



Feinstaub Projekt

By Witoschek, Specht, Flick, Ptaszyk

Inhaltsangabe

- Artefakt Dockerfile
 - Warum Docker
 - Wie nutzen wir Docker
 - Setup Skripte
- Arbeitsweise
- Projektstruktur
 - Abhängigkeiten
 - FastAPI
 - Module & Funktionen
- Live Demo & Fragen

Warum Docker

- Plattformunabhängigkeit
- Sauberes Host-System
- Minimale Umgebung für Applikation

Wie nutzen wir Docker

- Docker Volumes
- Docker Networks
- Docker Images
- Docker Container

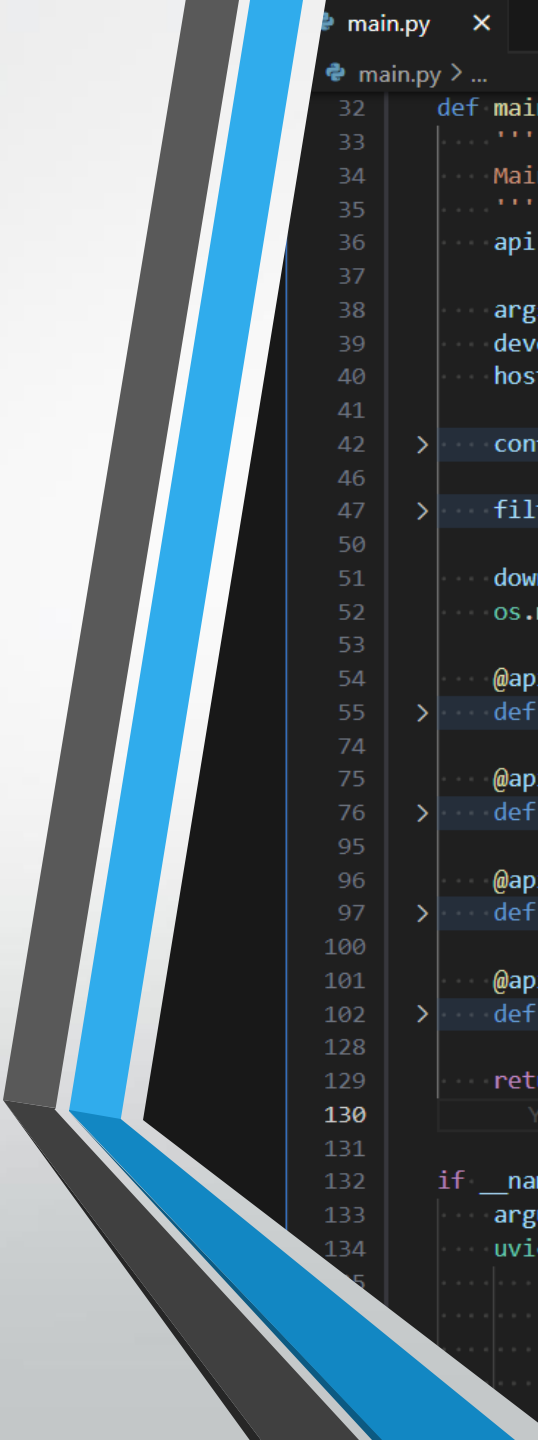
```
1 FROM python:3.10-bullseye
2 ENV PYTHONUNBUFFERED 1
3 WORKDIR /API/
4 COPY . .
5 RUN pip install -r requirements.txt
6 EXPOSE 9090
7 ENTRYPOINT ["python", "main.py"]
```

Setup Skripte

- Powershell als Einstiegspunkt
- Docker Desktop vs. WSL Docker vs. Native

Arbeitsweise

- Gemeinsames Coden (Mob-Programming)
- Weekly Zusammenfassungen
- Git



```
main.py x
main.py > ...
32 def main() -> FastAPI:
33     ...
34     ... Main entry point for uvicorn. Yields API with necessary
35     ...
36     ... api = FastAPI()
37
38     ... args = get_arguments()
39     ... develop = args.develop
40     ... host_name = 'localhost' if develop else 'feinstaub_data
41
42 > ... config = {'hostName': host_name, ...
46
47 > ... filter_map = {0: 'min', ...
50
51     ... download_path = os.path.join(os.getcwd(), 'downloaded_
52     ... os.makedirs(download_path, exist_ok=True)
53
54     ... @api.post('/insert_data', status_code=status.HTTP_201_
55 > ... def insert_data(date: Date = Date(), ...
74
75     ... @api.get('/read_data', status_code=status.HTTP_200_OK)
76 > ... def read_data(date: Date = Depends(), sensor_info: Sen
95
96     ... @api.post('/test_connection')
97 > ... def testing_database(): ...
100
101     ... @api.get('/filter_data')
102 > ... def get_filtered_data(date: Date = Depends(), ...
128
129     ... return api
130
131     You, 3 months ago • initial commit ...
132
133 if __name__ == '__main__':
134     ... arguments = get_arguments()
135     ... uvicorn.run('main:main',
136     ...     ... factory=True,
137     ...     ... host='0.0.0.0',
138     ...     ... port=9090,
139     ...     ... reload=arguments.develop)
```

Abhängigkeiten

```
fastapi==0.109.2  
psycopg[binary]==3.1.10  
requests==2.31.0  
uvicorn==0.20.0  
pydantic==2.6.4
```


FastAPI

- Benutzeroberfläche generiert
- Einfache Implementierung
- Leichter Einstieg in REST
- Pydantic zur Validierung

> database
> docs
✓ modules
 🐍 data_models.py
 🐍 database_handler.py
 🐍 downloader.py
 🐍 file_handler.py
 🐍 sensor_data.py
 🚢 .dockerignore
 💎 .gitignore
 ➤ docker_setup.ps1
 🚢 Dockerfile
 🐍 main.py
 ⓘ README.md
 ≡ requirements.txt
 ➤ wsl_docker_setup.ps1

Module & Funktionen

- Lesbarkeit vom Ablauf der Anwendung
- Strukturierung durch Module
- Modelle als Datenklasse



Live Demo & Fragen