

A1 Git

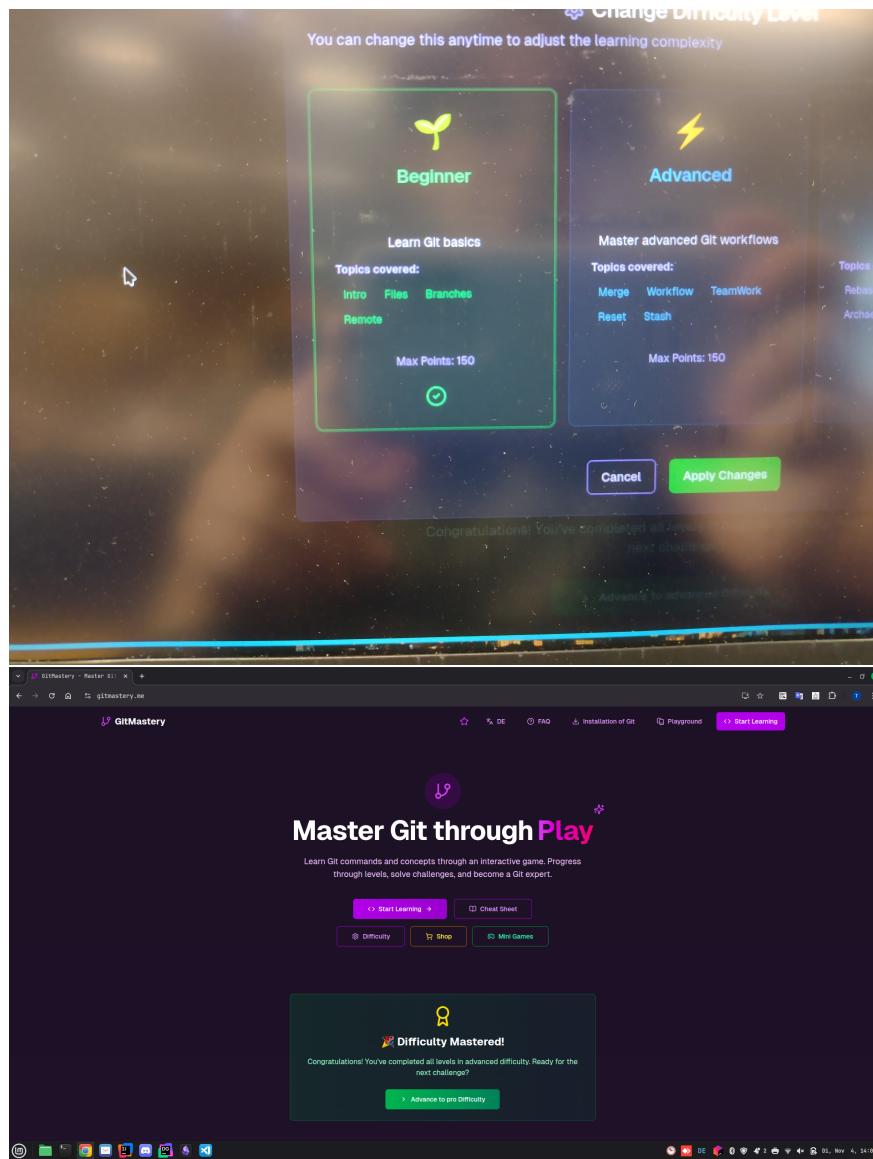
Beginner:

- git init
- git status
- git clone it clone <https://github.com/user/repo.git>, cd repo
- git init, git add
- git commit -m Ahhhhhhhhhh
- git rm temp.txt
- git branch -a
- git branch feature, git switch -c feature
- git switch feature
- git checkout feature
- git checkout -b feature
- git remote add mom aaa
- git add ., git commit -m aaa, git push -u origin main
- git checkout -b 69, git push -u origin 69

Advanced:

- git add .
- git checkout -b develop, git switch feature, git merge develop
- git switch develop, git merge main
- git merge --abort
- git switch -c feature/user-auth, git add ., git commit, git push -u origin main, git switch main, git merge feature/user-auth
- git switch -c hotfix/security, git add ., git commit -m aa, git switch main, git merge hotfix/security-patch
- git switch -c release/2.0.0, git add ., git commit -m "Prepare release 2.0.0", git switch main, git merge release/2.0.0, git tag v2.0.0
- git pull
- git add /src/auth/login.js, git commit -m "Improve password validation requirements, git pull origin main, git add ., git commit -m "Merge Sarah's email validation with my password improvements"

- git switch -c feature/password-reset, git add ., git commit -m "Add password reset functionality", git push origin feature/password-reset
- 3x(git reset --soft)
- 3x(git reset --soft)
- git log, git reset
- git switch -c hotfix, git switch -c feature, git stash pop
- git stash, git switch main, git switch -c feature, git switch feature/old-task, git stash pop



A2 Herausforderungen

[a] Datei zum letzten Commit hinzufügen, ohne neuen Commit im Verlauf zu erzeugen:

```
git add <datei>
git commit --amend --no-edit
```

[b] Commit aus dem lokalen Verlauf vollständig löschen:

```
git reset --hard HEAD~1
```

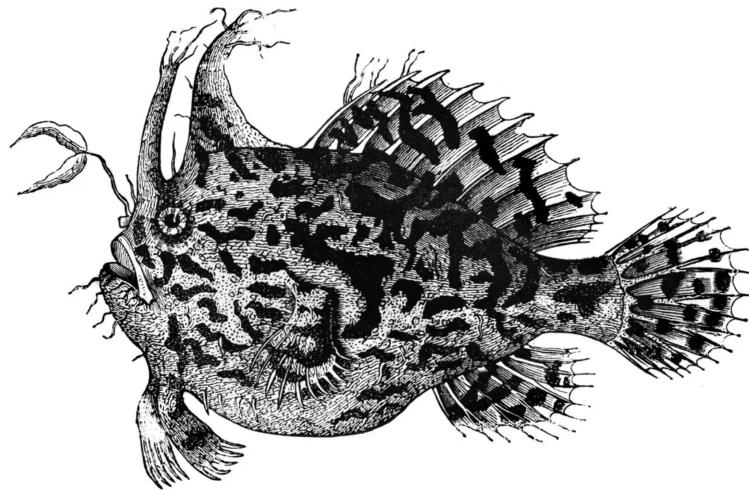
[c] Letzten Commit im Remote-Repository rückgängig machen, Rücknahme bleibt sichtbar:

```
git revert HEAD
git push origin <branch>
```

[d] Unterschiede zwischen main und feature seit dem gemeinsamen Vorgänger anzeigen:

```
git diff main...feature
```

The best version control is to keep multiple copies of the project



How to create Merge Conflicts at 3 A.M

Don't be a Git!

O RLY?