

1



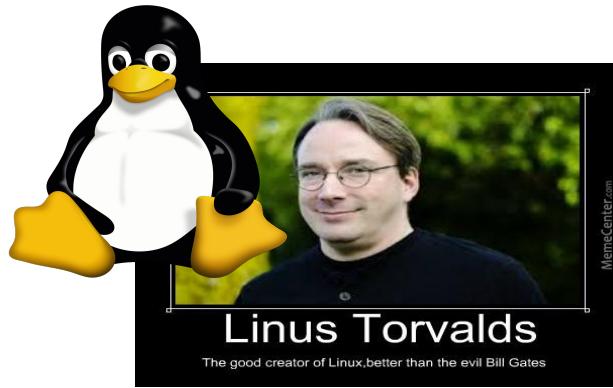
# Git und GitHub

Angewandte Generative KI und Prompt Engineering  
im Forschungsprojektmanagement  
(Projektmanagement)

Dr. Christopher Pollin  
<https://chpollin.github.io> | [christopher.pollin@dhcraft.org](mailto:christopher.pollin@dhcraft.org)  
Digital Humanities Craft OG  
[www.dhcraft.org](http://www.dhcraft.org)

Slides were generated AI-assisted.  
Images are partly AI-generated.





*"I'm an egotistical bastard, and I name all my projects after myself. First 'Linux', now 'Git'."*

– Linus Torvalds, <https://de.wikipedia.org/wiki/Git>  
[Git ~ Blödmann]



**Git** ist ein freies, verteiltes  
Versionsverwaltungssystem für Dateien.



# GitHub



Datenkrake  
(Microsoft)

## **Lernziele**

Die Grundlagen von Git verstehen.

GitHub Desktop kennenlernen und grundlegende Workflows anwenden.

Projektmanagement-Funktionen in GitHub nutzen (Issues, Labels, etc.) und Best Practices erlernen.

Möglichkeiten der Dokumentation in GitHub nutzen und Best Practices kennenlernen.

Ein Verständnis dafür entwickeln, wie Git und GitHub den Projektmanagementprozess in verschiedenen Phasen unterstützen können.

*[Schwerpunkt nicht auf Programmierer\*innen, sondern DH Projekte]*

# Git und seine Grundprinzipien

- **Open HPI: Let's Git - Versionsverwaltung und OpenSource**, <https://open.hpi.de/courses/git2020>  
*[Sehr guter und niederschwelliger Onlinekurs]*
- Blischak, J. D., Davenport, E. R., & Wilson, G. (2016). **A quick introduction to version control with Git and GitHub**. PLoS computational biology, 12(1), e1004668.  
*[Paper für Übersicht]*
- **Git and GitHub Introduction**, [https://www.w3schools.com/git/git\\_intro.asp?remote=github](https://www.w3schools.com/git/git_intro.asp?remote=github)  
*[Tutorial]*
- **Git and GitHub Tutorial For Beginners** | Full Course [2021] [NEW],  
[https://www.youtube.com/watch?v=3fUbBnN\\_H2c&ab\\_channel=Amigoscode](https://www.youtube.com/watch?v=3fUbBnN_H2c&ab_channel=Amigoscode)  
*[Youtube-Einführung, aber für Programmierer\*innen]*



# Begriffe

## Was ist eine Datei?

z.B. Text (.txt), Bild (.jpg), Musik (.mp3), TEI XML (.xml), JavaScript (.js) ...



## Was ist eine Version?

Eine Version beschreibt den Inhalt einer Datei zu einem bestimmten Zeitpunkt.

Die Änderung einer Datei erzeugt eine neue Version.

```
<?xml version="1.0" encoding="UTF-8"?>
<memo>
  <body>
    <p>Yoda says,
      “For your support thank I must!”
    </p>
  </body>
</memo>
```



Version 1

```
<?xml version="1.0" encoding="UTF-8"?>
<memo>
  <body>
    <p>Yoda says,
      “For your support thank I must!”
    </p>
  </body>
</memo>
```



