



git

Git Crashkurs

Versionsmanagement

Inhalte



01 **Einstieg**

Was ist Git?

02 **Repository**

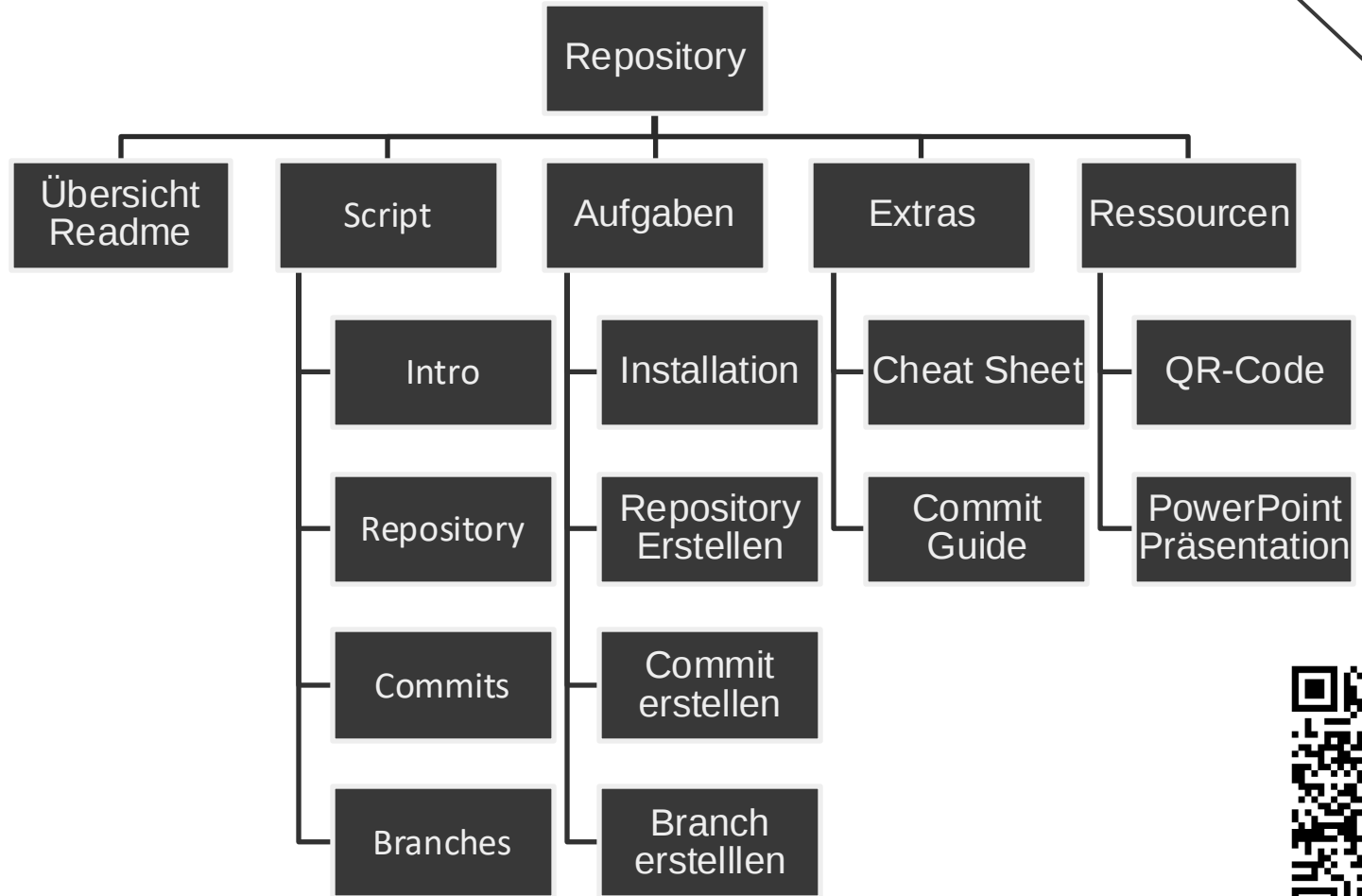
Wie erstellt man ein Repository?

03 **Commits**

Erstellen, pullen, pushen

04 **Branches**

Erstellen und mergen



01

Einstieg

Was ist Git?

Warum Versionskontrolle?



Zusammenarbeit

Erlaubt mehreren Entwicklern gleichzeitig an einem Projekt zu arbeiten.



Nachvollziehbarkeit

Jede Änderung wird protokolliert, sodass wir jederzeit wissen, wer was wann geändert hat.



Branching

Ermöglicht es verschiedene unabhängige Arbeitszweige zu führen.

