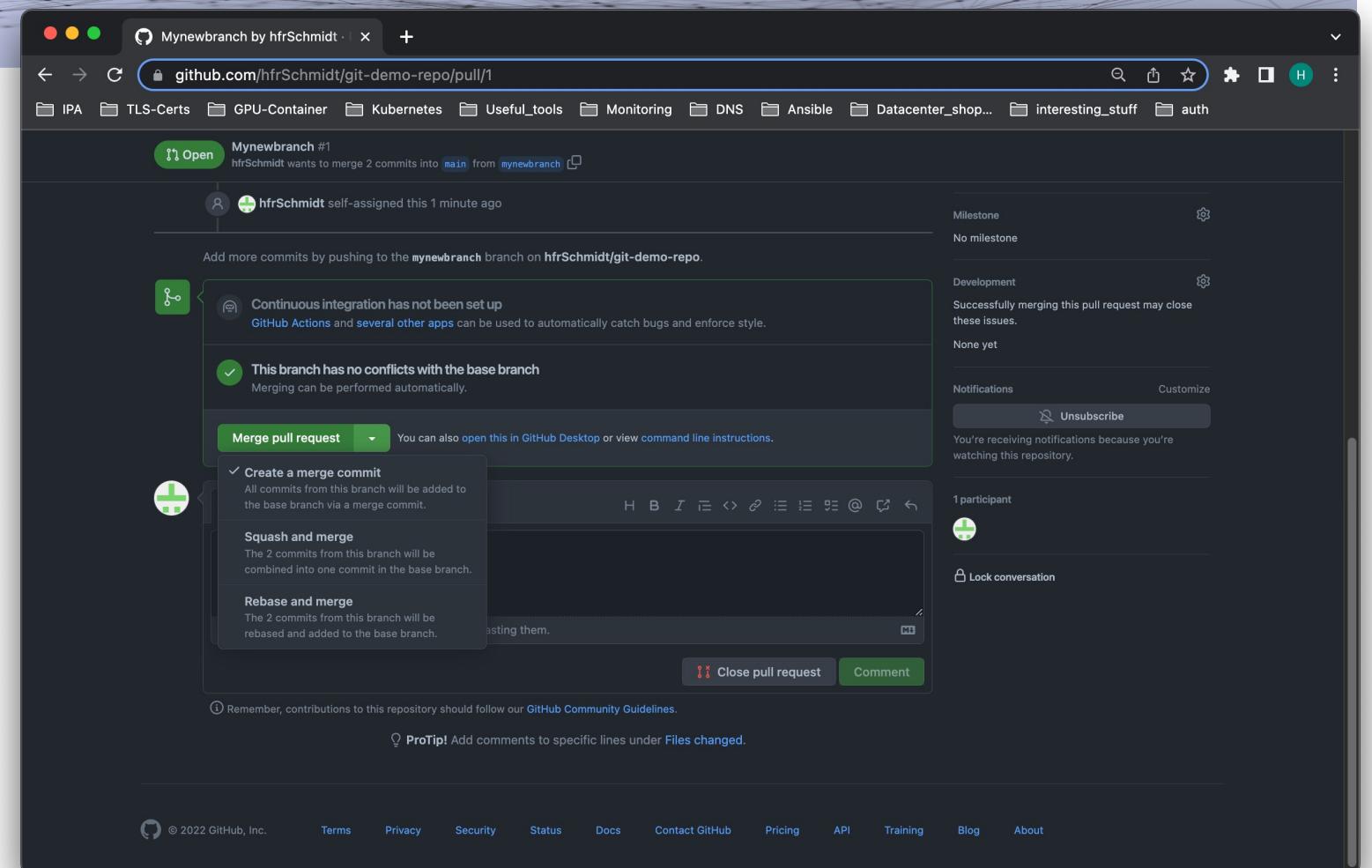
Git in action

- Creating a pull request



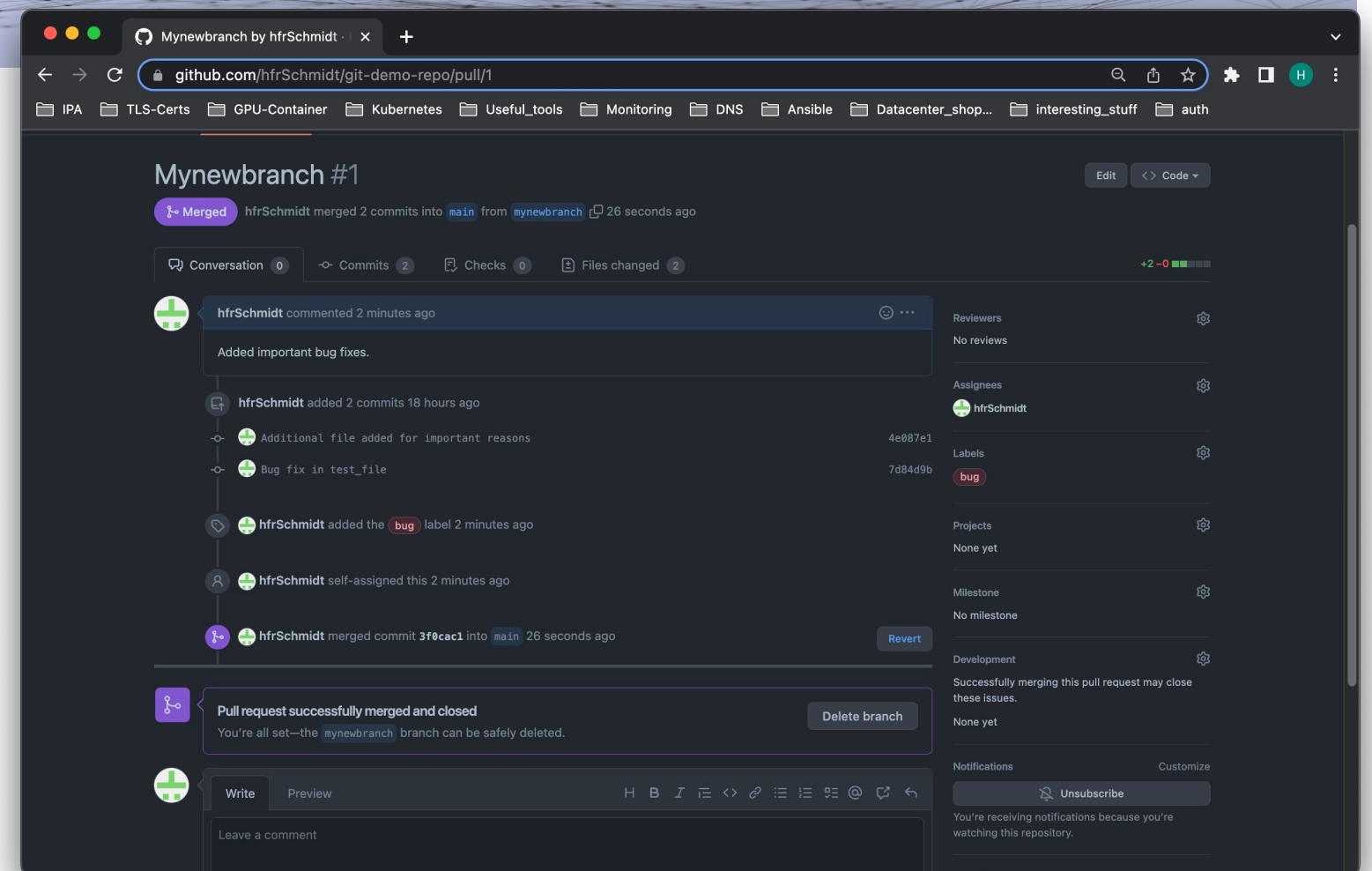
Git in action

- Creating a pull request: PR merge strategy



