

Befehl

git help	Liste der Kommandos wird aufgeführt (schließen mit q)
git help <command>	Hilfe für <command> wird aufgeführt (schließen mit q)
git init	Erstellt lokales Git Repository in aktuellem Ordner
git status	Zeigt welche Dateien neu hinzugekommen sind und evtl. committed werden müssen
git add <file>	Datei <file> wird zu staging Bereich hinzugefügt
git add <list of files>	Fügt alle aufgeführten Dateien zu staging Bereich hinzu
git add - -all	Fügt alle Dateien in aktuellem Projekt zu staging Bereich hinzu
git add *.txt	Fügt alle Dateien vom Typ txt aus aktuellem Ordner zum staging Bereich hinzu
git add docs/	Fügt alle Dateien aus „docs“ Ordner zu staging Bereich hinzu
git add docs/*.txt	Fügt alle Dateien vom Typ txt aus docs Ordner zum staging Bereich hinzu
git add "**.txt"	Fügt alle Dateien vom Typ txt im ganzen Projekt zum staging Bereich hinzu
git commit -m '<message>'	Es werden alle Änderungen aus staging Bereich mit Nachricht <message> (sollte zusammenfassen was geändert wird) bekennt
git log	Führt alle commits der Vergangenheit mit ihrer Message auf
git diff - - staged	Änderungen in staging Bereich die noch nicht committed wurden
git reset <file>	Entfernt <file> aus staging Bereich (geht nur vor commit)
git rm '<file>'	Entfernt <file> von Festplatte und fügt diese Änderung zum staging Bereich hinzu
git config - -global user.name "name"	Name der bei Änderungen angezeigt wird, wird definiert
git config - -global user.email <mail>	Email-Adresse die bei Änderungen angezeigt wird, wird definiert
git config - -global color.ui true	Farben für Kommandozeile werden aktiviert (eher unwichtig)
git checkout - -<target>	Entfernt alle Änderungen seitdem <target> das letzte mal committed wurde

Befehl

<code>git remote add <name> <https://.></code>	Hinterlegt online Repository zum hochladen von lokalen Dateien und setzt den Namen dieses Repository lokal auf <name> fest
<code>git push -u <name> <branch></code>	'Pushed' lokale Änderungen auf hinterlegtes online repository. <name> ist der festgelegte Name des Repository und <branch> ist der Ast, der hochgeladen werden soll
<code>git pull <name> <branch></code>	Überträgt Änderungen die von Anderen an online repository <name> und dessen Ast <branch> vorgenommen wurden auf das lokale Verzeichnis
<code>git diff HEAD</code>	Zeigt an was sich seit letztem eigenen commit geändert hat
<code>git branch <name></code>	Erstellt neuen Ast mit Name <name> und überträgt alle Dateien aus Master Ast in diesen Ast
<code>git checkout <branch></code>	Wechselt zu <branch> Ast
<code>git merge <branch></code>	Vereint Änderungen von aktuellem Ast mit Ast <branch>
<code>git branch -d <branch></code>	Löscht Ast <branch>