

Tutorial: <https://vimeo.com/572626937>

- 1. Git installieren**
- 2. Github Repository erstellen**
- 3. VSCode Authentifizieren
(einmalig)**
- 4. Repo initialisieren und mit
Github verknüpfen**
- 5. Commits prüfen**
- 6. Workflow auf VSCode**

GIT, GITHUB, VISUAL STUDIO CODE

1. Git installieren

Überprüft zuerst, ob Git bereits auf eurem Rechner installiert ist. Gebt in einem Terminal den Befehl

```
git --version
```

ein. Die Antwort könnte lauten

```
git version 2.24.3 (Apple Git-128)
```

dann ist git installiert.

Falls Git noch nicht installiert ist, kannst du hier einen Installer laden:

<https://git-scm.com/downloads>

GIT, GITHUB, VISUAL STUDIO CODE

2. Github Repository erstellen

Logge dich über den Browser ein in deinen Github Account.

Falls du noch keinen hast: [Studenten Account erstellen](#)

Erstelle auf Github ein neues Repository: **modul_webtech**. Setze es auf Public und mache **kein** Häckchen bei Add a Readme file

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Owner ^{*} / Repository name ^{*}

hzuell / kickstartTest ✓

Great repository names are short and memorable. Need inspiration? How about [special-enigma](#)?

Description (optional)

kickstartTest

☐ Public
Anyone on the internet can see this repository. You choose who can commit.

☒ Private
You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

☐ Add a README file
This is where you can write a long description for your project. [Learn more](#).

☐ Add .gitignore
Choose which files not to track from a list of templates. [Learn more](#).

☐ Choose a license
A license tells others what they can and can't do with your code. [Learn more](#).

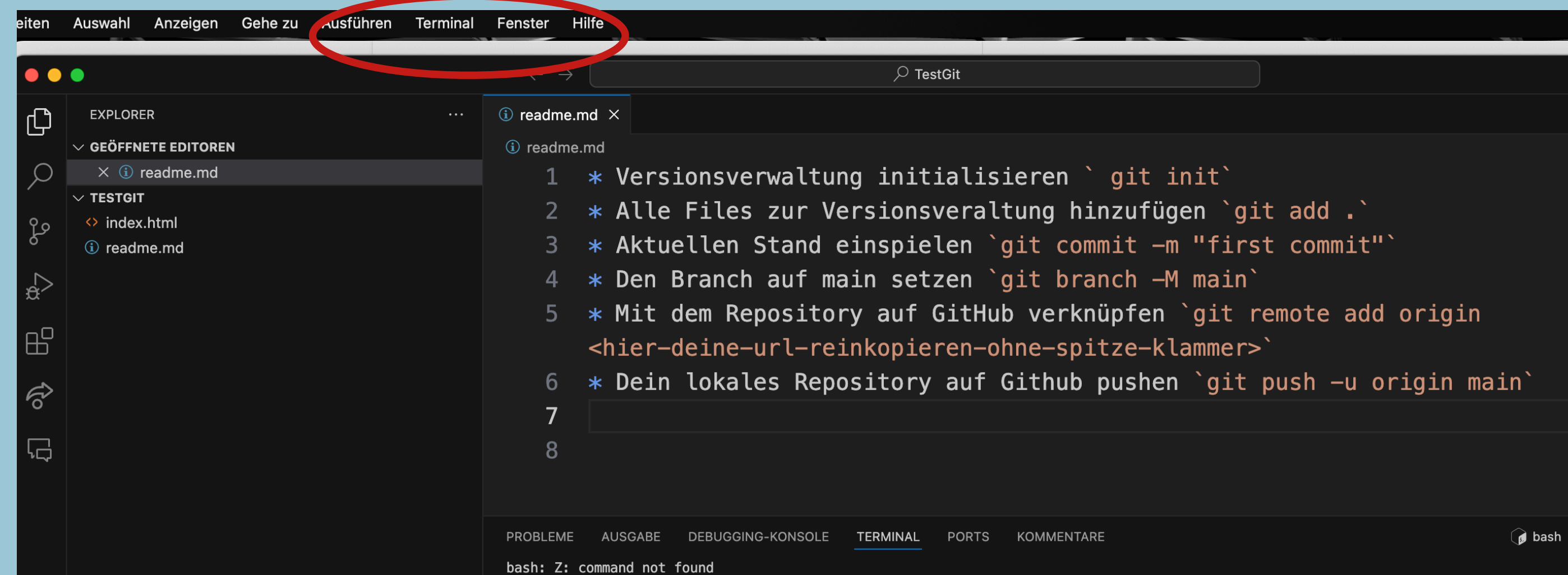
This will set `main` as the default branch. Change the default name in your [settings](#).