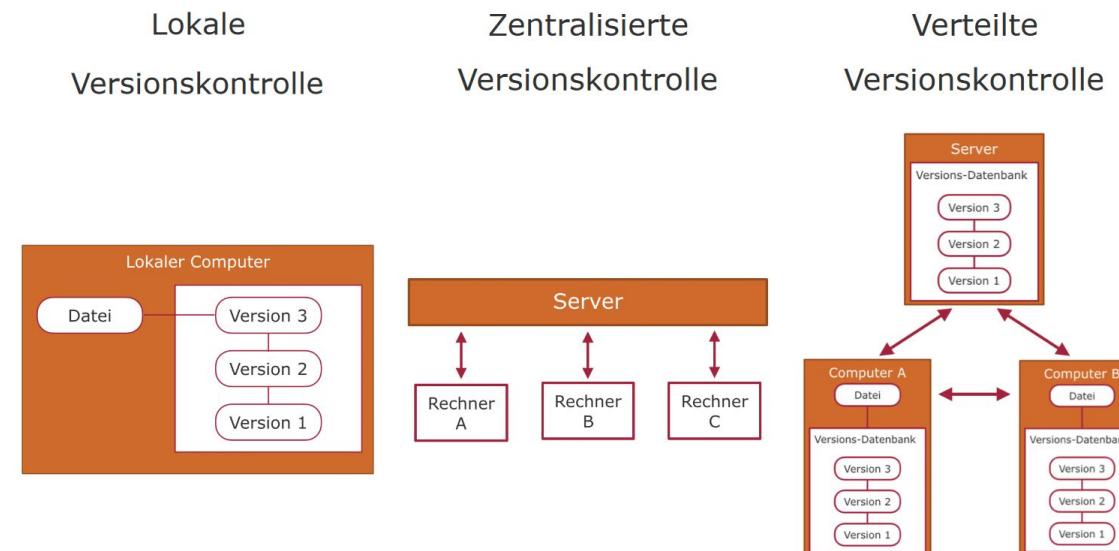
Version 2

# Begriffe

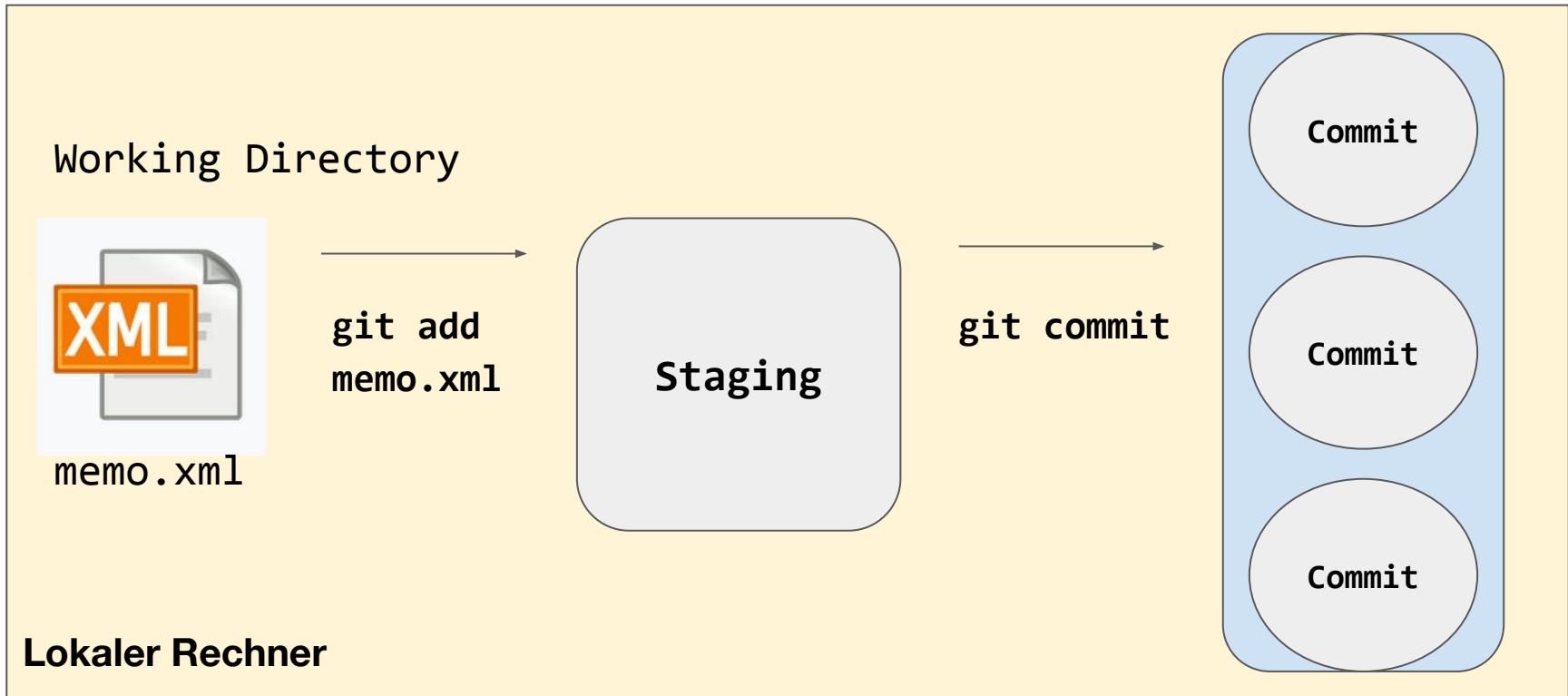
## Was ist Versionskontrolle?

Ermöglicht die Verwaltung verschiedener Versionen von Dateien.

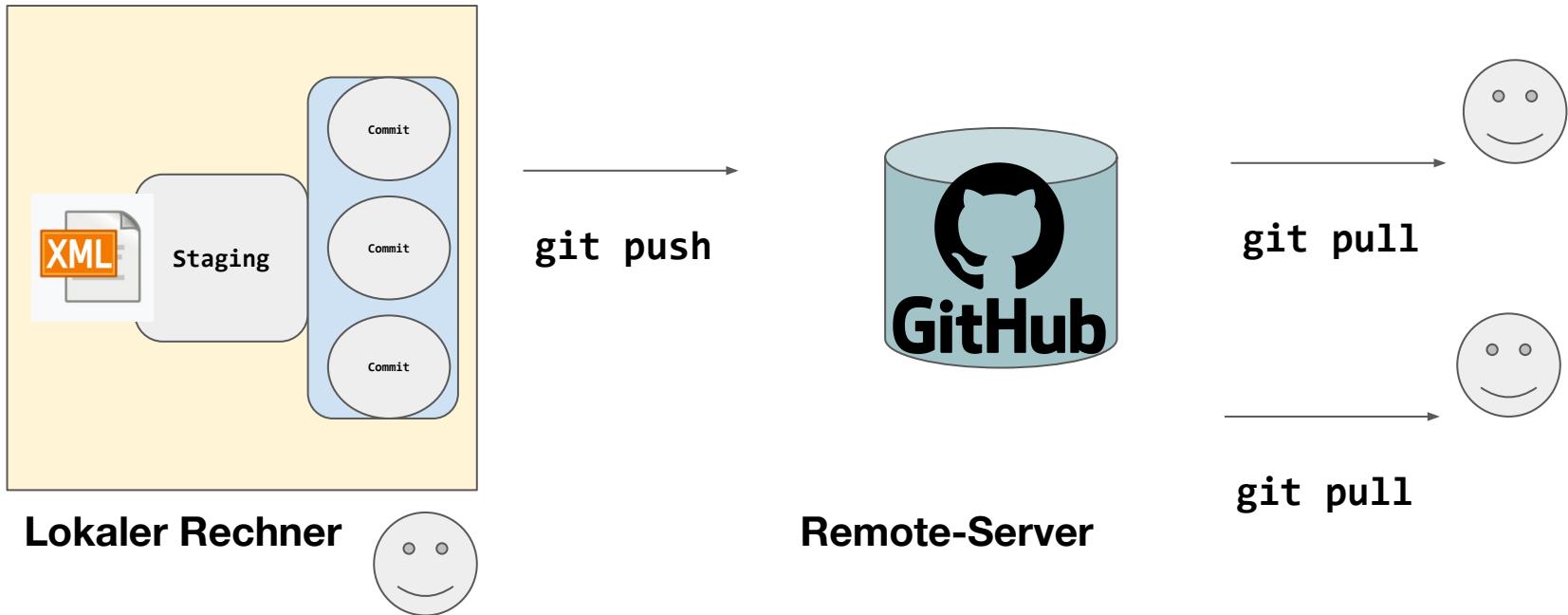
Erlaubt einen “Blick zurück” durch die Versionen von Dateien.



# Git ist ein freies, verteiltes Versionsverwaltungssystem für Dateien



# Lokaler Rechner und Remote-Server



# Git Befehle

## \$ git add

Datei (in ihrem gegenwärtigen Zustand) zum nächsten Commit hinzufügen (Staging)

## \$ git commit

Alles was in Staging ist, kommt als Commit in das Repository (Commit History)

## \$ git push

Wird verwendet, um Inhalte aus einem lokalen Repository in ein Remote-Repository hochzuladen.

## \$ git clone

Über eine URL ein neues Repository kopieren.

## \$ git pull

Der pull-Befehl wird verwendet, um Inhalte aus einem Remote-Repository herunterzuladen und unverzüglich das lokale Repository zu aktualisieren, damit die Inhalte übereinstimmen.

## \$ git fetch

Damit werden Commits, Dateien und Verweise aus einem Remote-Repository in das lokale Repository heruntergeladen.

## \$ git merge

Die heruntergeladenen Dateien etc. werden mit den lokalen zusammengeführt.

# **Grundprinzipien von Git**

1. Snapshots statt Unterschiede
2. Fast jede Operation ist lokal
3. Git stellt Integrität sicher
4. Git fügt im Regelfall nur Daten hinzu
5. Die drei Zustände (modified, staged, committed)

<https://git-scm.com/book/de/v2/Erste-Schritte-Was-ist-Git%3F>

# Arbeiten mit GitHub

- **GitHub Desktop Quick Intro For Windows**, <https://www.youtube.com/watch?v=77W2JSL7-r8>  
*[Kompaktes Einführungsvideo auf Youtube für GitHub Desktop]*
- **An Intro to Git and GitHub for Beginners (Tutorial)**,  
<https://product.hubspot.com/blog/git-and-github-tutorial-for-beginners>  
*[Command line only]*
- **Git for Professionals Tutorial - Tools & Concepts for Mastering Version Control with Git**,  
[https://www.youtube.com/watch?v=Uszj\\_k0DGsa](https://www.youtube.com/watch?v=Uszj_k0DGsa)

# Command Line vs. GUI Clients

```
Already up to date.  
chpollin@21-NF326:/mnt/c/Users/pollin/Documents/GitHub/SZD$ git --version  
git version 2.25.1  
chpollin@21-NF326:/mnt/c/Users/pollin/Documents/GitHub/SZD$ git pull  
Already up to date.  
chpollin@21-NF326:/mnt/c/Users/pollin/Documents/GitHub/SZD$ |
```

<https://docs.gitlab.com/ee/gitlab-basics/start-using-git.html>

- **GitHub Desktop**  
**Platforms:** Mac, Windows  
**Price:** Free
- **Git Extensions**  
**Platforms:** Linux, Mac, Windows  
**Price:** Free
- **TortoiseGit**  
**Platforms:** Windows  
**Price:** Free
- **GitKraken**  
**Platforms:** Linux, Mac, Windows  
**Price:** Free / \$29 / \$49

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

# GitHub Desktop

The screenshot shows the GitHub Desktop application interface. At the top, there's a navigation bar with File, Edit, View, Repository, Branch, and Help. Below the navigation bar, the current repository is set to "PM\_DH22" and the current branch is "main". A "Fetch origin" button indicates "Never fetched".

