

Abschlussauftrag Development Fundamentals



Mario Schwyzer

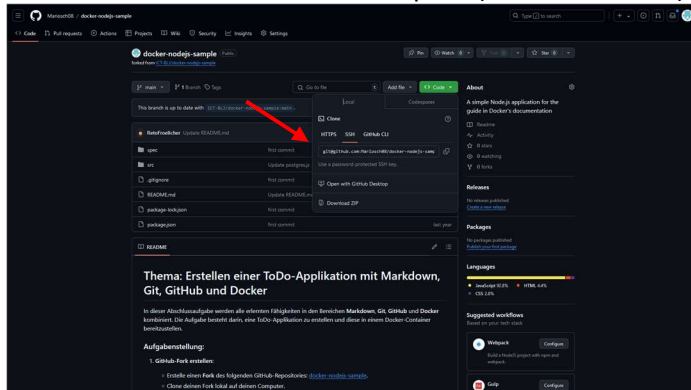
30.10.2024

Inhaltsverzeichnis

Clonen von Fork auf Lokalen Computer	3
Readme.md erstellen	4
Einrichtung der Entwicklungsumgebung.....	4
Verwendung von Git (Commit, Push).....	4
Erstellung und Nutzung von Docker-Containern.....	5

Clonen von Fork auf Lokalen Computer

1. Als erstes wird die Git Bash auf dem Computer geöffnet
2. Als nächstes wird das Repository auf Github geöffnet
3. Dann wird der SSH URL kopiert (siehe Bild unten)



4. Dann wird auf Git Bash in ein directory gegangen in dem man das Repos haben möchte
5. Danach wird der Befehl mit deinem SSH URL gemacht: `$ git clone https://github.com/YOUR-USERNAME/YOUR-REPOSITORY`

```
MINGW64:/c:/Users/MarioSchwyzer/Documents/020_zli/110_development_fundamentals/020_abschlussaufgabe

MarioSchwyzer@CHHHMPC0959 MINGW64 ~
$ cd Documents/020_zli/110_development_fundamentals/020_abschlussaufgabe/







MarioSchwyzer@CHHHMPC0959 MINGW64 ~/Documents/020_zli/110_development_fundamentals/020_abschlussaufgabe
$ git clone git@github.com:Mariosch08/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 | 3.26 MiB/s, done.
Resolving deltas: 100% (7/7), done.

MarioSchwyzer@CHHHMPC0959 MINGW64 ~/Documents/020_zli/110_development_fundamentals/020_abschlussaufgabe
$
```

6. Jetzt ist ihr Repository lokal geklont.

Readme.md erstellen

README Dateien sind hier um die Installation von Programmen einfacher zu gestalten. In dieser Datei sollte alles darüberstehen wie man dieses Programm ausführt. Die README.md Datei ist wie der Name schon verrät mit Markdown formatiert.

	spec	30.10.2024 08:40	Dateiordner	
	src	30.10.2024 08:40	Dateiordner	
	.gitignore	30.10.2024 08:40	Textdokument	1 KB
	package.json	30.10.2024 08:40	JSON-Quelldatei	1 KB
	package-lock.json	30.10.2024 08:40	JSON-Quelldatei	388 KB
	README.md	30.10.2024 08:48	Markdown-Quelld...	0 KB

1. Im Repository-Ordner eine Textdatei erstellen und diese auf README.md umbenennen (Wenn Datei Erweiterungen ausgeblendet sind funktioniert das nicht)
2. In Markdown die Installation Anleitungen schreiben.

Einrichtung der Entwicklungsumgebung

Für das Projekt haben wir ein Github account, Docker Desktop und Git bash verwendet. Die Komandozeile muss ebenfalls im Administrator ausgeführt werden können.

Verwendung von Git (Commit, Push)

Um Änderungen im Repository auf Github im Repository zu speichern, muss man folgende Anleitung befolgen.

1. Zuerst wird überprüft man den Status.

```
PS C:\Users\MarioSchwyzer\Documents\020_zli\110_development_fundamentals\020_abschlussaufgabe\docker-nodejs-sample> git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   README.md
```

2. Dann werden alle Änderung geaddet mit dem Befehl: git add .

```
PS C:\Users\MarioSchwyzer\Documents\020_zli\110_development_fundamentals\020_abschlussaufgabe\docker-nodejs-sample> git add .
```

3. Danach werden die Änderungen in einem Commit mit einer «messages» committed

```
PS C:\Users\MarioSchwyzer\Documents\020_zli\110_development_fundamentals\020_abschlussaufgabe\docker-nodejs-sample> git commit -m "Änderungen README"
[main 2ae4053] Änderungen README
1 file changed, 3 insertions(+), 1 deletion(-)
```

4. Am schluss wird der Commit mit dem push Befehl auf Github gepusht

```
PS C:\Users\MarioSchwyzer\Documents\020_zli\110_development_fundamentals\020_abschlussaufgabe\docker-nodejs-sample> git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 340 bytes | 340.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To github.com:Mariosch08/docker-nodejs-sample.git
4039b7f..2ae4053  main -> main
```

Erstellung und Nutzung von Docker-Containern

1. Zuerst wird in den Ordner des Repositorys navigiert
2. Danach wird der Befehl: `docker init` ausgeführt

```
C:\Users\MarioSchwyzer\Documents\020_zli\110_development_fundamentals\020_abschlussaufgabe\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!
Review your Docker files and tailor them to your application.
Consult README.Docker.md for information about using the generated files.

What's next?
Start your application by running → docker compose up --build
Your application will be available at http://localhost:3000
```

3. Danach werden die Fragen die sie oben sehen so beantwortet wie sie im Bild zu sehen sind
4. Jetzt wird der Docker-Container gestartet und lokal gehostet mit dem befehl:

[illegible]

- Jetzt sollte der Docker Container Lokal gehosted sein
- Um dies zu überprüfen können sie im Browser suchfeld `localhost:3000` eingeben und die ToDo sollte erscheinen

Link zum Github Repository: [Github Repository Docker nodejs](#)

Es konnten keine Einträge für ein Abbildungsverzeichnis gefunden werden.

Version	Änderungen	Datum
1.0	-Erstellung des Dokumentes -Erstellung der Dokumentation -Dokumentation Fertig gestellt	30.10.2024