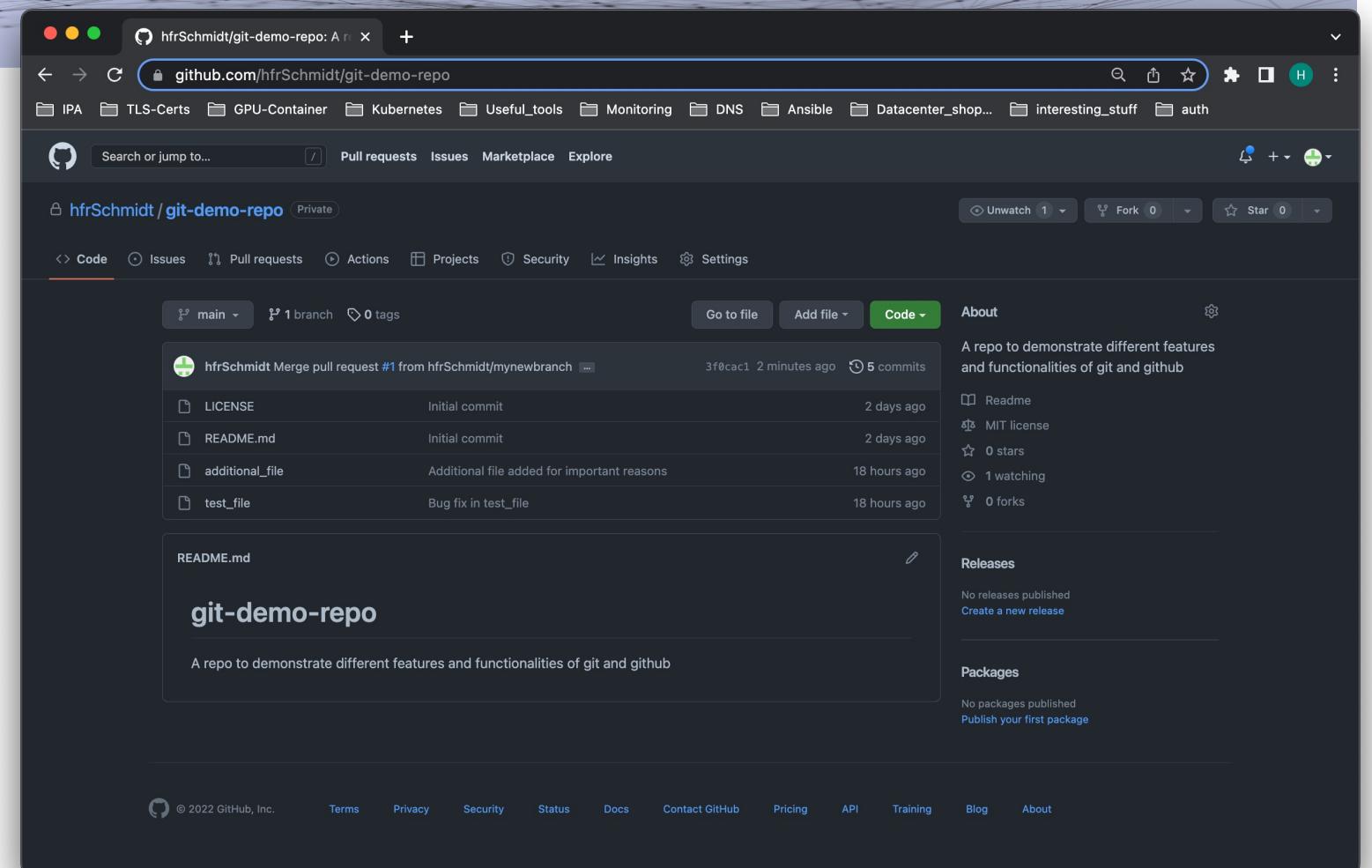
Git in action

- Merged PR



Git in action

- Merged PR



Software development workflows

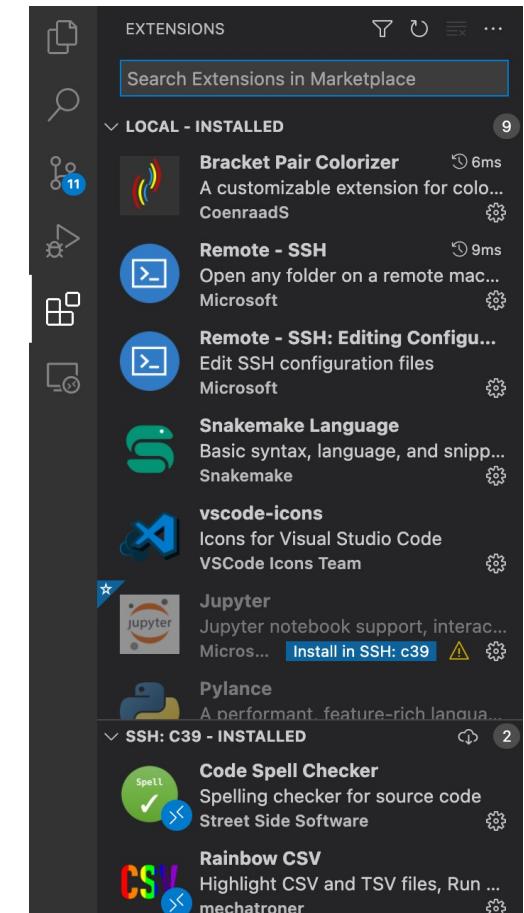
- Visual Studio Code:
 - Remote development extension
 - (Remote development container extension)
- Edit files directly on the server (not entirely recommended):
 - VI
 - VIM
 - Nano
- Edit files locally and sync them with the remote server (not recommended)