The main area is divided into two tabs: "Changes" and "History". The "History" tab is selected, showing a single commit:

- adding test.txt** (commit b6c42c1 by Christopher Pollin 2 years ago)
- No branches to compare

On the left side, there's a sidebar listing recent activities:

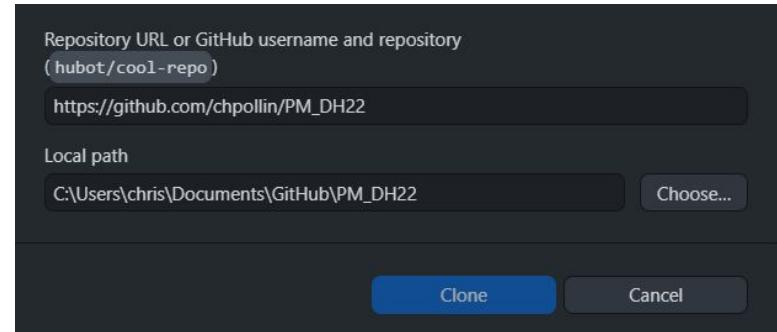
- adding test.txt (by Christopher Pollin 2 years ago)
- remove old files (by chpollin 2 years ago)
- init (by chpollin 2 years ago)

The commit details show "1 changed file" named "test.txt". The diff view shows the content "Hello Git!".

<https://desktop.github.com/download>

# Gemeinsames Hands-On: “mein memo.xml”

- Erzeugen Sie einen neuen Ordner mit dem Namen GitHub.  
(Z.B.: C:\Users\pollin\Documents\GitHub)
- Legen Sie innerhalb des Ordners GitHub einen Ordner “PM\_DH22” an.
- Öffnen Sie GitHub Desktop.
- Clonen Sie das Repository [https://github.com/chpollin/PM\\_DH22](https://github.com/chpollin/PM_DH22)
  - das Repository heißt “PM\_DH22”.
  - das entspricht dem Befehl  
**\$ git clone**
- In unserem neuen Ordner befinden sich jetzt alle Daten - lokal bei uns - aus dem Repository.



## Gemeinsames Hands-On: “mein memo.xml”

- Erstellen Sie eine neue XML bzw. verändern Sie helloworld\_template.xml, benennen Sie es um und speichern Sie es - Datei mit folgender Struktur und Ihrem Vornamen.

```
<?xml version="1.0" encoding="UTF-8"?>
<memo>
  <body>
    <p>Hello Git from Christopher</p>
  </body>
</memo>
```

- Legen Sie die Datei mit dem Dateinamen helloworld\_VORNAME.xml in Ihrem lokalen Ordner PM\_DH22\xml ab.

Name	Änderungsdatum	Typ	Größe
helloworld_christopher	04.07.2022 13:38	XML Document	1 KB
helloworld_template	04.07.2022 13:38	XML Document	1 KB

Wenn die neue Datei angelegt wurde, erscheinen folgende Änderungen in GitHub Desktop auf.

Jeder Commit muss immer eine Nachricht beinhalten. Damit soll ausgedrückt werden, was Sie gerade bearbeitet oder hinzugefügt haben.

Die Dateien befinden sich in der Staging Area.

Mit dem Button “Commit to main” wird git commit aktiviert und der Commit kommt in die Commit History.

The screenshot shows the GitHub Desktop application interface. At the top, the menu bar includes File, Edit, View, Repository, Branch, and Help. The repository dropdown shows "Test-Training-Repo" and the branch dropdown shows "main". A "Fetch origin" status indicates it was last fetched 3 minutes ago. The main area has tabs for "Changes" (selected) and "History". Under "Changes", two files are listed with checkboxes: "xml\\helloworld\_christopher.xml" and "xml\\helloworld\_template.xml". The "xml\\helloworld\_christopher.xml" file is selected. The right pane displays the XML content of this file, showing changes from line 1 to 6. Lines 1 through 5 are additions, and line 6 is a deletion. Below the XML content, a commit dialog box is open. It has fields for "Summary (required)" (containing "add template and file") and "Description". There is also a "Commit to main" button at the bottom. A dashed orange box highlights the "Summary" field, and an orange arrow points from the "Summary" field in the dialog to the "Summary" field in the commit dialog.

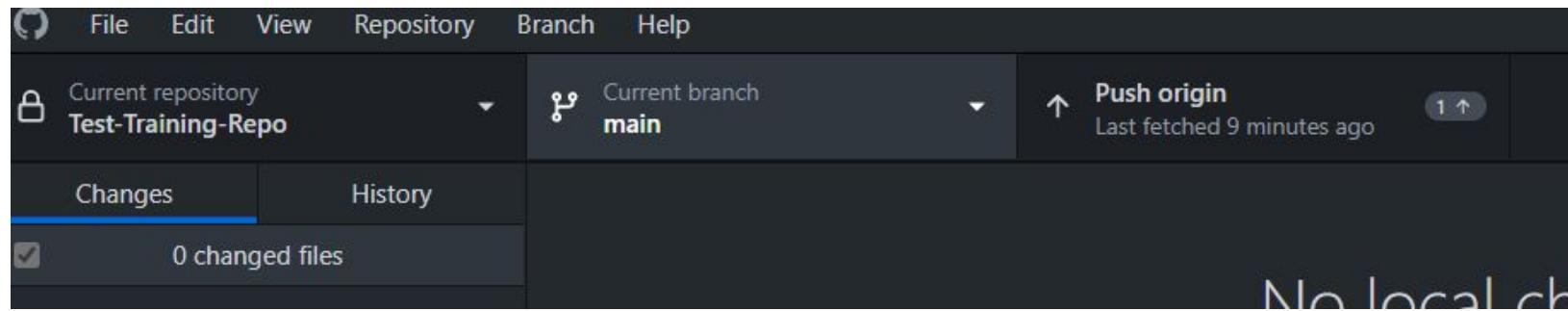
```
@@ -0,0 +1,6 @@
1 +<?xml version="1.0" encoding="UTF-8"?>
2 +<memo>
3 +  <body>
4 +    <p>Hello Git from Christopher</p>
5 +  </body>
6 +</memo>
```

add template and file

Summary (required)

Description

Commit to main



Jetzt muss das Ganze noch in das Repository (Test-Training-Repo) gepusht werden - git push aktiv.