Create repository

GIT, GITHUB, VISUAL STUDIO CODE

2. dein lokales Projekt im VSCode öffnen, Terminal öffnen

Öffne deinen Folder, in den du bis jetzt gearbeitet hast (modul_webtech) im VisualStudioCode. Achte darauf, dass in diesem Folder bloss .html, .css, .js und sogenannte Assets, also Bilder und Fonts sind – keine Entwürfe als InDesign Dateien etc. Öffne ein Terminal Fenster:



GIT, GITHUB, VISUAL STUDIO CODE

3. Authentifizieren (einmalig)

Weil du das erste Mal Git benutzt, musst du dich authentifizieren und ins Terminal deinen Github Namen und dein Github E-Mail eingeben. **Die Angaben in Anführungszeichen mit deinen Github Angaben ersetzen!**

Über das Menü Terminal kannst du im Visual Studio Code ein Terminalfenster öffnen.

Nach der Eingabe drückst du Enter. Du wirst nun auf ein Browserfenster geführt, auf dem du den Zugriff von Visual Studio Code auf das Github Repo bestätigst. Nachher kannst du den Push fertig machen.

```
$ git config --global user.name "John Doe"  
$ git config --global user.email "johndoe@example.com"
```

GIT, GITHUB, VISUAL STUDIO CODE

4. Projekt initiieren, verknüpfen und auf den GitHub Service laden

```
`git init`
```

```
`git add .`
```

```
`git commit -m "first commit"`
```

```
`git branch -M main`
```

```
`git remote add origin <hier-deine-url-reinkopieren-ohne-spitze-klammer>`
```

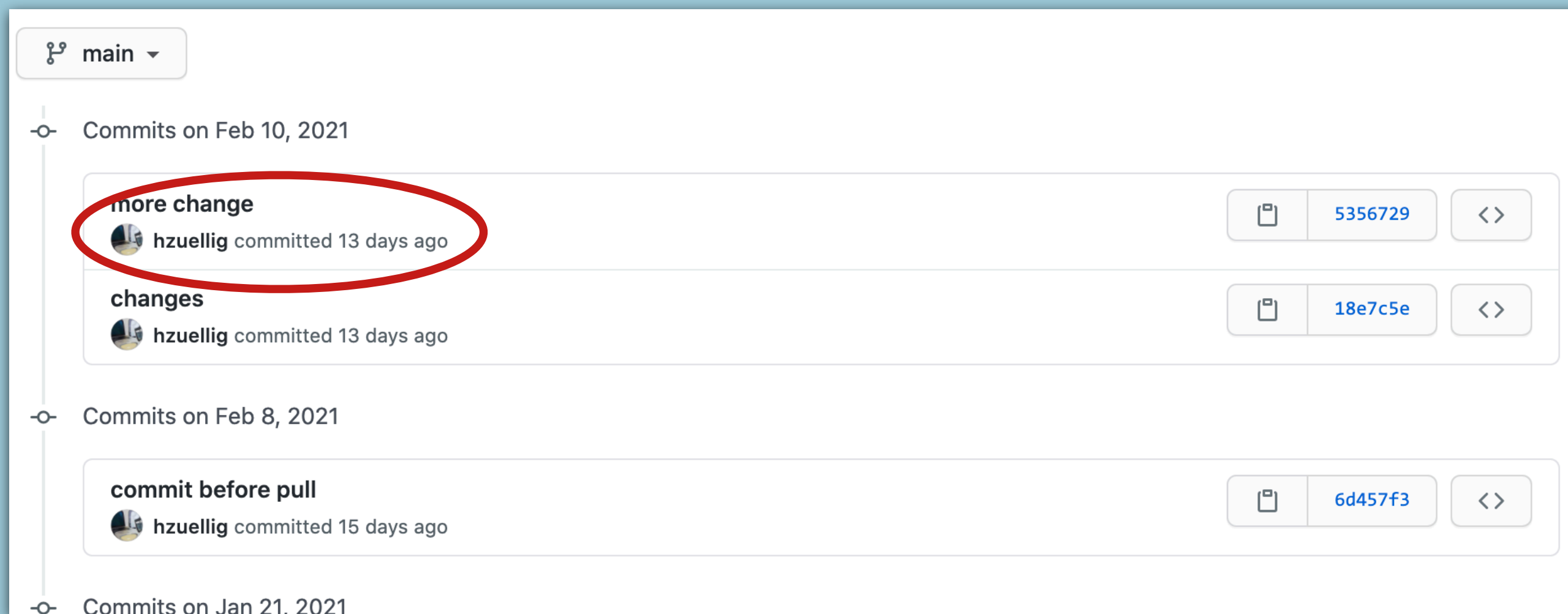
```
`git push -u origin main`
```

Kopiere die Befehle ohne diese Anführungszeichen ``!!
Die URL zu deinem Repo für den 5. Befehl findest du auf GitHub unter dem grünen Button Code

GIT, GITHUB, VISUAL STUDIO CODE

5. Commits prüfen

Über commits kannst du auf Github prüfen, ob den push erfolgreich war.



