

# Docker

# Motivacija



# Motivacija

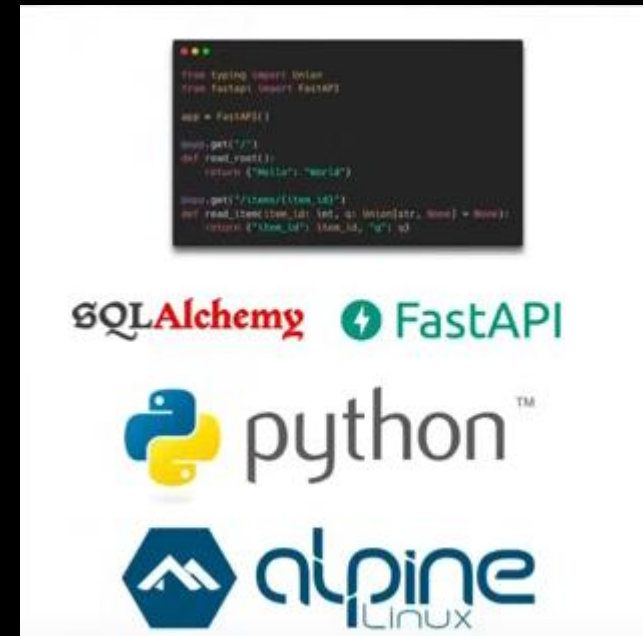
- Instaliraj ove alate
- Instaliraj DBMS
- Instaliraj ove biblioteke
- Pronijeni ovaj konfig fajl
- Idalje ne radi, koristim Windows OS, a ovo je pravljeno za Linux