Software development workflows

- Example:
 - VS Code Remote SSH extension
 - VS Code extensions:
 - Code Spell Checker
 - Rainbow CSV
 - Bracket pair colorizer
 - Git lens
 - Conda / Mamba environments

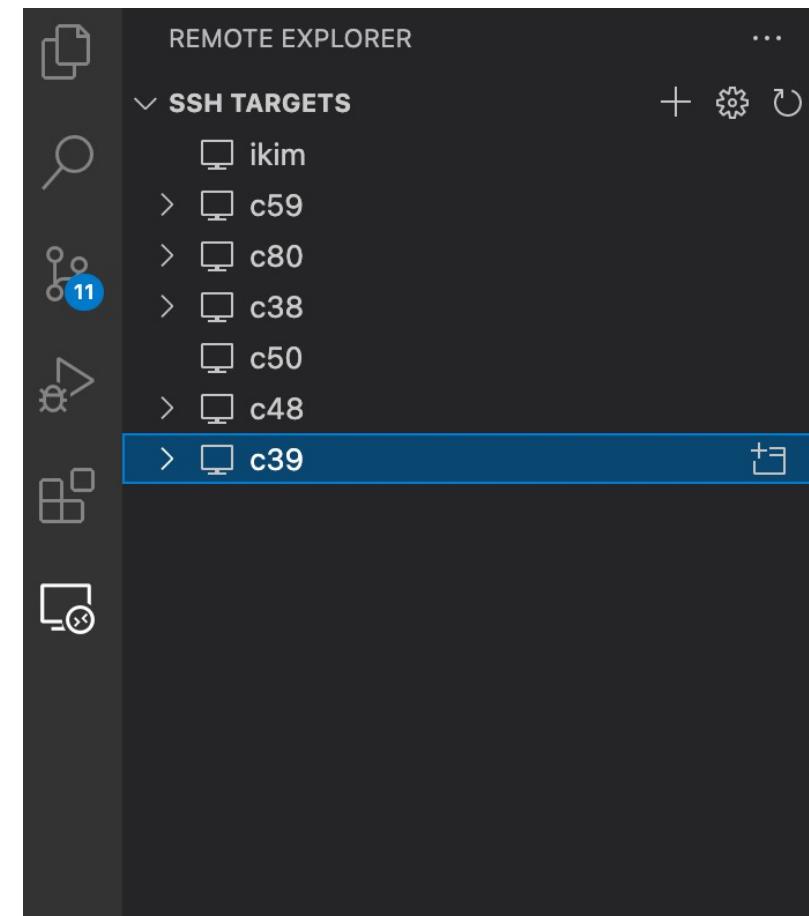
Software development workflows

- Different extensions that improve the usability of VSCode
- Helpful code corrections



