Git Cheat Sheet

Create

From existing data

```
cd ~/my_project_dir
$ git init
$ git add .
```

From existing repo

```
$ git clone ~/existing/repo ~/new/repo
$ git clone you@host.de:dir/project.git
$ git clone http://[USER@]host.de/project.git
```

Browse

Files changed in working directory

```
$ git status
```

Changes of tracked files

```
$ git diff
```

Changes between ID1 and ID2

```
$ git diff <ID1> <ID2>
```

History of changes

```
$ git log
```

Commit

Commit all local changes

```
$ git commit -a
```

Commit all local changes and add message

```
$ git commit -am „<message>“
```

Useful tips

Documentation/help

```
$ git help [command]
$ man git-[command]
```

Push branch to remote

```
$ git push <origin> <branch>
```

Delete branch (locally and remote)

```
$ git branch -d <branch>
$ git push <origin> :<branch>
```

Change not pushed commits untill ID

```
$ git rebase -i <ID>
```

Revert

Revert the last commit

```
$ git revert HEAD
```

Return to last committed state

```
$ git checkout -f | git reset –hard
```

Revert specific commit

```
$ git revert <ID>
```

fix/change last commit

```
$ git commit ––amend
```

branch

List all branches (locally and remote)

```
$ git branch -a
```

Switch to BRANCH

```
$ git checkout BRANCH
```

Merge Branch B1 into B2

```
$ git checkout <B2>
$ git merge <B1>
$ git merge –no-ff -m „Message“
```

generates commit even as –ff option

```
$ git merge –squash
```

merge without generated commit

Create Branch based on HEAD

```
$ git branch <branch>
```

Create Branch based on HEAD and checkout

```
$ git checkout -b <branch>
```

Create Branch based on another

```
$ git branch <new> <branch>
```

Delete a branch

```
$ git branch -d <branch>
```

Update

Fetch latest changes from origin

```
$ git fetch
```

see if something changed

```
$ git remote update
```

Resolve merge conflicts

View merge conflicts

```
$ git diff
```

View merge conflicts against base file

```
$ git diff –base <FILE>
```

View merge conflicts against other changes

```
$ git diff ––theirs <FILE>
```

View merge conflicts against your changes

```
$ git diff ––ours <FILE>
```

After resolving conflicts, merge with

```
$ git add <CONFLICTING_FILE>
$ git rebase ––continue
```

Publish

Push changes to origin

```
$ git push [origin] [branch]
```

Make a Version

```
$ git tag <version_name>
```

Prepare a patch

```
$ git format-patch origin
```

Configuration

```
$ git config [––global] [option]
```

global is stored in ~/.gitconfig

user

```
user.name NAME
user.email EMAIL
```

color

```
color.ui auto
```

optimisation

```
packs.threads 0
diff.renamelimit 0
```

Date: October 12, 2012 - Version 1.3
Author: Max Oberberger (max@oberbergers.de)
https://github.com/chiemseesurfer/latex-gitCheatSheet