- Pokreni **docker compose up**

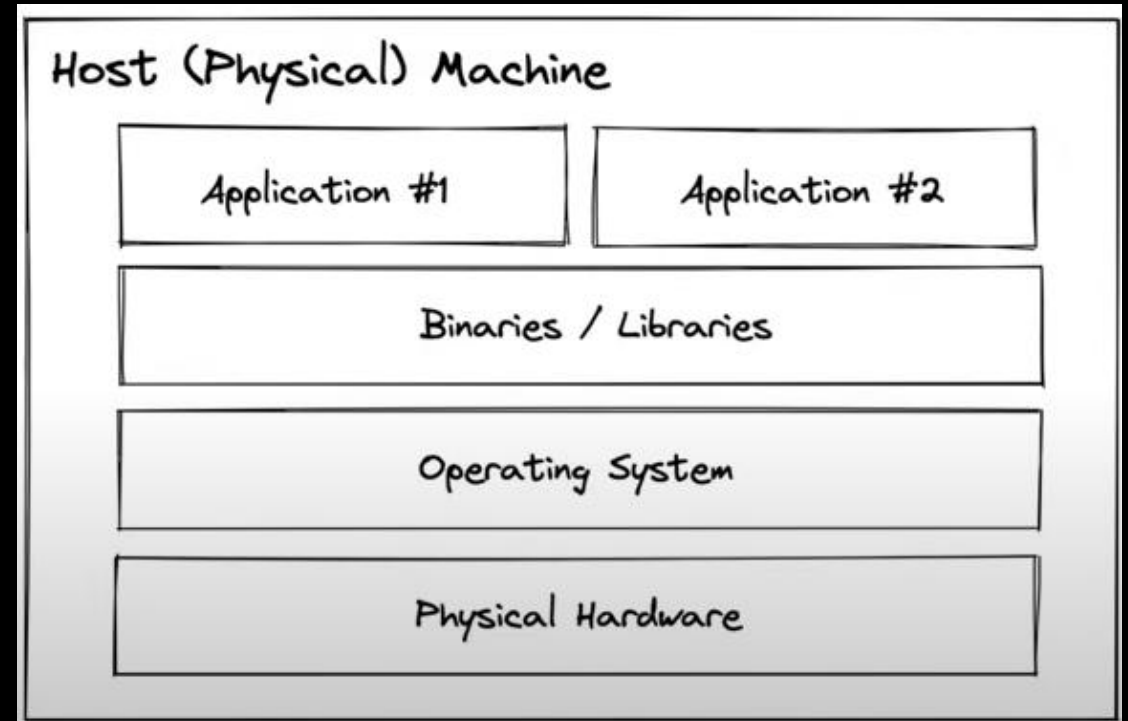
# Šta je kontejner

- Docker kontejner je skup softverskih paketa koji sadrži sve što je potrebno da bi se pokrenula aplikacija
  - OS pakete
  - Runtime okruženje (Npr Python ili Node JS)
  - Sve biblioteke koje su potrebne
  - Sadrži kod aplikacije
- Ne zahtijeva neke druge softvere osim dockera kako bi se pokrenula aplikacija
- Možemo lako pokrenuti više instanci aplikacije po potrebi



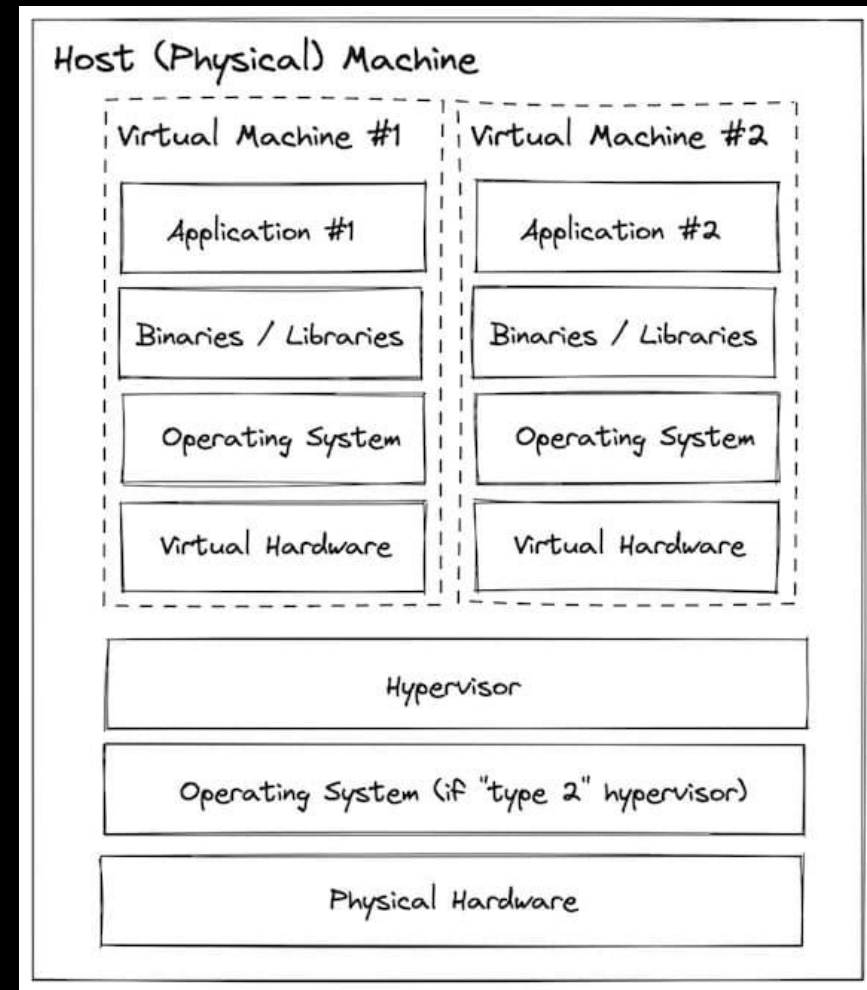
# Evolucija Virtualizacije

- Bare metal
  - Konflikt između verzija softvera
  - Slaba iskorištenost hardvera
  - Slaba izolacija problema
  - Sporo restartovanje sistema
  - Zahtjevna instalacija na novi hardver