Unsere Dateien sind jetzt auf dem Server.

to... / Pull requests Issues Marketplace Explore

Weimar / **Test-Training-Repo** Private

Watch 0 Forks

Pull requests Actions Projects Wiki Security Insights

main Test-Training-Repo / xml /

Go to file Add file ...

chpollin add template and file 5eb8de5 4 minutes ago History

..

helloworld\_christopher.xml add template and file 4 minutes ago

helloworld\_template.xml add template and file 4 minutes ago

Current repository  
**Test-Training-Repo**

Changes History

No branches to compare

**add template and file**  
Christopher Pollin • 3 minutes ago

**Create DDL\_GNM\_MAIN.sql**  
KSWmanager • 3 days ago

**Update Museumsdatenbank.dpr**  
KSWmanager • 3 days ago

**Create Museumsdatenbank.dpr**  
KSWmanager • 3 days ago

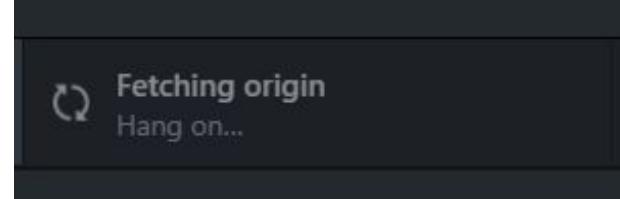
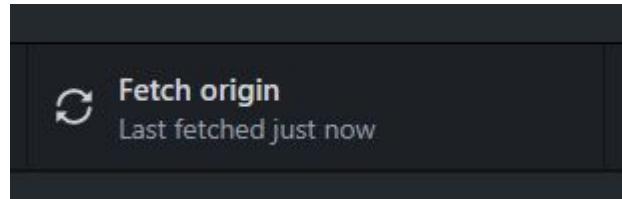
**Create main.cs**  
KSWmanager • 3 days ago

**Initial commit**  
KSWmanager • Jun 4, 2021

In der History in GitHub Desktop bzw. auf der Webseite  
[https://github.com/chpollin/PM\\_DH22/tree/main/xml](https://github.com/chpollin/PM_DH22/tree/main/xml)  
können wir die Commits einsehen.

```
$ git fetch
```

Gleichen Sie ab: Gibt es was Neues bei uns lokal? Gibt es etwas Neues im Repository am Server?



Bearbeiten Sie nun das XML-Dokument von Ihrer/Ihrem alphabetisch nächsten Kollegin/Kollegen. Fügen Sie eine Kleinigkeit hinzu, wie z.B. ein zweites <p> mit dem Text “Greetings from Christian”.

```
<?xml version="1.0" encoding="UTF-8"?>
<memo>
  <body>
    <p>Hello Git from Christopher</p>
    <p>Greetings from Christian</p>
  </body>
</memo>
```

Es könnte jetzt zu einem Konflikt gekommen sein, weil zwei Personen gleichzeitig dieselbe Datei bearbeitet haben.

*"Git can handle most merges on its own with automatic merging features. A **conflict arises when two separate branches have made edits to the same line in a file, or when a file has been deleted in one branch but edited in the other.** Conflicts will most likely happen when working in a team environment."*

The screenshot shows a Git commit interface with the following details:

- Repository:** Test-Training-Repo
- Branch:** main
- Last fetched:** Just now
- Changes:** 1 changed file
- File:** xml\helloworld\_christopher.xml
- Diff View:** Shows a conflict at line 5. The original code was "<p>Hello Git from Christopher</p>". The code from Christian added a new line: "<p>Greetings from Christian</p>".
- Commit Message:** adding "Greetings from Christian"
- Description:** (empty)
- Commit Buttons:** Commit to main



Search or jump to...



Pull requests Issues Marketplace Explore

KlassikStiftungWeimar / Test-Training-Repo

Private

Watch 0

Fork 0

Code Issues Pull requests Actions Projects Wiki Security Insights

main

1 branch

0 tags

Go to file

Add file

Code



chpollin adding "Greetings from Christian"

27c7f65 22 minutes ago 7 commits



xml

adding "Greetings from Christian"

22 minutes ago



DDL\_GNM\_MAIN.sql

Create DDL\_GNM\_MAIN.sql

3 days ago



Museumsdatenbank.dpr

Update Museumsdatenbank.dpr

3 days ago



README.md

Initial commit

12 months ago



main.cs

Create main.cs

3 days ago

README.md



## Test

Test

### About

Test

Readme

0 stars

0 watching

0 forks

### Releases

No releases published

Create a new release

### Packages

No packages published

Publish your first package

Contributors 2

# “The Perfect Commit”

- Jeder Commit gehört zu einem Thema. Themen sollten nicht in den gleichen Commits gemischt werden.
  - Commit: Annotation der Personen im XML/TEI
  - Überarbeitung des msDesc im teiHeader
  - Das XSLT zur Normalisierung von X

→ nur die Dateien ankreuzen, die zum Thema passen.

→ gemeinsame Regeln definieren, was ein Commit ist.
- Commit-Message
  - Kleine Zusammenfassung
  - “Kurz und knackig” & effektiv
  - Ungewöhnliches und Wichtiges explizieren

→ gemeinsame Regeln definieren, wie eine Commit-Message aufgebaut sein muss.

# GitHub als Software Engineering Tool

- **GitHub Projects Demo: Automation, Kanban, Scrum, Issues, Pull Request, Milestones, Issues, Tasks,** <https://www.youtube.com/watch?v=YxKhb3fxtsU>
- **GitHub Issues,** <https://docs.github.com/en/enterprise-server@3.5/issues>
- **Labels and Milestones,**  
<https://docs.github.com/en/enterprise-server@3.5/issues/using-labels-and-milestones-to-track-work/managing-labels>
- **Managing Project Boards,**  
<https://docs.github.com/en/enterprise-server@3.5/issues/organizing-your-work-with-project-boards/managing-project-boards/about-project-boards>

# Issues

## Beispiel Issue #3

[Open](#)

1 of 3 tasks

chpollin opened this issue on Jun 14 · 1 comment



chpollin commented on Jun 14 • edited

...

### Beispiel Issue

Hier ist die Beschreibung der Task, die bearbeitet werden muss.

Diese könnte man in weitere, kleinere Aufgaben unterteilen und schön mit einer task list dokumentieren

- Christian macht X im teilHeader
- @chpollin macht Y im body
- zusammenführen

Auch Screenshots können helfen Dinge zu visualisieren. Unter Windows einfach mit dem Snipping Tool einen Screenshot erstellen und mit strg-v hier direkt einfügen.

```
</msIdentifier>
</msDesc>
<listBibl>
    <bibl xml:id="bibl.1"><title>Lorem Ipsum 1</title></bibl>
    <bibl xml:id="bibl.2"><title>Lorem Ipsum 2</title></bibl>
    <bibl xml:id="bibl.3"><title>Lorem Ipsum 3</title></bibl>
</listBibl>
</sourceDesc>
```

<https://github.com/KlassikStiftungWeimar/Test-Training-Repo/issues/3>

# Milestones

The screenshot shows a GitHub milestones page for a project. At the top, there are navigation links: Issues (6), Pull requests, Actions, Projects, Wiki, Security, and Insights. Below these, there are two tabs: Labels and Milestones, with 'Milestones' being the active tab. The main title is 'TEI Model'. A progress bar indicates the milestone is 33% complete, due by July 11, 2022. The description is 'Planung und Konzeption des TEI Models für X.' Below the description, there is a list of tasks:

- ⏪ 2 Open ✓ 1 Closed
- ⏪ TEI Model: Reading  
#6 opened 4 minutes ago by chpollin 3 of 4 tasks
- ⏪ TEI Model: Prototype priority  
#7 opened 3 minutes ago by chpollin

At the bottom, a tip is displayed: ⚡ Tip! You can use `shift + j` or `shift + k` to move items with your keyboard.