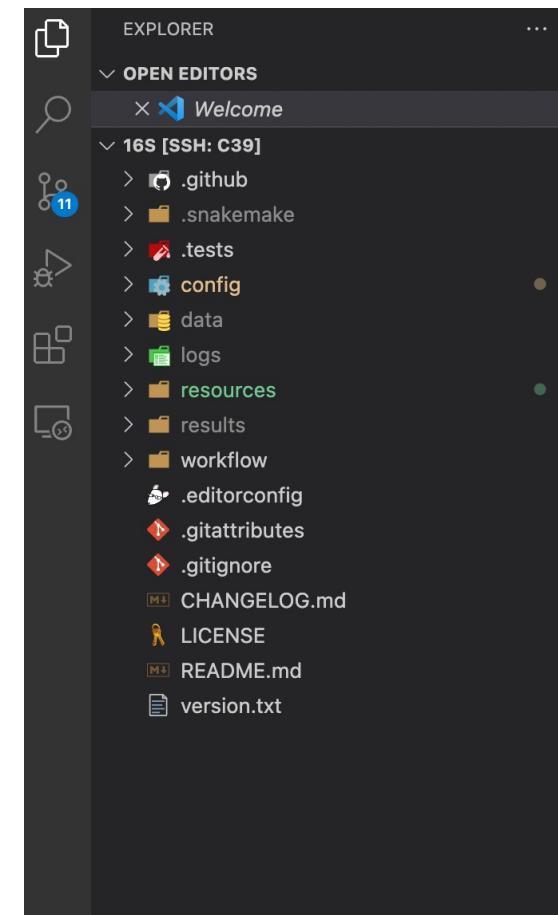
Software development workflows

- Remote-SSH extension
 - Showing the server nodes that are saved in the ssh-config
 - Makes it easier to connect to the nodes



Software development workflows

- Cloned repository of our 16S pipeline
- Branch symbol shows changes made to the code in comparison to main-branch
- Tracks the changes automatically

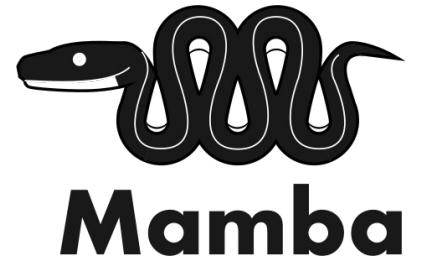


Dependency management

- Common problem: „Your code does not run on my system.“
 - Potential causes: Different versions of libraries, missing libraries, different interpreter versions, ...
- Use cases for fixing dependencies in one's code:
 - Reproducibility (especially in science)
 - Stability
- How can you do this?
 - Don't update your OS (very dangerous! Do not do this!)
 - Use a dependency / environment / package manager
 - Pipenv
 - Conda
 - ...?

Dependency management

- Conda
 - Package manager + environment manager
- Conda vs Mamba
- Anaconda vs Miniconda vs Mambaforge
- You can install mambaforge via the links at:
<https://github.com/conda-forge/miniforge#mambaforge>



Dependency management

- For example: a basic python app

```
git > demo > git-demo-repo > ✨ test_sample.py > ...
1  import pytest
2  import requests
3
4  class FunctionalityClass:
5      def get_gh_public_timeline(self):
6          return requests.get('https://api.github.com/events')
7
8      def post_httpbin_org(self):
9          return requests.post('https://httpbin.org/post', data={'key': 'value'})
10
11     class TestFuctionalityClass:
12         def test_get_gh_public_timeline_success(self):
13             response = FunctionalityClass().get_gh_public_timeline()
14             assert(response.status_code == requests.codes.ok)
15
16         def test_post_httpbin_org_success(self):
17             response = FunctionalityClass().post_httpbin_org()
18             assert(response.status_code == requests.codes.ok)
19
```

