Getting started with containers

Managing containers and images

Two important concepts

Image

Zutaten

Hauptgericht
Für 4 **Personen**

250 g	Brot, z. B. Weggli
2 dl	Bouillon
250 g	junger Spinat
2	Eier
5 EL	Paniermehl
50 g	Reibkäse, z.B. Greyerzer
	Salz
1	Schalotte
2	Knoblauchzehen
400 g	gemischte Pilze, z. B. Eierschwämmchen und Steinpilze
½ Bund	Thymian
50 g	Butter
	rosa Pfeffer
20 g	Brunnenkresse

So gehts

Zubereitung: ca. 40 Minuten
Ruhen lassen: ca. 30 Minuten
Gesamt: **1 Std. 10 Min.**

FAST FERTIG

1

Brot in Würfelchen schneiden und in Bouillon einweichen. Spinat waschen und tropfnass in eine Pfanne geben. Erhitzen, bis er zusammenfällt. Mit kaltem Wasser abschrecken. Spinat abgiessen und gut ausdrücken. Spinat und Eier mit einem Stabmixer pürieren. Mit Paniermehl und Käse zur Brotmasse geben und gut verkneten. Mit Salz abschmecken. Masse ca. 30 Minuten ruhen lassen. Aus der Masse mit nassen Händen Knödel à ca. 50 g formen.

2

Inzwischen Schalotte und Knoblauch hacken. Pilze rüsten und nach Belieben klein schneiden. Kräuterblättchen abzupfen. Reichlich Wasser in einer weiten Pfanne aufkochen. Knödel portionenweise darin knapp unter dem Siedepunkt ca. 5 Minuten ziehen lassen, bis sie an der Oberfläche schwimmen und fest werden. Herausnehmen und abtropfen lassen. Butter in einer weiten, beschichteten Bratpfanne erhitzen. Schalotte, Knoblauch und Pilze darin bei mittlerer Hitze ca. 3 Minuten braten. Knödel dazugeben und kurz mitbraten. Mit Thymianblättchen und Pfeffer bestreuen. Mit Brunnenkresse servieren.

Rezept: Daniel Tinembart

- read-only description
- stored on longer term
- can be used as a base

Container

- based on the image
- short-lived
- usually only minor adjustments

The concept of layers

- In docker: images have a starting point
- User makes changes/installations
- Stored in a layer on top of existing
- Creating an image:
  - From a container: `docker commit` (not reproducible)
  - From a `Dockerfile`

Quiz question 3

The docker engine

- Manages both the
  - images
  - containers
- Layers are efficiently handled
- Images and containers will remain available unless specified
- Two important commands:
  - `docker image ls`
  - `docker container ls`

Sharing an image

- docker hub (open to the world)
  - Alternatives: quay.io, singularityhub, gitlab and github container repositories …
- command `docker save`
- `Dockerfile`