[https://github.com/chpollin/PM\\_DH22/milestone/1](https://github.com/chpollin/PM_DH22/milestone/1)

## **Beispiele**

Consulting Schönberg Edition

<https://github.com/chpollin/Schoenberg/issues>

Interne Arbeit in DHCraft

<https://github.com/DigitalHumanitiesCraft/Projects/issues>

Stefan Zweig digital

<https://github.com/chpollin/SZD/issues>

# Projects

[KlassikStiftungWeimar / Test-Training-Repo](#) Private

[Code](#) [Issues 6](#) [Pull requests](#) [Actions](#) [Projects 1](#) [Wiki](#) [Security](#) [Insights](#)

[test](#)  
Updated now

[todo](#) + ...

[add new <p> to helloworld\\_christopher.xml](#) ...  
#9 opened by chpollin

[TEI Model: Reading](#) ...  
3 of 4 tasks  
#6 opened by chpollin  
↳ TEI Model

[in progress](#) + ...

[Beispiel Issue](#) ...  
1 of 3 tasks  
#3 opened by chpollin

[done](#) + ...

[Filter cards](#)

[+ Add cards](#)

[is:open](#)  
You can use the filters available in issue search.

**Search results**

- [TEI Model: Prototype](#)  
#7 opened by chpollin  
priority  
↳ TEI Model
- [teiBody: normalize <persName>](#)  
#5 opened by chpollin  
teiBody
- [teiHeader: add msDesc](#)  
#4 opened by chpollin  
teiHeader

<https://github.com/KlassikStiftungWeimar/Test-Training-Repo/projects/1>

# Hands-On: Issue anlegen

- Legen Sie ein Issue an und assignen Sie jemanden, der/die eine Änderung in einem der vielen helloworld\_XXX.xml machen soll.
- Fügen Sie ein Label und einen Milestone hinzu.
- Die zugeordnete Person führt den Issue aus, committet die Änderung und referenziert mit #ISSUE ID auf das zugehörige Issue.
- Pushen Sie die Änderung.
- Fügen Sie ein Comment hinzu und schließen Sie das Issue.

add new <p> to helloworld\_christopher.xml #9

 Open chpollin opened this issue now · 0 comments

chpollin commented now

[https://github.com/KlassikStiftungWeimar/Test-Training-Repo/blob/main/xml/helloworld\\_christopher.xml](https://github.com/KlassikStiftungWeimar/Test-Training-Repo/blob/main/xml/helloworld_christopher.xml)

chpollin self-assigned this now

Current repository  
Test-Training-Repo

Changes 1 History

1 changed file

xml\helloworld\_christopher.xml

Update helloworld\_christopher.xml

#9

R+

Commit to main

## **“The best Issue”**

Simpel, prägnant und beschreibend

Einheitliche Formatierung

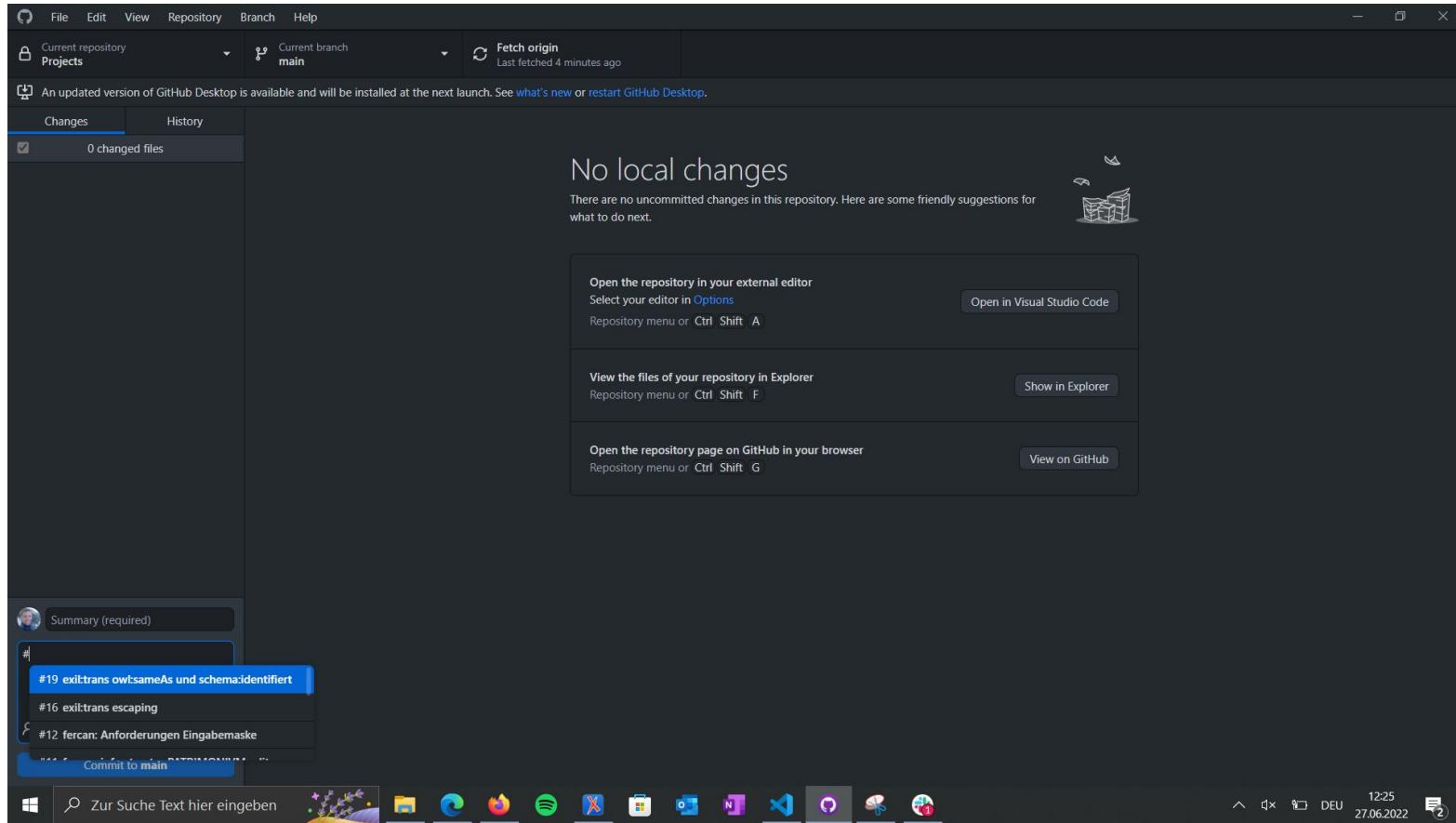
Bearbeitbar / Durchführbar

Relevant

Kontext

Problem oder Idee

Lösung und nächste Schritte



- <https://github.com/DigitalHumanitiesCraft/Projects/commit/5e9e5c05eecc19fa2eca9d45456ac3f89b29e8c5>
- <https://github.com/DigitalHumanitiesCraft/Projects/issues/18>

