

Git Cheat Sheet

Git herunterladen und installieren

- Linux: <https://git-scm.com/download/linux>
- macOS: <https://git-scm.com/download/mac>
- Windows: <https://git-scm.com/download/win>

Konfiguration

Username festlegen	git config --global user.name "Ihr Name"
E-Mail-Adresse angeben	git config --global user.email "beispieladresse@beispiel.de"

Repository erstellen

Ein neues lokales Repository erstellen und benennen	git init beispielname
Ein bestehendes Repository klonen	git clone http://beispielseite.de

Änderungen am Repository durchführen

Status des Verzeichnisses anzeigen	git status
Objekte anzeigen	git show
Eine Datei hinzufügen	git add beispieldatei
Alle Dateien eines Repositories hinzufügen	git add*
Alle neuen oder geänderten Dateien anzeigen	git diff
Aktuellen Stand einer Datei indizieren	git diff --staged
Änderungen nach dem Commit verfolgen	git diff HEAD
Unterschiede zwischen Index und aktuellem Stand anzeigen	git reset beispieldatei
Derzeit indizierte Dateien dauerhaft in die Versionshistorie aufnehmen	git commit -m "Erklärungen zu den Änderungen"

Branches

Branches erstellen und bearbeiten

Neuen Branch erstellen	git branch beispielname
Alle Branches auflisten	git branch --list
Branch löschen	git branch -d
Remote Branch entfernen	git push origin -delete
Branch umbenennen	git branch -m

Branch wechseln

Wechseln zu einem anderen Branch	git checkout anderer-branch
Neuen Branch anlegen und dorthin wechseln	git checkout -b

Branch Merge und Fetch

Historie eines Branches mit dem aktuellen Branch zusammenführen	git merge beispielname
Externes Repository registrieren und Historie tauschen	git fetch http://www.beispielseite.de
Alle Branches registrieren	git fetch -all
Lokales Repository registrieren	git fetch origin

Git Cheat Sheet – Seite 2

Branch Zwischenspeicher

Aktuelle Dateien übertragen, ohne den Branch zu committen	git stash
Änderungen speichern mit einer Erklärung	git stash save "Hier Erklärung einfügen"
Speicherstände auflisten	git stash
Änderungen am Speicherstand erneut durchführen	git stash apply
Änderungen am Speicherstand nachvollziehen	git stash show
Bereits durchgeführte Commits erneut anwenden	git stash pop
Zwischenstände verwerfen	git stash drop
Alle verfügbaren Speicherstände löschen	git stash clear
Speichern auf einem separaten Branch	git stash branch beispielbranch

Push und Pull

Externen Branch in den aktuellen lokalen Branch integrieren	git push http://www.beispielseite.de lokaler-beispielbranch
Daten auf den Remote Server übertragen	git push origin master
Push erzwingen	git push -f
Remote Branch über Push entfernen	git push origin -delete edited
Historie vom externen Repository ziehen	git pull
Daten vom Server ziehen	git pull origin master

Log anzeigen und bearbeiten

Alle Commits eines Branches anzeigen	git log
Anzahl der Commits limitieren (in diesem Beispiel auf 3)	git log -3
Commits eines bestimmten Autors oder einer bestimmten Autorin suchen	git log --author= "beispielname"
Commits für einen bestimmten Zeitraum anzeigen	git log <since>...<until>
Commits mit einer speziellen Datei anzeigen	git log --beispieldatei

Änderungen rückgängig machen

Bestehenden Commit verändern	git commit --amend
Datei aus der Staging-Area entfernen	git reset HEAD beispieldatei oder git restore --staged beispieldatei
Lokale Änderungen an einer Datei in der Staging-Area verwerfen	git checkout --beispieldatei oder git restore beispieldatei