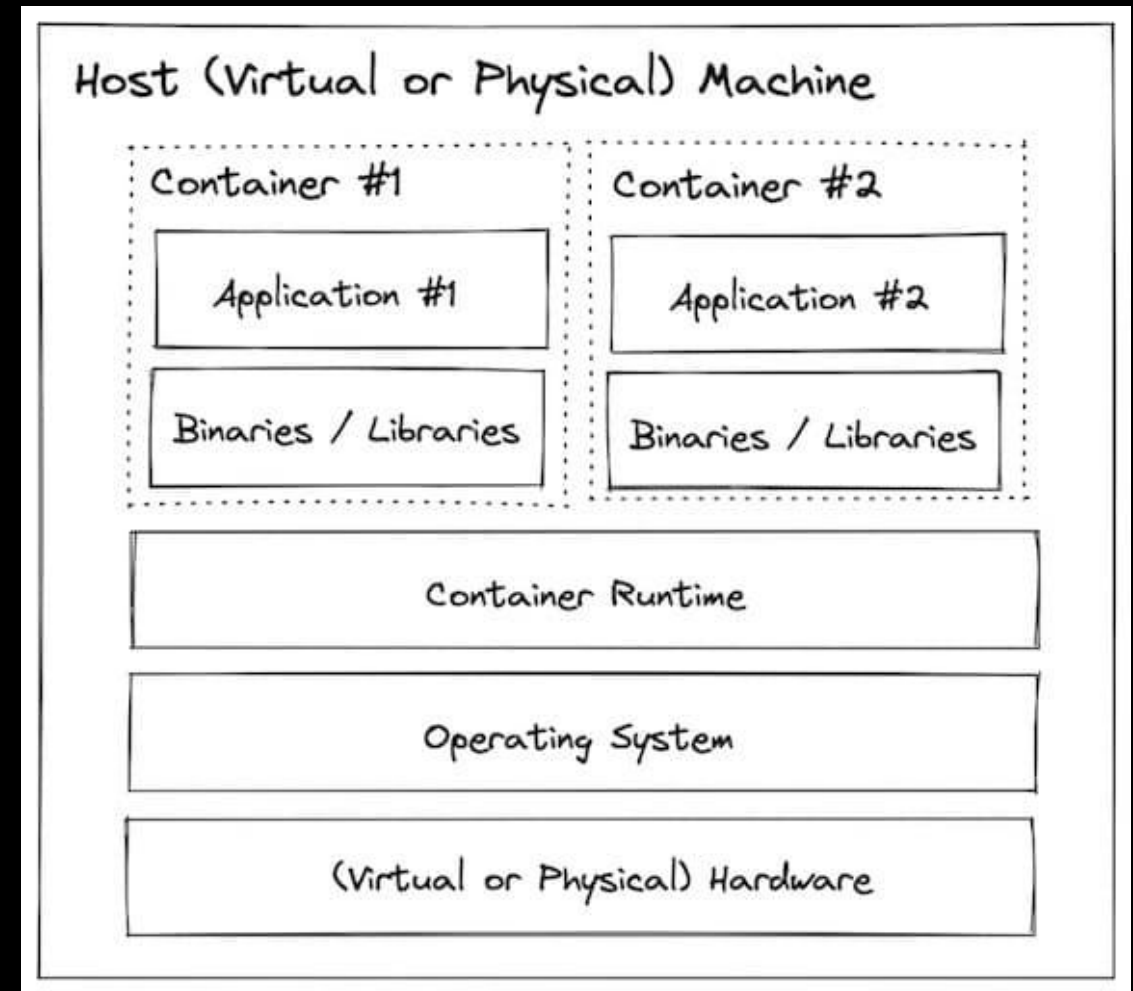
# Evolucija Virtualizacije

- Virtuelne mašine
  - Hiervizor dijeli resurse host sistema i simulira više odvojenih virtuelnih mašina
  - Nema konflikta između verzija softvera
  - Bolja iskorištenost hardvera
  - Problem na jednoj APP ne utiče na druge APP
  - Brže restartovanje sistema
  - Lakša instalacija na novi hardver