Dependency management

- Create an environment with mamba

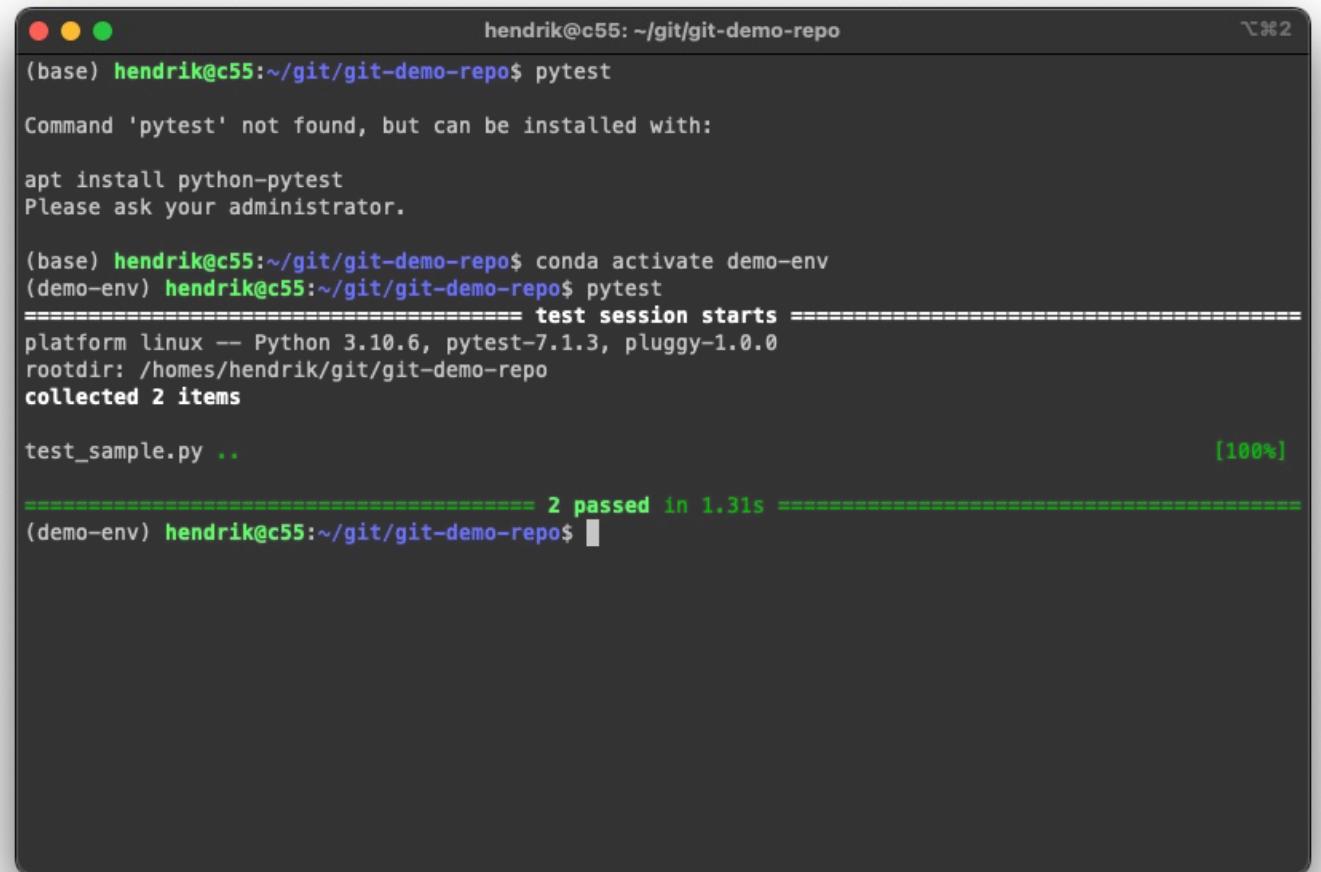
```
(base) hendrik@c55:~/git/git-demo-repo$ mamba create -p ~/.conda/envs/demo-env pytest requests
[██████████] 100%|██████████| 26.3MB / 26.3MB @ 2.7MB/s Finalizing 10.5s
mamba (0.27.0) supported by @QuantStack
GitHub: https://github.com/mamba-org/mamba
Twitter: https://twitter.com/QuantStack

Looking for: ['pytest', 'requests']

  conda-forge/noarch                               No change
[+] 10.5s
  conda-forge/linux-64
```

Dependency management

- Test the environment



```
hendrik@c55: ~/git/git-demo-repo
(base) hendrik@c55:~/git/git-demo-repo$ pytest
Command 'pytest' not found, but can be installed with:
apt install python-pytest
Please ask your administrator.

(base) hendrik@c55:~/git/git-demo-repo$ conda activate demo-env
(demo-env) hendrik@c55:~/git/git-demo-repo$ pytest
===== test session starts =====
platform linux -- Python 3.10.6, pytest-7.1.3, pluggy-1.0.0
rootdir: /homes/hendrik/git/git-demo-repo
collected 2 items

test_sample.py .. [100%]

===== 2 passed in 1.31s =====
(demo-env) hendrik@c55:~/git/git-demo-repo$
```

Containers and VMs

- Large differences concerning:
 - Iteration speed
 - Storage overhead
 - Security / isolation
 - Data storage