# GitHub und Dokumentation

- **Introduction to Markdown**, <https://www.geeksforgeeks.org/introduction-to-markdown/>
- **Markdown Cheat Sheet. A quick reference to the Markdown syntax**,  
<https://www.markdownguide.org/cheat-sheet/>
- **Markdown Tips & Tricks 2022 - Markdown Crash Course**,  
<https://www.youtube.com/watch?v=ftOBvusMHjQ>
- **Markdown Crash Course**, <https://github.com/codeSTACKr/markdown-crash-course>

# Markdown

cheatsheet.md [C:\Users\pollin\Documents\GitHub\KlassikStiftungWeimar\Test-Training-Repo\documents\cheatsheet.md] - <oXygen/> XML Editor (Academic use only)

File Edit Find Project Options Tools Document Window Help

XPath 2.0 Execute XPath on 'Current File' |  ABC |    | H1 H2 H3 — | B I S | {} » | ⌂ |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | <img alt="Image

# Überschriften, Text, ungeordnete Liste, Code Block

The image shows a code editor window with a dark theme. On the left, there is a file named 'cheatsheet.md' containing the following Markdown code:

```
1 # Heading 1
2
3 Text
4
5 * item
6 * item
7 * item
8
9 ``
10 <memo>
11   <from>Christopher</from>
12   <to>Christian</to>
13 </memo>
14 ``
15
16 ## Heading 2
17
18 ### Heading 3
19
20
```

On the right, the rendered output is displayed in a dark-themed browser or viewer. It includes:

- Heading 1**: A large, bold heading.
- Text**: A regular paragraph.
- Unordered List**: A list of three items.
- <memo>**: A block element containing XML-like content.
- Heading 2**: A second-level heading.
- Heading 3**: A third-level heading.

# Absätze, geordnete Liste, Schriftauszeichnung (kursiv & fett)

```
18 #### Heading 3
19
20 Lorem ipsum dolor sit amet, consetetur sadipscing
21 elitr, sed diam nonumy eirmod tempor
22 invidunt ut labore et dolore magna aliquyam
23 erat, sed diam voluptua.
24 At vero eos et accusam et justo
25 duo dolores et ea rebum.
26
27 Erst eine zusätzliche Zeile macht einen neuen Absatz.
28
29 ## Ordered List
30 1. *item 1 is italic*
31 2. **item 2 is bold**
32 3. ***item 3*** is italic and bold
33
34 Aber auch so:
35
36 1. item 1
37 1. item 2
38 3. item 3
```

## Heading 3

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum.

Erst eine zusätzliche Zeile macht einen neuen Absatz.

## Ordered List

1. *item 1 is italic*
2. **item 2 is bold**
3. ***item 3*** is italic and bold

Aber auch so:

1. item 1
2. item 2
3. item 3

# **Blockzitate, Links, Bilder, Tabellen, Inhaltsverzeichnis**

```
59 2. item 4
60
61
62
63
64
65
66
67
68
40
41 ### Blockquote
42
43 > some text
44 >
45 > some text
46
47 ### Links
48
49 Das ist ein Link: [text you see](https://de.wikipedia.org/wiki/Markdown)
50
51 Das ist ein interner Link: [Spring zu Heading 1](#heading1)
52
53 ![nur ein Beispiel](https://commons.wikimedia.org/wiki/File:Example_de.jpg "Beispielbild")
54
55 ## Extended Syntax
56
57 ### Tabelle
58
59 | Packages | Description | Version |
60 | :--- | :---: | ---: |
61 | React | JavaScript Framework | v18.0 |
62 | Next.js | React Framework | v12.0 |
63
64
65
66
67
68
```

## Blockquote

some text

some text

## Links

Das ist ein Link: [text you see](#)

Das ist ein interner Link: [Spring zu Heading 1](#)

nur ein Beispiel

## Extended Syntax

## Tabelle

Packages	Description	Version
React	JavaScript Framework	v18.0
Next.js	React Framework	v12.0

# **README.md zur Dokumentation**

The screenshot shows a dark-themed GitHub repository interface. At the top left, there's a navigation bar with three horizontal lines and the text "README.md". On the far right is a small edit icon. The main content area contains the following text:

**Ich bin ein ReadMe.md**

---

Liegt ein ReadMe.md in einem Ordner in einem Git Repo, dann wird es besonders auf GitHub dargestellt. Und zwar hier als dieser Text der direkt unter "Code" angezeigt wird.

**Header 2**

---

Dieses ReadMe.md nutzt man für die Dokumentation von Code etc.

**Header 3**

[https://github.com/chpollin/PM\\_DH22](https://github.com/chpollin/PM_DH22)

About READMEs,

<https://docs.github.com/en/repositories/managing-your-repositorys-settings-and-features/customizing-your-repository/about-readmes>

# Wiki: Beispiel Kodierungsregeln

The screenshot shows a GitHub Wiki page titled "Kodierungsregeln". The page content includes sections for "Entitäten (durch Registerfunktion)" and "Hervorhebungen", each containing a bulleted list of XML-like code snippets with annotations. To the right, a sidebar displays a navigation menu with sections for "Pages" (2), "Home", and "Kodierungsregeln", which further branches into sub-sections like "Entitäten (durch Registerfunktion)", "Hervorhebungen", etc.

Issues 5 Pull requests Discussions Actions Projects Wiki Security Insights Settings

## Kodierungsregeln

phkehrer edited this page on Mar 3 · 46 revisions

Edit

### Entitäten (durch Registerfunktion)

- Datum: <date when="1900-01-01" calendar="#Gregorian"/> [ediarum.Base]
- Personenname: <persName key="person.2">Emil Hertzka</persName> [ediarum.Base]
- Organisationsname: <orgName key="org.3">Kolisch-Quartett</orgName> [ediarum.Base]
- Ortsname: <placeName key="place.1.1">Wien</placeName> [ediarum.Base]
- Werktitel: <bibl sameAs="werke.31">Komm, Gott, Schöpfer, Heiliger Geist</bibl> [ediarum.Base]
- Sachbegriffe:

### Hervorhebungen

- Fett: <hi rendition="#b"> [ediarum.Base]
- Unterstrichen: <hi rendition="#u"> [ediarum.Base]
  - <hi rendition="#u">Bläser-Partituren-Stimmen</hi> [ediarum.Base]

Pages 2

Find a Page...

Home

Kodierungsregeln

- Entitäten (durch Registerfunktion)
- Hervorhebungen
- Streichungen
- Hinzufügungen
- Lesbarkeit
- Korrektur durch den Herausgeber
- Normalisierte Schreibung

<https://github.com/chpollin/Schoenberg/wiki/Kodierungsregeln>

About wikis, <https://docs.github.com/en/communities/documenting-your-project-with-wikis/about-wikis>

# Hands-On: Markdown

Erstellen Sie eine neue Wiki Page (mit ihrem Namen). Probieren Sie Markdown aus → Überschrift, Liste und eine Code Snippet (memo.xml).

The screenshot shows a GitHub repository interface for 'KlassikStiftungWeimar / Test-Training-Repo'. The repository is private. The top navigation bar includes links for Code, Issues (6), Pull requests, Actions, Projects (1), Wiki (highlighted with a red underline), Security, and Insights. The Wiki page itself is titled 'Home'. It displays a message: 'Welcome to the Test-Training-Repo wiki!'. Below this is a dashed box containing a '+ Add a custom footer' button. To the right, there's a sidebar with 'Pages 1' (with a dropdown arrow), a search bar ('Find a Page...'), and a 'Home' link. Another dashed box at the bottom right contains a '+ Add a custom sidebar' button.

