

Kommandozeileninterpreter:

```
C:\Users\Maria Aschenberger>git help
usage: git [--version] [--help] [-C <path>] [-c <name>=<value>]
       [--exec-path<path>] [--html-path] [--man-path] [--info-path]
       [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
       [--git-dir<path>] [--work-tree<path>] [--namespace=<name>]
       [--super-prefix<path>] [--config-env=<name>=<envvar>]
       <command> [<args>]

These are common Git commands used in various situations:

start a working area (see also: git help tutorial)
  clone      Clone a repository into a new directory
  init       Create an empty Git repository or reinitialize an existing one

work on the current change (see also: git help everyday)
  add        Add file contents to the index
  mv         Move or rename a file, a directory, or a symlink
  restore    Restore working tree files
  rm         Remove files from the working tree and from the index
  sparse-checkout Initialize and modify the sparse-checkout

examine the history and state (see also: git help revisions)
  bisect     Use binary search to find the commit that introduced a bug
  diff       Show changes between commits, commit and working tree, etc
  grep       Print lines matching a pattern
  log        Show commit logs
  show       Show various types of objects
  status     Show the working tree status

grow, mark and tweak your common history
  branch     List, create, or delete branches
  commit     Record changes to the repository
  merge      Join two or more development histories together
  rebase     Reapply commits on top of another base tip
  reset      Reset current HEAD to the specified state
  switch     Switch branches
  tag        Create, list, delete or verify a tag object signed with GPG

collaborate (see also: git help workflows)
  fetch      Download objects and refs from another repository
  pull       Fetch from and integrate with another repository or a local branch
  push       Update remote refs along with associated objects

'git help -a' and 'git help -g' list available subcommands and some
concept guides. See 'git help <command>' or 'git help <concept>'
to read about a specific subcommand or concept.
See 'git help git' for an overview of the system.
```

```
C:\Users\Maria Aschenberger>cd..
C:\Users>cd..
C:\>dir
Datenträger in Laufwerk C: ist OS
Volumeserienummer: DA89-44A3

Verzeichnis von C:\

11.06.2020 09:19 <DIR>      Apps
14.01.2021 23:19 <DIR>      Dell
18.09.2020 07:53 <DIR>      Downloads
11.06.2020 09:16 <DIR>      Drivers
22.11.2021 21:18 <DIR>      Intel
05.11.2021 21:49 <DIR>      Landspinnerei
07.12.2019 10:14 <DIR>      Penflogs
16.11.2021 08:20 <DIR>      Program Files
14.10.2021 17:47 <DIR>      Program Files (x86)
07.11.2021 14:15 <DIR>      Prvt
22.05.2021 15:52 <DIR>      Riot Games
28.10.2021 15:11 <DIR>      Schule
06.04.2021 15:15 <DIR>      Users
13.11.2021 16:59 <DIR>      Windows
                0 Datei(en),          0 Bytes
                14 Verzeichnis(se), 281 632 874 496 Bytes frei

C:\>cd Schule
C:\Schule>cd 2AHITM
C:\Schule\2AHITM>cd CPR
C:\Schule\2AHITM\CPR>cd SW
C:\Schule\2AHITM\CPR\SW>cd\
C:\>cd\
```

```
C:\Users\Maria Aschenberger>help löscht den Bildschirminhalt
Zeigt Hilfeinformationen zu Windows-Befehlen an.

HELP [Befehl]

    Befehl - Zeigt Hilfeinformationen zu diesem Befehl an.

C:\Users\Maria Aschenberger>help cd
Wechselt das Verzeichnis oder zeigt dessen Namen an.

CHDIR [/D] [Laufwerk:][Pfad]
CHDIR [...]
CD [/D] [Laufwerk:][Pfad]
CD [...]

    ... Gibt an, dass Sie in das übergeordnete Verzeichnis wechseln möchten.

Geben Sie "CD Laufwerk:" ein, um das aktuelle Verzeichnis auf dem angegebenen
Laufwerk anzuzeigen. Mit CD ohne Parameter wird das aktuelle Laufwerk und
Verzeichnis angezeigt.

Verwenden Sie die /D-Option, um zusätzlich zum Wechseln des Verzeichnisses
auch das aktuelle Laufwerk zu wechseln.

Wenn die Befehlsweiterungen aktiviert sind, wird CHDIR folgendermaßen
verändert:

Der angegebene Verzeichnisname wird so konvertiert, dass dieser bezüglich
Groß- und Kleinschreibung dem Namen auf dem Laufwerk entspricht. So wird durch
CD C:\TEMP der aktuelle Pfad auf das Verzeichnis C:\Temp festgelegt, wenn ein
Verzeichnis mit diesem Namen auf dem Laufwerk existiert.

Der CHDIR-Befehl behandelt Leerzeichen nicht als Trennzeichen, sodass es
möglich ist, in ein Unterverzeichnis zu wechseln, dessen Name Leerzeichen
enthält, ohne den Namen in Anführungszeichen einzuschließen. Beispiel:

    cd \\WINNT\Profile\Benutzername\Programme\Startmenü

ist dasselbe wie:

    cd "\\WINNT\Profile\Benutzername\Programme\Startmenü"
```

