Hier sind einige grundlegende Git-Befehle, die du im Terminal verwenden kannst, um mit Git-Repositories zu arbeiten:

**Einrichten**

1. **Git initialisieren:** Ein neues Git-Repository im aktuellen Verzeichnis erstellen.

sh

Code kopieren

git init

1. **Repository klonen:** Ein bestehendes Repository von GitHub klonen.

sh

Code kopieren

git clone https://github.com/username/repository.git

**Grundlegende Git-Operationen**

1. **Status anzeigen:** Den aktuellen Status des Arbeitsverzeichnisses anzeigen.

sh

Code kopieren

git status

1. **Änderungen hinzufügen:** Änderungen für den nächsten Commit vormerken.

sh

Code kopieren

git add <dateiname>

# Alle Änderungen hinzufügen

git add .

1. **Commit durchführen:** Vorgemerkte Änderungen in das Repository übernehmen.

sh

Code kopieren

git commit -m "Commit-Nachricht hier"

1. **Änderungen rückgängig machen:** Änderungen an einer Datei rückgängig machen (vor dem Staging).

sh

Code kopieren

git checkout -- <dateiname>

1. **Änderungen anzeigen:** Details zu den letzten Commits anzeigen.

sh

Code kopieren

git log

**Remote-Repositories**

1. **Remote hinzufügen:** Ein Remote-Repository hinzufügen.

sh

Code kopieren

git remote add origin https://github.com/username/repository.git

1. **Branches anzeigen:** Lokale und Remote-Branches anzeigen.

sh

Code kopieren

git branch

git branch -r

1. **Branch wechseln:** Zwischen Branches wechseln.

sh

Code kopieren

git checkout <branchname>

1. **Änderungen pushen:** Lokale Änderungen in das Remote-Repository hochladen.

sh

Code kopieren

git push origin <branchname>

1. **Änderungen pullen:** Änderungen vom Remote-Repository herunterladen und zusammenführen.

sh

Code kopieren

git pull origin <branchname>

**Erweiterte Operationen**

1. **Merge durchführen:** Änderungen eines Branches in den aktuellen Branch zusammenführen.

sh

Code kopieren

git merge <branchname>

1. **Rebase durchführen:** Historie eines Branches auf einen anderen anwenden.

sh

Code kopieren

git rebase <branchname>

1. **Tags erstellen:** Ein Tag an einem bestimmten Commit erstellen.

sh

Code kopieren

git tag -a v1.0 -m "Release v1.0"

1. **Tags pushen:** Tags in das Remote-Repository hochladen.

sh

Code kopieren

git push origin --tags

**Konfiguration**

1. **Git-Konfiguration anzeigen:** Aktuelle Git-Konfiguration anzeigen.

sh

Code kopieren

git config --list

1. **Globale Konfiguration ändern:** Git-Konfiguration global ändern (Benutzername, E-Mail usw.).

sh

Code kopieren

git config --global user.name "Dein Name"

git config --global user.email "deine.email@example.com"

**Sonstiges**

1. **Hilfe anzeigen:** Hilfe für Git-Befehle und Optionen anzeigen.

sh

Code kopieren

git --help

git <command> --help

Diese Befehle bieten eine solide Grundlage, um mit Git in der Terminal-Umgebung zu arbeiten. Je nach Projektanforderungen und Workflow können zusätzliche Git-Befehle erforderlich sein.