# Evolucija Virtualizacije

- Kontejneri
  - Kontejner za razliku od VM koristi kernel operativnog sistema, i koristi mogućnosti linux OS-a za izolaciju
    - cgroup, namespace i union file system
  - Nema konflikta između verzija softvera
  - Još bolja iskorištenost hardvera
  - Problem na jednoj APP ne utiče na druge APP
  - Još brže restartovanje sistema
  - Još lakša instalacija na novi hardver
  - **Manji sistemski resursi u donosu na VM**



# Instalacija Docker engine-a

- Ispratite instrukcije za vaš OS na [Install | Docker Docs](#)
- Možete da instalirate docker i na linux virtuelnoj mašini koju ste koristili za prvi dio kursa



# Pokretanje docker containera

- `docker run --interactive --tty --rm ubuntu:22.04`

Nardba za pokretanje  
Docker kontejnera

Želimo interaktivni shell  
Unutar kontejnera koji pokrecemo

Kad zaustavimo konterjner  
obriši ga, ne čuvaj ga na  
disku

Docker slika od koje želimo  
da napravimo konterjner

- `docker ps --all`

Nardba za prikazivanje Docker  
kontejnera

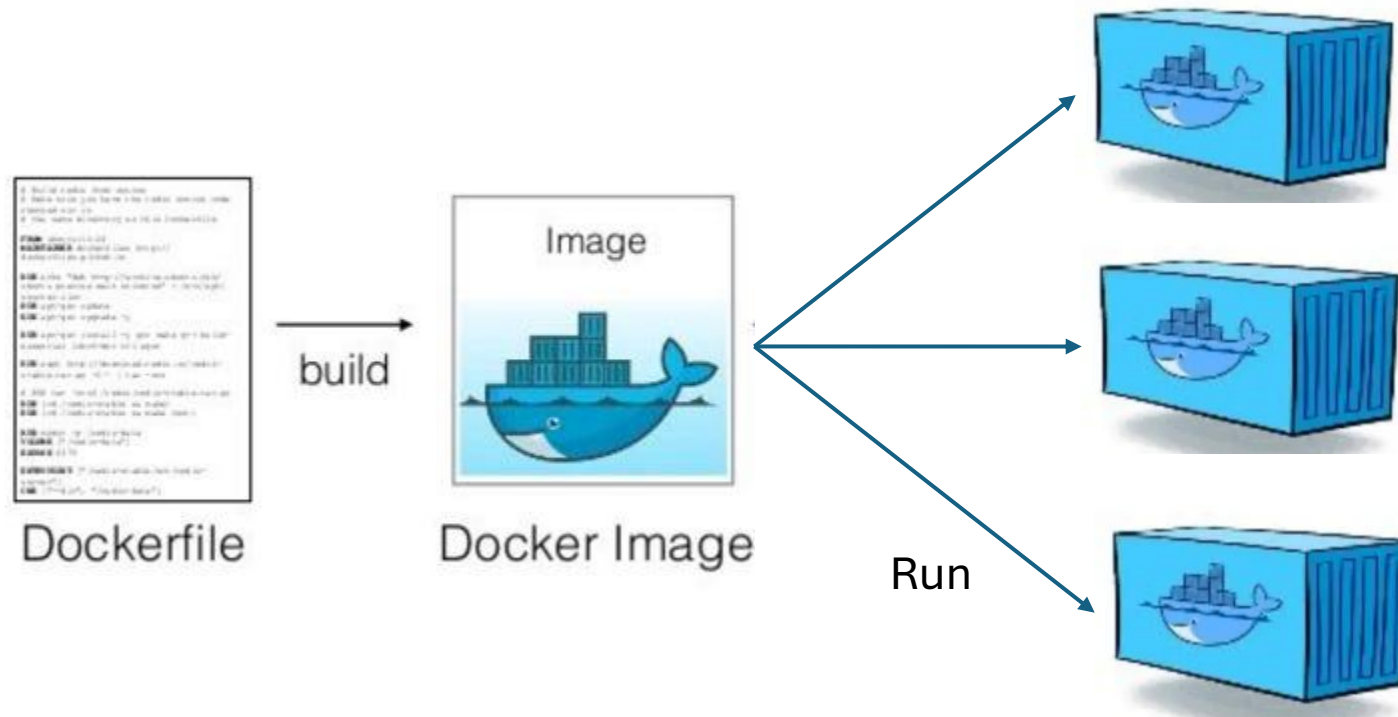
Da se prikažu i kontejneri  
koji su zaustavljeni

