

Git Mini Cheat Sheet

Einstellungen

- Konfiguration:

```
git config --global --add user.name "John Doe"
```

- Einige wichtige Einstellungen:

Setting	Meaning
user.name	User-Name.
user.email	User-E-Mail.
color.ui	Farben? ("auto!")?
core.editor	Welcher Editor?

Basic Git

- Repository erzeugen:

```
cd projDir  
git init
```

- Dateien neu zum Repository hinzufügen bzw. geänderte Dateien für den nächsten Commit 'stagen':

```
git add file1 file2
```

- Status und log:

```
git status  
git log
```

- Änderungen commiten:

```
git commit  
git commit -a
```

Erste Zeile: alle Dateien, die mit git add zur staging area hinzugefügt wurden; zweite Zeile: alle geänderten Dateien.

- fileName auf letzte comittete Version zurücksetzen:

```
git checkout -- fileName
```

- Hilfe:

```
git stash --help
```

zeigt die manpage für git stash.

Branches

- Branches auflisten:

```
git branch
```

-r hinzufügen für remote branches, -a für remote und lokale Branches.

- Neuen Branch erzeugen:

```
git branch newBranch
```

Branch erzeugen und gleich in den neuen Branch wechseln:

```
git checkout -b newBranch
```

- In einen Branch wechseln:

```
git checkout branchName
```

- Branch löschen:

```
git branch -d branchName
```

- other Branch in den aktuellen Branch 'mergen':

```
git merge other
```

Mit remotes arbeiten

Bare repositories

- 'bare repository' (keine Arbeitskopie) anlegen:

```
git init --bare
```

Remotes

- Hinzufügen:

```
git remote add name user@host:/path/bareRepo  
git remote add name user@host:/path/repo/.git
```

- In einen remote 'pushen':

```
git push remoteName branchToPush
```

Nach einem erfolgreichen `git push -u origin master` "folgt" der aktuelle Branch automatisch `origin/master` (push und pull!).

- Aus einem remote 'pullen':

```
git pull remote
```

- Einen remote klonen, dabei neues Repository `repoName` anlegen:

```
git clone urlToRemote repoName
```

Graphische Tools

- GUI für Alltägliches bzw. GUI für Geschichte/Branches:

```
git gui  
gitk
```

- merge und diff tools:

```
git difftool [files]  
git mergetool [files]
```

Vorher die Einstellungen `merge.tool` und `diff.tool` setzen (meld!)