



GIT, MARKDOWN, DOCKER



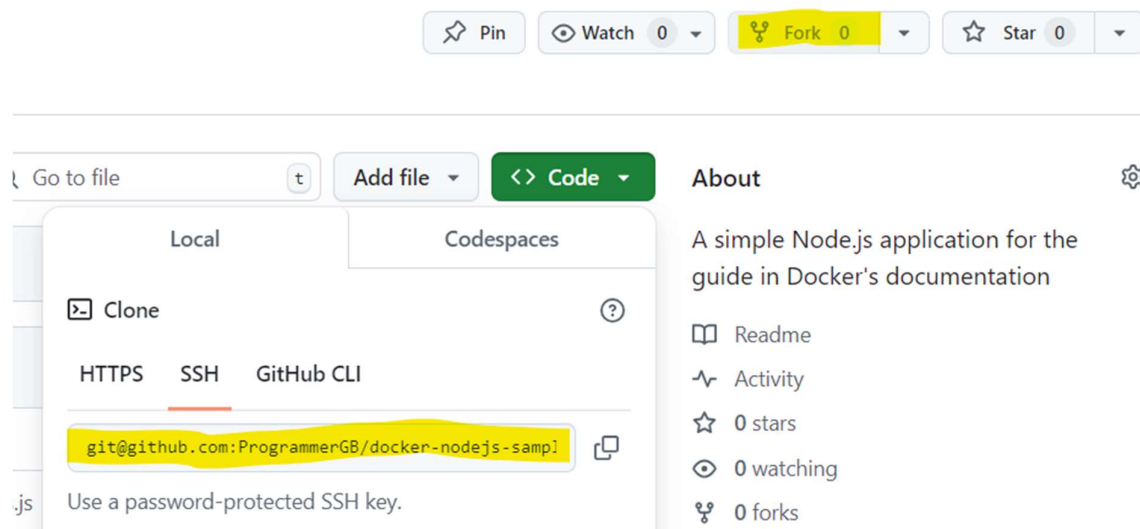
Giacomo Betsch

Inhalt

Klonen Repositories	3
Einrichtung der Entwicklungsumgebung.....	4
Erstellung der README.md	5
Verwendung von Git (Commit, Push).....	6
Erstellung und Nutzung von Docker-Containern	7

Klonen Repositories

Als erstes muss man eine Fork auf meinen Github Akkount erstellt.



Danach muss man in den gewünschten Ordner navigieren. Das macht man mit dem Befehl `cd` und dem Ordernamen (siehe unten). Um das Repositories zu Klonen braucht man den SSH key den man in Git hub unter «Code» bekommt und dort kopieren kann.

```
giaco@G-Lehre-1 MINGW64 ~
$ cd ZLI

giaco@G-Lehre-1 MINGW64 ~/ZLI
$ cd Abschlussprojekt

giaco@G-Lehre-1 MINGW64 ~/ZLI/Abschlussprojekt
$ git clone git@github.com:ProgrammerGB/docker-nodejs-sample.git
Cloning into 'docker-nodejs-sample'...
remote: Enumerating objects: 64, done.
remote: Counting objects: 100% (37/37), done.
remote: Compressing objects: 100% (33/33), done.
remote: Total 64 (delta 6), reused 4 (delta 4), pack-reused 27 (from 1)
Receiving objects: 100% (64/64), 1.71 MiB | 3.20 MiB/s, done.
Resolving deltas: 100% (7/7), done.
```

Einrichtung der Entwicklungsumgebung

Ich musste Docker aus dem Internet herunterladen damit ich dort einen Container erstellen kann. Ausserdem habe ich neue Ablageordner erstellt damit ich meine Codes dort abspeichern kann.

Erstellung der README.md

Wir hatten schon aus der Vorlage ein README.md Datei. Den Inhalt mussten wir löschen. Wir haben als erstes Titel mit ## vor dem Titel geschrieben damit der Text zum Titel wird (der Titel wird grösser und dicker). Ausserdem habe ich Bilder hinzugefügt mit ![alternativtext](images/image.png) . Ich haben die Themen