Mit der auf/ab Taste kann man die zuletzt verwendeten Befehle gleich hinein kopieren ohne sie selbst zu schreiben!

Lokales Repo erzeugen:

```
C:\Schule\2AHITM\CPR\SW>mkdir CPRSW
C:\Schule\2AHITM\CPR\SW>cd CPRSW
C:\Schule\2AHITM\CPR\SW\CPRSW>mkdir firstRepo
C:\Schule\2AHITM\CPR\SW\CPRSW>cd firstRepo
C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git init
Initialized empty Git repository in C:/Schule/2AHITM/CPR/SW/CPRSW/firstRepo/.git/

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>dir /adh
Datenträger in Laufwerk C: ist OS
Volumeserienummer: DA89-44A3

Verzeichnis von C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo

23.11.2021 09:37 <DIR>      .git
                0 Datei(en),          0 Bytes
                1 Verzeichnis(se), 281 638 223 872 Bytes frei

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>
```

Status prüfen:

```
C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>
```

Inhalt adden:

```
C:\Schule\2AHITM\CPR\SW\CPRSW>cd firstRepo

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>copy con firstFile.txt
Hello World,
my name is Maria^Z
    1 Datei(en) kopiert.

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    firstFile.txt

nothing added to commit but untracked files present (use "git add" to track)

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git add firstFile.txt

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   firstFile.txt

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>
```

Inhalt committen:

```
C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git commit -m "First file added"
[master (root-commit) 28cdd6a] First file added
 1 file changed, 2 insertions(+)
 create mode 100644 firstFile.txt

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git status
On branch master
nothing to commit, working tree clean

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>
```

Inhalt verändern:

```
C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>copy con firstFile.txt
...
firstFile.txt überschreiben? (Ja/Nein/Alle): Ja
Hallo Welt.
^Z

    1 Datei(en) kopiert.

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   firstFile.txt

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   firstFile.txt

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git commit -a -m "Overwrite first Line"
[master bc609f3] Overwrite first Line
 1 file changed, 2 insertions(+), 2 deletions(-)

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>
```

Dateien löschen (vollständig):

```
C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>copy con delFile.txt
Hallo Welt 1.
^Z

    1 Datei(en) kopiert.

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>copy con delFile2.txt
Hallo Welt 2.
^Z

    1 Datei(en) kopiert.

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        delFile.txt
        delFile2.txt

nothing added to commit but untracked files present (use "git add" to track)

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git add .

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git rm delFile.txt
error: the following file has changes staged in the index:
    delFile.txt
(use --cached to keep the file, or -f to force removal)

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git rm -f delFile.txt
rm 'delFile.txt'

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   delFile2.txt

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git commit -m "remove Files"
[master 783ca0b] remove Files
 1 file changed, 1 insertion(+)
 create mode 100644 delFile2.txt

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git rm -f delFile2.txt
rm 'delFile2.txt'
```

```
C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        deleted:    delFile2.txt

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git commit -m "deletion"
[master 3bcb705] deletion
 1 file changed, 1 deletion(-)
 delete mode 100644 delFile2.txt

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>
```