Vorteile Von Git

01

Standard

Git ist der Industriestandard

03

Leistung & Flexibilität

Git is schnell und effizient

02

Dezentral

Jeder Entwickler hat eine lokale Kopie

04

Open Source

Git ist kostenlos und open-source

Aufgabe 1

Git installieren und einstellen





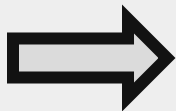
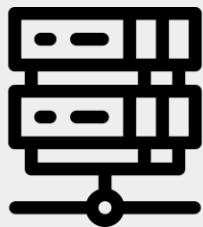
02

Repository

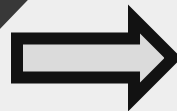
Was ist ein Repository?

Remote Repository

Wird auf einem Server gehosted.



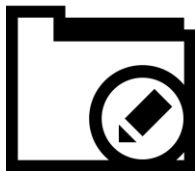
CLONE



Locale Repository

Befindet sich auf dem eigenen
Computer.

Aufbau



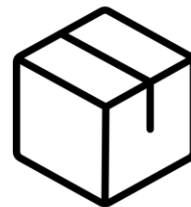
Arbeitsverzeichnis

Enthält die aktuellen
Projektdateien



Git Verzeichnis

Ein versteckter
Unterordner im
Arbeitsverzeichnis
mit dem Namen `.git`



Staging Area

Hier werden
Änderungen für den
nächsten Commit
vorbereitet

Aufgabe 2

Ein lokales Repository erstellen





03

Commits

Was sind Commit?

Bestandteile



Änderungen

Der Zustand der
getrackten Dateien



Commit Message

Beschreibung der
Änderung



Hash

SHA-1 Wert zur
Identifikation



Autor

Infos über den Autor
(Name, E-Mail, Zeitstempel)

Tracking & Staging



Untracked

Die Datei ist neu und Git kennt sie noch nicht



Staged

Die Datei wurde für den nächsten Commit vorgemerkt



Committed

Die Datei ist nun fester Bestandteil der Repository

Pull

Lädt alle neuen Commits aus dem Remote-Repository herunter und führt sie mit dem aktuellen Stand des lokalen Repositories zusammen.

Push

Lädt alle lokalen Commits, die noch nicht im Remote-Repository vorhanden sind, in das Remote-Repository hoch.

Aufgabe 3

Committen und Pushen einer
Readme



04

Branches

Neue Branches erstellen

Vorteile

01 **Parallelentwicklung**

Mehrere Entwickler arbeiten am selben Projekt

02 **Bessere Nachverfolgbarkeit**

Einfaches nachvollziehen welche Änderungen zu welchem Branch gehören

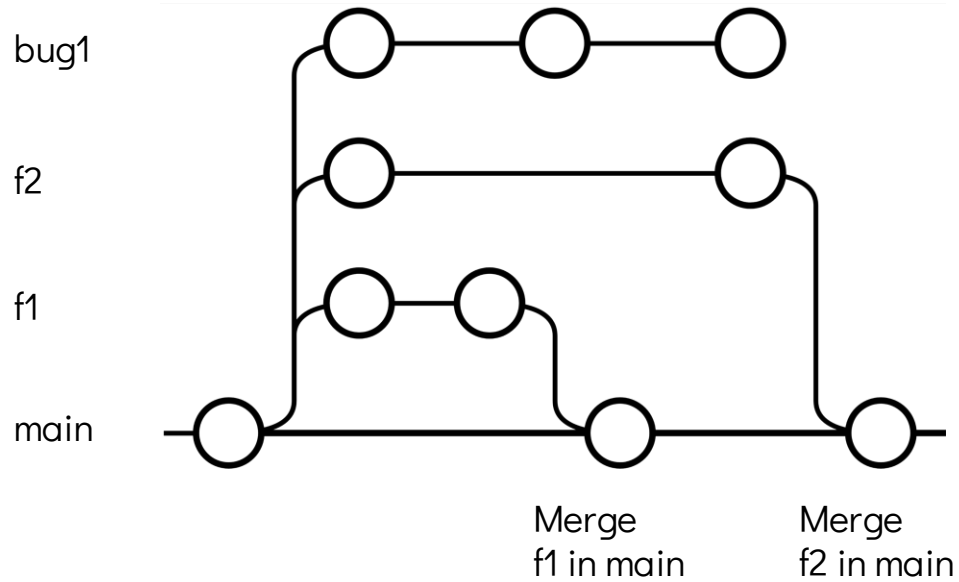
03 **Sauberer Hauptzweig**

Der main / master branch bleibt stabil

04 **Einfaches Zusammenführen**

Änderungen lassen sich gezielt zusammenführen

Branches



Aufgabe 4

Eine Branch erstellen



Danke

Habt ihr noch Fragen?

pr1.schergen@2clever4you.net

<https://moodle.hs-mannheim.de/mod/forum/view.php?id=106515>



CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon**, infographics & images by **Freepik**