



git

A thick blue L-shaped bar is positioned on the left side of the slide, extending from the top to the bottom. A horizontal light blue bar is attached to the right side of the vertical part of the L-shape, spanning across the middle of the slide.

Konfiguration

Links

- Erster Eindruck: <https://lerneprogrammieren.de/git/>
- Tutorial: <https://docs.github.com/de/get-started/using-git/about-git>
- Cheatsheet Ionis: <https://www.ionos.de/digitalguide/websites/web-entwicklung/git-cheat-sheet/>
- ...

Installation

- Download, z.B. git for windows (shell)
- GitHubDesktop (GUI)
- Browser: www.github.com

Konfiguration Schule

Eingeben über Kommandozeile:

```
git config --global http.proxy 10.23.2.73:@proxy.its-stuttgart.de:3128
```

 

(Ihre IP erfahren Sie zum Beispiel über ipconfig auf der Kommandozeile)

Proxy ausschalten (evt. Zu Hause):

```
D:>git config --global --unset https.proxy  
D:>git config --global --unset http.proxy
```

Konfiguration Benutzer:

```
D:>git config --global user.name "Daniel"  
D:>git config --global user.email daniel@meine-email.de
```

A decorative blue L-shaped bar is positioned on the left side of the slide, extending horizontally across the middle. The word "Beispiel" is written in a bold, italicized blue font within the horizontal part of this bar.

Beispiel

Beispiel-Session

```
// Lokale Kopie des Repo ziehen
mkdir myHelloGit
cd myHelloGit
git clone https://github.com/bl-its/HelloGit1

// Aktuelle Version laden
git pull

//Program.cs neu geschrieben und jetzt
hinzufügen
git add Programm.cs git commit -m "Program.cs
hinzugefügt."
git push

// Program.cs geändert
git commit Programm.cs -m "Kommentar gelöscht"
git push
```

A decorative blue L-shaped bar is positioned on the left side of the slide, extending horizontally across the middle. The word "Befehle" is centered within this bar.

Befehle

Befehle 1/3

Auschecken [\[edit\]](#)

```
D:\Dropbox\tmp\git1>git clone https://github.com/bl-its/HelloGit1
```

Einzelnen Branch [\[edit\]](#)

```
git clone -b V1.1 https://github.com/bl-its/HelloGit1
```

```
git clone user@git-server:project_name.git -b <branch_name> --single-branch /your/folder
```

Informationen [\[edit\]](#)

```
git --help
```

```
git status
```

```
git show
```

```
git diff
```

```
git log
```

Befehle 2/3

Dateien hinzufügen / ändern [\[edit\]](#)

<code>add</code>	Add file contents to the index //staging area
<code>mv</code>	Move or rename a file, a directory, or a symlink
<code>restore</code>	Restore working tree files
<code>rm</code>	Remove files from the working tree and from the index

Änderungen festlegen [\[edit\]](#)

<code>commit</code>	Record changes to the repository
<code>merge</code>	Join two or more development histories together

Änderungen übertragen [\[edit\]](#)

<code>pull</code>	Fetch from and integrate with another repository or a local branch
<code>push</code>	Update remote refs along with associated objects

Befehle 3/3

Lokale Änderungen rückgängig machen [\[edit\]](#)

```
git reset --hard HEAD          //setzt gesamte lokalen Repo auf letzten Commit zurück
git reset HEAD <filename>      //setzt <filename> auf letzten Commit zurück

git log                        // zeigt die letzten commits mit <commitId>
git show <commitId>           // zeigt infos zu einem commit an
git checkout <commitId> <filename> // setzt <filename> auf commit <commitId>
```

```
git revert HEAD                // letzten commit rückgängig machen
git revert <CommitId>          // commit <CommitId> rückgängig machen
```

Branches [\[edit\]](#)

```
git branch                    // branch anzeigen
git branch <branchName>      // branch erstellen
git branch -d <branchName>    // branch löschen
git checkout <branchName>     // branch wechseln
git merge <branchName>        // branch in aktuelle branch mergen
```