Dateien löschen (nur Versionsverwaltung):

```
C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>copy con doNotTrack.txt
DO NOT TRACK!
^Z
    1 Datei(en) kopiert.

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>copy con doNotTrack2.txt
do not track!
^Z
    1 Datei(en) kopiert.

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    doNotTrack.txt
    doNotTrack2.txt

nothing added to commit but untracked files present (use "git add" to track)

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git add .

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git rm --cached doNotTrack.txt
rm 'doNotTrack.txt'

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   doNotTrack2.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    doNotTrack.txt

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git commit -m "untrack doNotTrack.txt"
[master b6f202c] untrack doNotTrack.txt
1 file changed, 1 insertion(+)
create mode 100644 doNotTrack2.txt

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git rm --cached doNotTrack2.txt
rm 'doNotTrack2.txt'

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    deleted:    doNotTrack2.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    doNotTrack.txt
    doNotTrack2.txt

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git commit -m "untrack"
[master e9c9785] untrack
1 file changed, 1 deletion(-)
delete mode 100644 doNotTrack2.txt

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>
```

Dateien umbenennen:

```
C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git add doNotTrack.txt

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git commit -m "add and commit doNotTrack.txt"
[master 9d46943] add and commit doNotTrack.txt
 1 file changed, 1 insertion(+)
 create mode 100644 doNotTrack.txt

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git mv doNotTrack.txt doTrack.txt

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        renamed:    doNotTrack.txt -> doTrack.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        doNotTrack2.txt

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git commit -m "renamed files"
[master fed98e7] renamed files
 1 file changed, 0 insertions(+), 0 deletions(-)
 rename doNotTrack.txt => doTrack.txt (100%)

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>
```

Änderungen einer Datei restaurieren:

```
C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git status
On branch master
nothing to commit, working tree clean

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>copy con firstFile.txt
restore after this
firstFile.txt überschreiben? (Ja/Nein/Alle): ja
^Z
        1 Datei(en) kopiert.

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   firstFile.txt

no changes added to commit (use "git add" and/or "git commit -a")

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git restore firstFile.txt

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>git status
On branch master
nothing to commit, working tree clean

C:\Schule\2AHITM\CPR\SW\CPRSW\firstRepo>
```

Bestehendes GitHub Repo clonen:

```
C:\Schule\2AHITM\CPR\SW\CPRSW>git clone https://github.com/MartinaReisHTL/CPRSW_StudentsShare.git
Cloning into 'CPRSW_StudentsShare'...
remote: Enumerating objects: 16, done.
remote: Counting objects: 100% (16/16), done.
remote: Compressing objects: 100% (13/13), done.
remote: Total 16 (delta 4), reused 9 (delta 1), pack-reused 0R
Receiving objects: 100% (16/16), 1.14 MiB | 1.77 MiB/s, done.
Resolving deltas: 100% (4/4), done.

C:\Schule\2AHITM\CPR\SW\CPRSW>cd CPRSW_StudentsShare

C:\Schule\2AHITM\CPR\SW\CPRSW\CPRSW_StudentsShare>dir /adh
Datenträger in Laufwerk C: ist OS
Volumeseriennummer: DA89-44A3

Verzeichnis von C:\Schule\2AHITM\CPR\SW\CPRSW\CPRSW_StudentsShare

25.11.2021  12:02    <DIR>          .git
             0 Datei(en),              0 Bytes
             1 Verzeichnis(se), 271 332 376 576 Bytes frei

C:\Schule\2AHITM\CPR\SW\CPRSW\CPRSW_StudentsShare>
```