Git & Github Tutorial

Daniel Schädler & Jean-Pierre Hotz

DHBW Karlsruhe

22. October 2018

General

Terminology

Git areas

Use your Github user in Git

Ignoring files

Local repositories

Staging and committing

Display changes and commits

Remote repositories

Remote repositories

General

General

- Git is a **V**ersion **C**ontrol **S**ystem

General

- Git is a **V**ersion **C**ontrol **S**ystem
- created in 2005 by Linus Torvalds

Terminology

Terminology

- Commit (History)

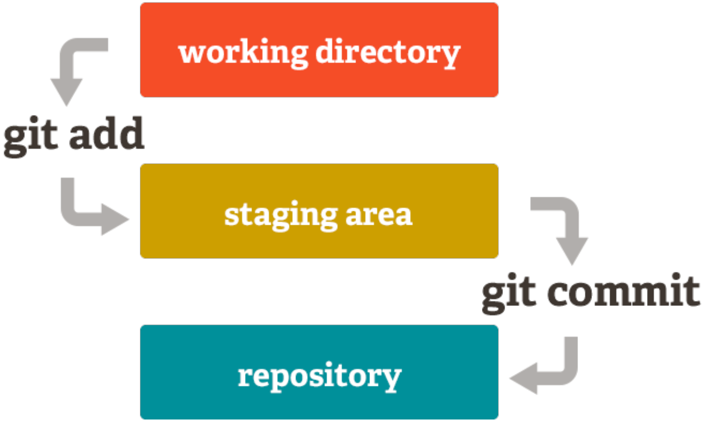
Terminology

- Commit (History)
- Branch (Merge)

Terminology

- Commit (History)
- Branch (Merge)
- (Remote) Repository

Git areas


working directory
git add
staging area
git commit
repository

Use your Github user in Git

Use your Github user in Git

- set your username and e-mail

  ```
  $ git config --global user.name "<username>"
  $ git config --global user.email <e-mail>
  ```

Ignoring files

Ignoring files

- exclude files with `.gitignore`-file

Ignoring files

- exclude files with `.gitignore`-file
- contains patterns

Ignoring files

- exclude files with `.gitignore`-file
- contains patterns
- * matches anything

Ignoring files

- exclude files with `.gitignore`-file
- contains patterns
- $*$ matches anything
- postceding / matches a folder

Ignoring files

- exclude files with `.gitignore`-file
- contains patterns
- $*$ matches anything
- postceding / matches a folder
- preceding / matches only in the root folder

Ignoring files

- exclude files with `.gitignore`-file
- contains patterns
- $*$ matches anything
- postceding / matches a folder
- preceding / matches only in the root folder
- preceding `!` negates the pattern (i.e. includes files)

What do the following rules exclude?

```
/src/
!/src/working/
```

Local repositories

Local repositories

- create new local repository

```
$ git init
```

Local repositories

- create new local repository

```
$ git init
```

- clone a remote repository

```
$ git clone <URL>
```

Staging and committing

Staging and committing

- stage changes

  ```
  $ git add (--all / <files>)
  ```

Staging and committing

- stage changes

  ```
  $ git add (--all / <files>)
  ```

- commit all the staged changes

  ```
  $ git commit -m "<Message>"
  ```

Staging and committing

- stage changes

  ```
  $ git add (--all / <files>)
  ```

- commit all the staged changes

  ```
  $ git commit -m "<Message>"
  ```

- Commit messages are important!

Display changes and commits

Display changes and commits

- see changes in code

```
$ git diff <commit1> <commit2> <files>
```

Display changes and commits

- see changes in code

```
$ git diff <commit1> <commit2> <files>
```

- see staged changes

```
$ git status
```

Display changes and commits

- see changes in code

  ```
  $ git diff <commit1> <commit2> <files>
  ```

- see staged changes

  ```
  $ git status
  ```

- see commit history

  ```
  $ git log
  ```

Remote repositories

Remote repositories

- add remote repository

```
$ git remote add <Name> <URL>
```

Remote repositories

- add remote repository

```
$ git remote add <Name> <URL>
```

- not needed when repository was cloned

Remote repositories

- add remote repository

```
$ git remote add <Name> <URL>
```

- not needed when repository was cloned
- set remote repository as up-stream

```
$ git branch -u <Name> <Branch>
$ git push -u <Name> <Branch>
```

Remote repositories

- add remote repository

  ```
  $ git remote add <Name> <URL>
  ```

- not needed when repository was cloned
- set remote repository as up-stream

  ```
  $ git branch -u <Name> <Branch>
  $ git push -u <Name> <Branch>
  ```

- push all commits to remote repository

  ```
  $ git push
  ```

Remote repositories

- add remote repository

```
$ git remote add <Name> <URL>
```

- not needed when repository was cloned
- set remote repository as up-stream

```
$ git branch -u <Name> <Branch>
$ git push -u <Name> <Branch>
```

- push all commits to remote repository

```
$ git push
```

- pulls all commits from remote repository

```
$ git pull
```

Remote repositories

Remote repositories

- list tags

```
$ git tag --list
```

Remote repositories

- list tags

```
$ git tag --list
```

- tag a commit

```
$ git tag -a <version> -m "<message>"<commit>
```

Remote repositories

- list tags

```
$ git tag --list
```

- tag a commit

```
$ git tag -a <version> -m "<message>"<commit>
```

- push all tags

```
$ git push origin --tags
```