KlassikStiftungWeimar / Test-Training-Repo Private

Watch 2 Fork 0 Star 0

Code Issues 6 Pull requests Actions Projects 1 Wiki Security Insights

Home

Christopher Pollin edited this page now · 1 revision

Welcome to the Test-Training-Repo wiki!

+ Add a custom footer

Pages 1

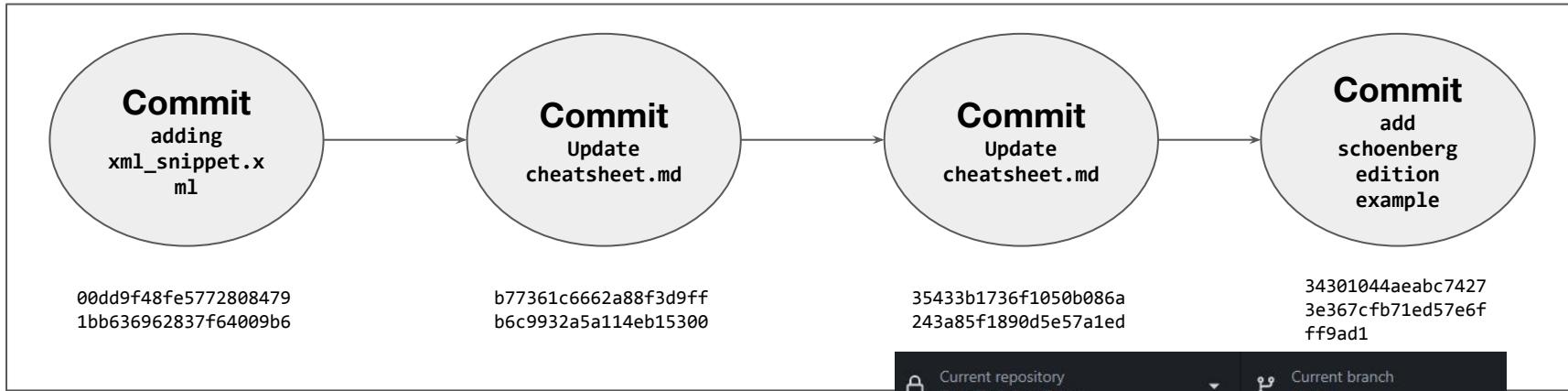
Find a Page...

Home

+ Add a custom sidebar

# **Git Branches**

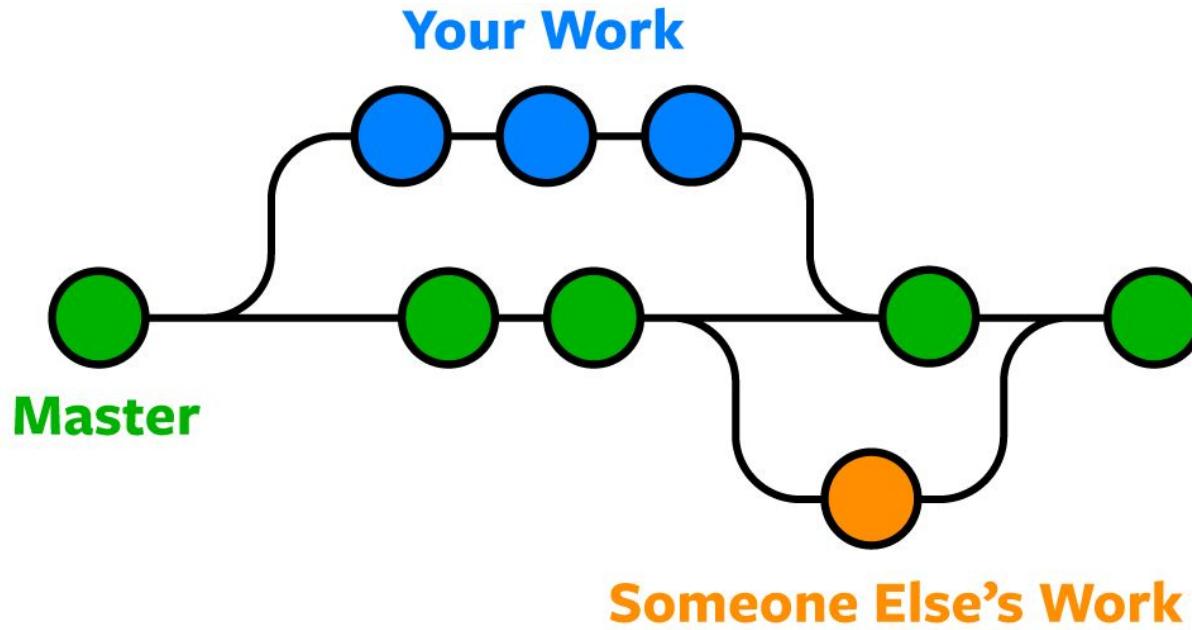
# Branches



main

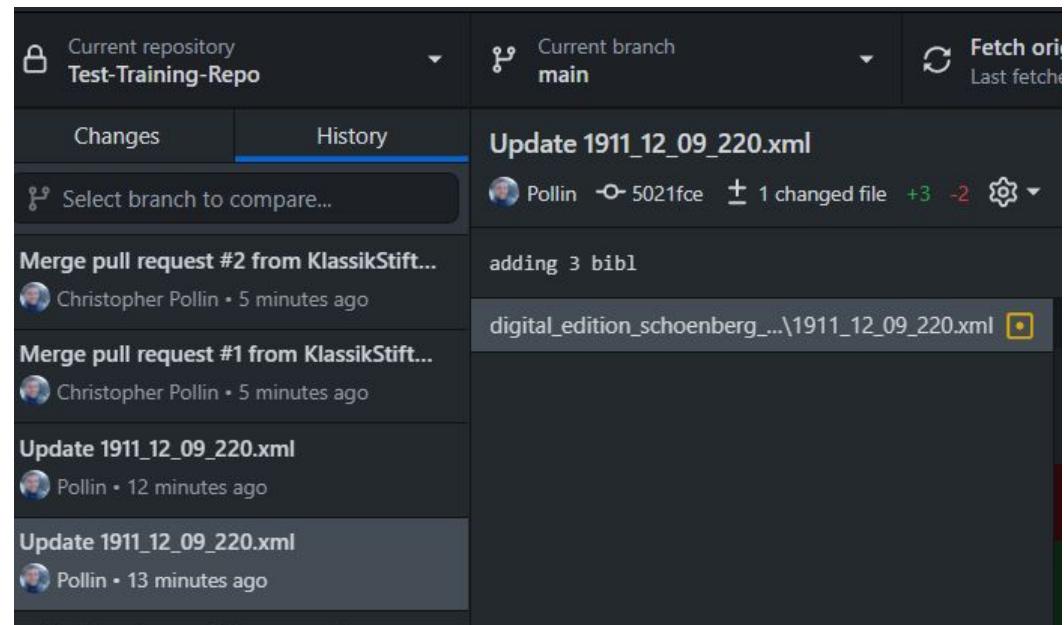
Changes	History
No branches to compare	
	<b>add schoenberg edition example</b>
	• Pollin • 8 hours ago
	<b>Update cheatsheet.md</b>
	• Pollin • 11 hours ago
	<b>Update cheatsheet.md</b>
	• Pollin • 11 hours ago
	<b>adding xml_snippet.jpg</b>
	• Pollin • 11 hours ago