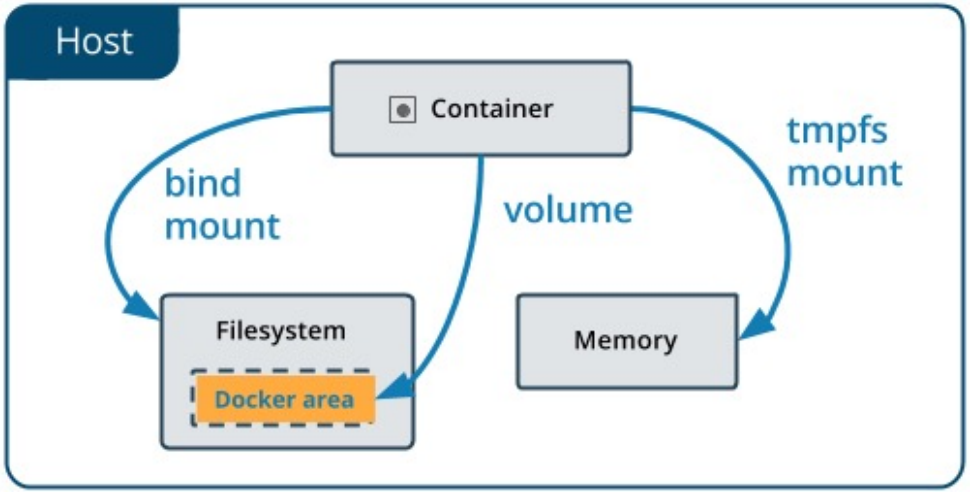
Question 4

Frequently used features

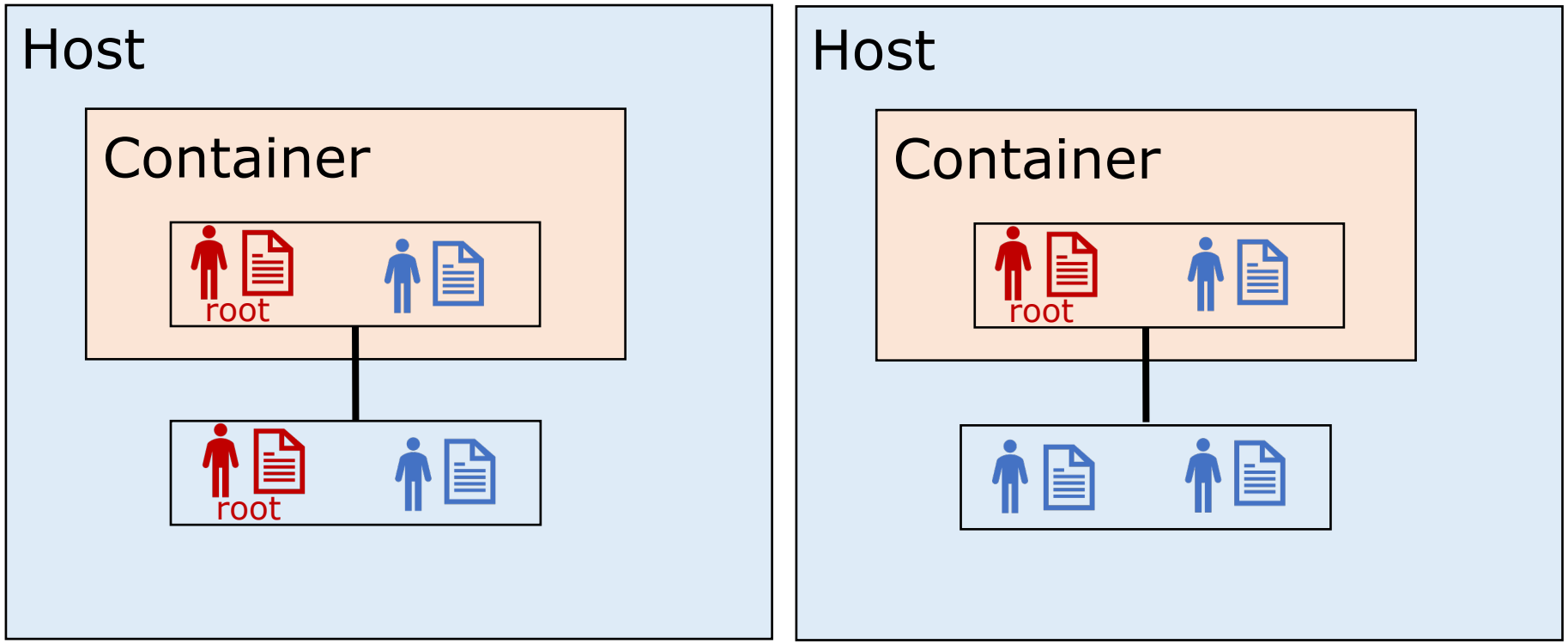
- Mounting directories
- Managing identities
- Mapping ports

Mounting

- **Bind-mount:** Make a directory on the host available to the container
- **Volume:** Disk space reserved and managed by docker (isolated)



Identity



Linux

Other systems

```
docker run -u "$(id -u):$(id -g)"
```

Mapping ports

- Processes that display browser content:
  - Jupyter
  - Rstudio server
  - Any other web server
- These are published at [IP]:[PORT], so e.g: 127.0.0.1:8000
- Forward the port from the container to port on the host: `docker run –p 80:8000`
- Meaning: publish port 8000 in the container at port 80 on the host