

## 1. Ziele

- Sie erkennen welchen Zusammenhang zwischen internem Containerport und externem Hostport besteht.

## 2. Aufgaben

Suchen Sie in dockerhub nach dem Image getting-started. Getting-started bietet eine Webseite mit elementaren Informationen zu docker an. Lösen Sie folgende Aufgaben:

- Auf welchem Port läuft die Webseite im Container

auf dem Port 80:80

- Starten Sie einen Container aus getting-started, der auf dem Host auf <http://localhost:8080> erreichbar ist

`docker run -d -p 8080:80 docker/getting-started`

- Worin könnte eine Problematik bei der Integration in ein lokales Netzwerk auftreten? Denken Sie dabei an den Fall, bei Sie bereits einen lokalen Webserver laufen haben

ein problem könnte auftauchen wenn der Port welchen man benutzen möchte schon vergeben ist und gleichzeitig am laufen ist.

- Beantworten Sie die Frage: Warum funktioniert `docker run -d --name my-apache-webserver -p 8080:80 httpd:latest` aus Aufgabe 2.3.1 nicht mehr?

weil getting-started diesen port schon genommen hat

- Lesen Sie getting-started durch

-

- Beenden Sie am Schluss den Container, löschen Sie ihn und auch das Image.

`docker rm (id)`  
`docker rmi (id)`

## 3. Erwartete Resultate

Mit Screenshots dokumentiertes und kommentiertes Vorgehen

Zeit: 30 Minuten