# Branches



# Branches und Pull request

- 3 <bibl> in einer Datei im teiHeader branch hinzugefügt
- 1 fehlendes <p> im teiBody branch
- merge pull request 1 → 3 <bibl> aus teiHeader branch nach main
- merge pull request 2 → 1 <p> aus teiBody branch nach main



<https://github.com/KlassikStiftungWeimar/Test-Training-Repo/pulls>

## **Hands-On: eigener Branch**

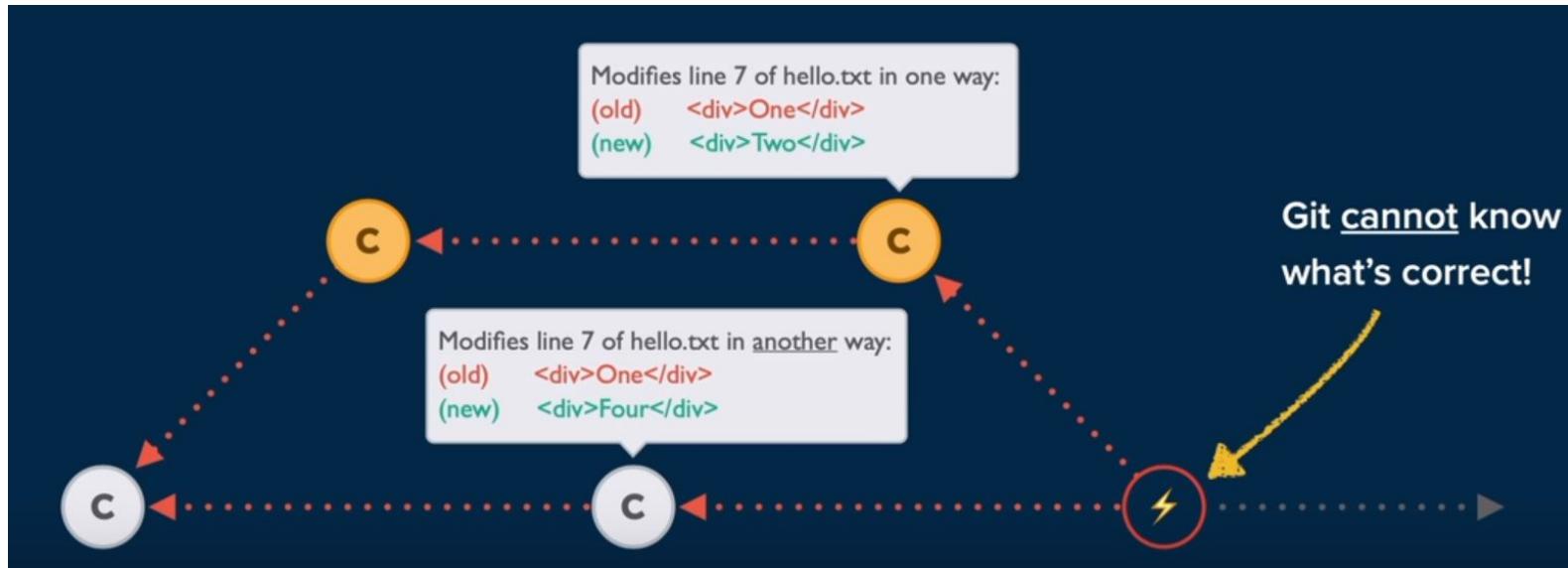
Erzeugen Sie einen neuen Branch mit Ihrem Namen (z.B.: “christopher”).

Nehmen Sie eine Änderung im XML/TEI vor:

[https://github.com/KlassikStiftungWeimar/Test-Training-Repo/tree/main/digital\\_edition\\_schoenberg\\_example/sample\\_letter](https://github.com/KlassikStiftungWeimar/Test-Training-Repo/tree/main/digital_edition_schoenberg_example/sample_letter)

Committen Sie “lustig drauflos”, damit jede/r einen Konflikt bekommt, den wir dann bereinigen (oder auch nicht) bereinigen können.

# Merge Conflicts



- About merge conflicts,  
<https://docs.github.com/en/pull-requests/collaborating-with-pull-requests/addressing-merge-conflicts/about-merge-conflicts>
- Git for Professionals Tutorial - Tools & Concepts for Mastering Version Control with Git, [https://www.youtube.com/watch?v=Uszj\\_k0DGsq](https://www.youtube.com/watch?v=Uszj_k0DGsq)

# Merge Conflicts

```
14 ▼      <ul>
15 <<<<<< HEAD
16 ▼      <li><a href="index.html">Home</a></li>
17 ▼      <li><a href="about.html">About Us</a></li>
18 ▼      <li><a href="product.html">Product</a></li>
19 ▼      <li><a href="imprint.html">Imprint</a></li>
20 =====|
21 >>>>> develop
22 ▲      </ul>
```

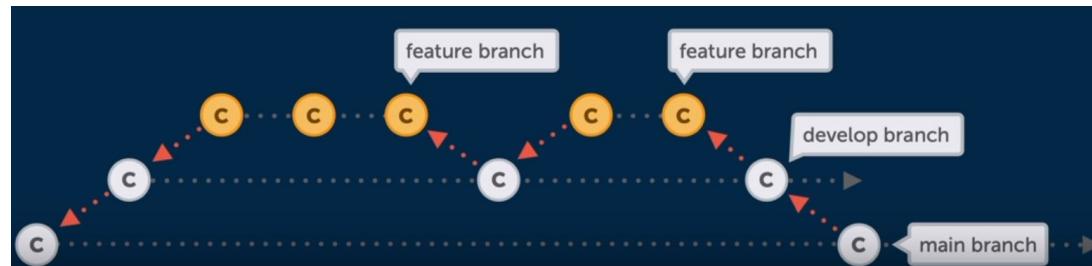
- About merge conflicts,  
<https://docs.github.com/en/pull-requests/collaborating-with-pull-requests/addressing-merge-conflicts/about-merge-conflicts>
- Git for Professionals Tutorial - Tools & Concepts for Mastering Version Control with Git, [https://www.youtube.com/watch?v=Uszj\\_k0DGsq](https://www.youtube.com/watch?v=Uszj_k0DGsq)

# “Branching Strategies”

## Mainline Development

- Immer integrierend sein → achten Sie darauf, dass Sie Ihre Inhalte gut, zügig und ordentlich mit der Arbeit Ihrer Kolleginnen/Kollegen zusammenführen.
- Wenig Branches
- Kleine Commits

## State, Releases, and Feature Branches



# Kritik an Git



Hadi Hariri  
@hhariri



Git is not a success story. Git is a failure as a system with a crap user experience that forces you to learn more about the tool you're using than about getting your work done.

2:20 PM · Oct 8, 2020



1.8K 626 people are Tweeting about this

Blogpost:

<https://hadihariri.com/2020/10/14/a-few-words-on-git/>

## Dinge, die wir nicht besprochen haben...

- **\$ git init**  
Um einen lokalen Ordner zu einem Git Repository zu machen.
- **Fork a repo**  
Eine eigene Kopie eines existierenden Repos, um es selbst weiter zu bearbeiten.
- **GitHub Actions**
- **weitere Tools (z.B. GitKraken)**

# Git bzw. GitHub installieren

## Distribution Ubuntu

Paketverwaltungstool: apt  
\$ apt-get install git

## macOS

Xcode Kommandozeilen-Tools  
\$ git --version

## Windows

Binärer Installationsassistent, <https://git-scm.com/download/win>

Oder GitHub Desktop <https://desktop.github.com/>