GIT, GITHUB, VISUAL STUDIO CODE

6. Workflow

Wenn du mir anderen arbeitest, solltest du dir folgenden Workflow angewöhnen:

```
commit  
pull  
push
```

Mit **commit** spielst du deine Änderungen in deine lokale Versionsverwaltung ein.

Mit **pull** holst du dir den aktuellen Stand vom Server.
So verhinderst du merge Konflikte mit Anpassungen von anderen.

Mit **push** spielst du deine Änderungen auf den Server.

GIT, GITHUB, VISUAL STUDIO CODE

6. Workflow

Wenn du mir anderen arbeitest, solltest du dir folgenden Workflow angewöhnen:

```
commit  
pull  
push
```

Mit **commit** spielst du deine Änderungen in deine lokale Versionsverwaltung ein.

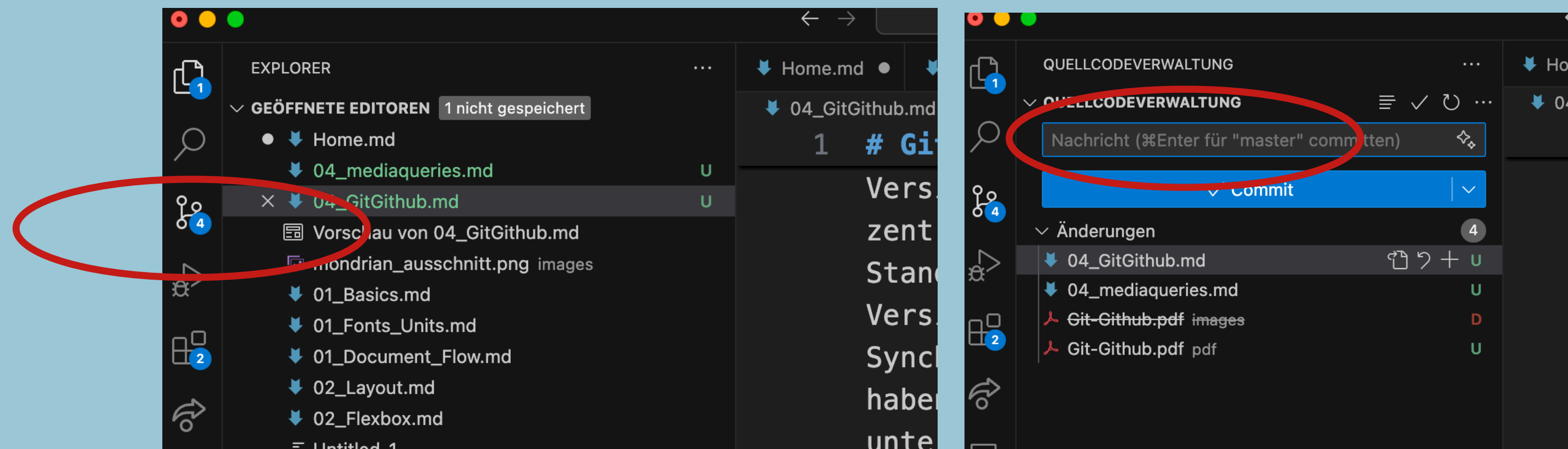
Mit **pull** holst du dir den aktuellen Stand vom Server.
So verhinderst du merge Konflikte mit Anpassungen von anderen.

Mit **push** spielst du deine Änderungen auf den Server.

GIT, GITHUB, VISUAL STUDIO CODE

6. Workflow auf VSCode

Das Icon mit den Ästen (Branches) zeigt dir die Anzahl Änderungen an. Klick darauf und du kommst zu den Menüs für commit, pull, push. Im **commit** Message Fenster gibst du die Nachricht ein, die kurz informiert, was du hier einspielst.



GIT, GITHUB, VISUAL STUDIO CODE

6. Workflow auf VSCode

Sicherer als den blauen Button synchronisieren zu drücken, ist es, das Menü unter den drei Punkten zu öffnen und zuerst **pull** und dann **push** auszuführen.

