

Spickzettel: Dockerfile – Aufbau & Beispiele

Was ist ein Dockerfile?

- Eine **Textdatei**, die alle Anweisungen enthält, um ein Docker-Image zu bauen.
- Definiert **Basis-Image, Befehle, Umgebungen, Dateien, Startbefehle**.

Grundstruktur eines Dockerfiles

```
# Basis-Image festlegen
FROM python:3.10-slim

# Arbeitsverzeichnis setzen
WORKDIR /app

# Dateien ins Image kopieren
COPY . .

# Abhängigkeiten installieren
RUN pip install -r requirements.txt

# Umgebung konfigurieren (optional)
ENV PYTHONUNBUFFERED=1

# Standardbefehl zum Starten des Containers
CMD ["python", "main.py"]
```

Wichtige Anweisungen im Dockerfile

Anweisung	Zweck
FROM	Basis-Image definieren (z. B. python:3.10)
WORKDIR	Arbeitsverzeichnis im Container setzen
COPY	Dateien vom Host in das Image kopieren
RUN	Shell-Befehl beim Imagebau ausführen
ENV	Umgebungsvariablen setzen
CMD	Standardkommando beim Containerstart
EXPOSE	Port für Kommunikation nach außen angeben
ENTRYPOINT	Fester Startpunkt, z. B. für Skripte

Beispiel: Minimaler Python-Service

```
FROM python:3.10-alpine
WORKDIR /code
COPY requirements.txt ./
RUN pip install -r requirements.txt
COPY . .
CMD ["python", "app.py"]
```

.dockerignore

```
__pycache__/  
*.pyc  
*.pyo  
.env  
venv/  
.git/
```

Best Practices

- Nur notwendige Dateien mit COPY einbinden
- .dockerignore immer verwenden
- RUN-Befehle kombinieren (&&) um Layer zu minimieren
- Für große Projekte: **Multi-Stage Build** nutzen

Nützlich zum Testen

```
# Image bauen  
docker build -t mein-image .
```

```
# Container starten  
docker run --rm mein-image
```