Klonen des Repositories

Installation der notwendigen Pakete

Docker-Konfiguration und -Installation

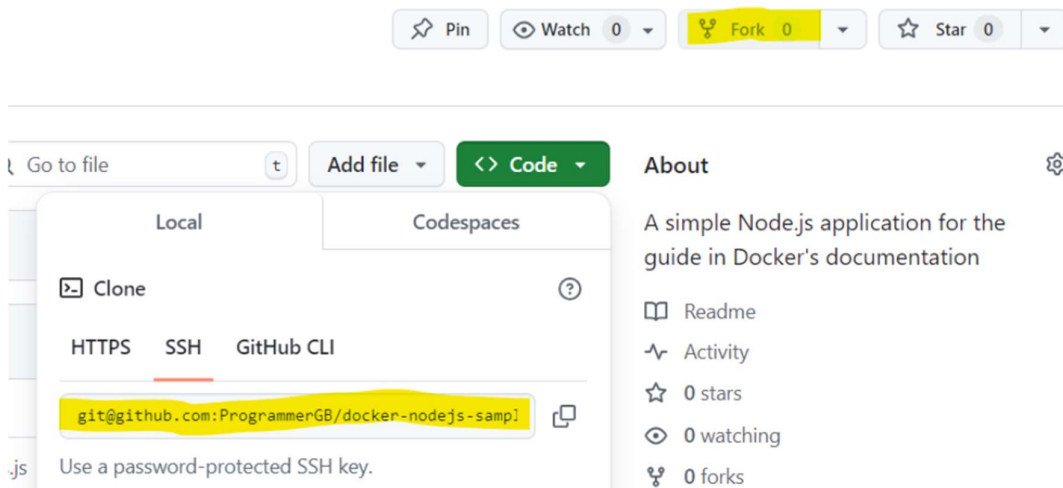
Starten der Applikation in einem Docker-Container

Zusammengefasst und die Anforderungen aufgeschrieben.



Klonen des Repositories

Als erstes muss man eine Fork auf meinen Github Akkount erstellt.



Danach muss man in den gewünschten Ordner navigieren. Das macht man mit dem Befehl `cd` und dem Ordernamen (siehe unten). Um das Repositories zu Klonen brauch man den SSH key den man in Git hub unter «Code» bekommt und dort kopieren kann.

```
giaco@G-Lehre-1 MINGW64 ~  
$ cd ZLI  
  
giaco@G-Lehre-1 MINGW64 ~/ZLI  
$ cd Abschlussprojekt  
  
giaco@G-Lehre-1 MINGW64 ~/ZLI/Abschlussprojekt  
$ git clone git@github.com:ProgrammerGB/docker-nodejs-sample.git
```

2 min read



Verwendung von Git (Commit, Push)

Ich musste ein Git code ins Terminal vom VS Code (geht auch über das normale Terminal) schreiben. Als erstes musste ich «git add .» schreiben damit er die Dateien innerhalb auswählt. Danach musste ich «git commit -m "README.md hochladen"» eingeben damit diese Files abgespeichert werden. Am Schluss musste ich noch «git push» schreiben damit er die Files auch auf Github hochlädt.

```
PS C:\Users\giaco\ZLI\Abschlussprojekt\docker-nodejs-sample> git commit -m "README Neu"
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
PS C:\Users\giaco\ZLI\Abschlussprojekt\docker-nodejs-sample> git push
Enumerating objects: 12, done.
Counting objects: 100% (12/12), done.
Delta compression using up to 20 threads
Compressing objects: 100% (10/10), done.
Writing objects: 100% (10/10), 271.14 KiB | 1.64 MiB/s, done.
Total 10 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:ProgrammerGB/docker-nodejs-sample.git
   2a77b9d..8315bc5  main -> main
PS C:\Users\giaco\ZLI\Abschlussprojekt\docker-nodejs-sample> |
```

Erstellung und Nutzung von Docker-Containern

Für die Erstellung von Docker Containern musste ich zuerst das Terminal öffnen. Ich musste mich zuerst zu den Ordnern mit `cd` navigieren. Danach musste ich «docker init» eingegeben. Das Terminal gab mir ein paar Fragen (siehe unten) für die Konfiguration des Containers.

```
PS C:\Users\giaco\ZLI\Abschlussprojekt\docker-nodejs-sample> docker init

Welcome to the Docker Init CLI!

This utility will walk you through creating the following files with sensible defaults for your project:
- .dockerignore
- Dockerfile
- compose.yaml
- README.Docker.md

Let's get started!

? What application platform does your project use? Node
? What version of Node do you want to use? 18.0.0

? What version of Node do you want to use? 18.0.0
? Which package manager do you want to use? npm
? What command do you want to use to start the app? [tab for suggestions] (node index.js) node src/index.js
? What command do you want to use to start the app? node src/index.js
? What port does your server listen on? 3000

? What port does your server listen on? 3000

✓ Created → .dockerignore
✓ Created → Dockerfile
✓ Created → compose.yaml
✓ Created → README.Docker.md

→ Your Docker files are ready!
```