# Pokretanje docker containera

- `ping google.com -c 1` → Aplikacija za provjeru da li postoji data adresa u mreži i da li može da joj se pristupi
- `apt update` → Updatovanje repositorijuma sa aplikacijama koje mogu da se instaliraju na Debian based operativnim sistemima
- `apt install iputils-ping` → Instaliranje aplikacije ping
- `exit` → Zatvaranje terminala. Zaustavice i kontejner u kojem se nalazimo

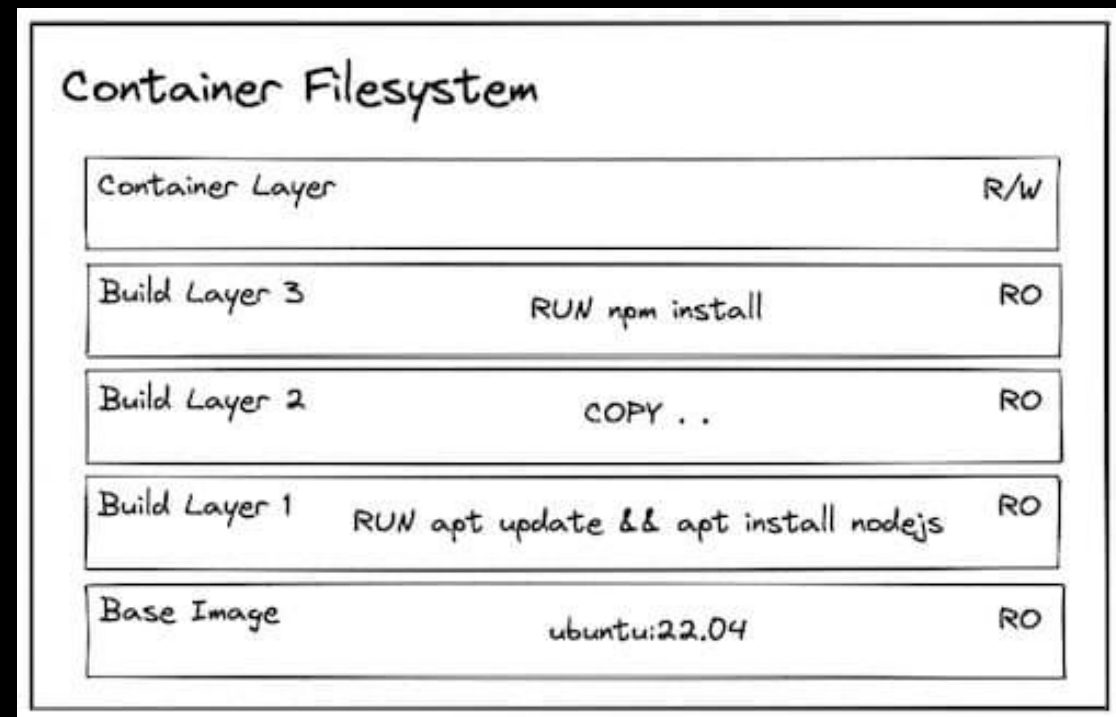
# Pokretanje docker containera

- Kad zaustavimo docker kontejner, sve što smo ranije instalirali i fajlovi koje smo kreirali u tom kontejneru biće obrisani.



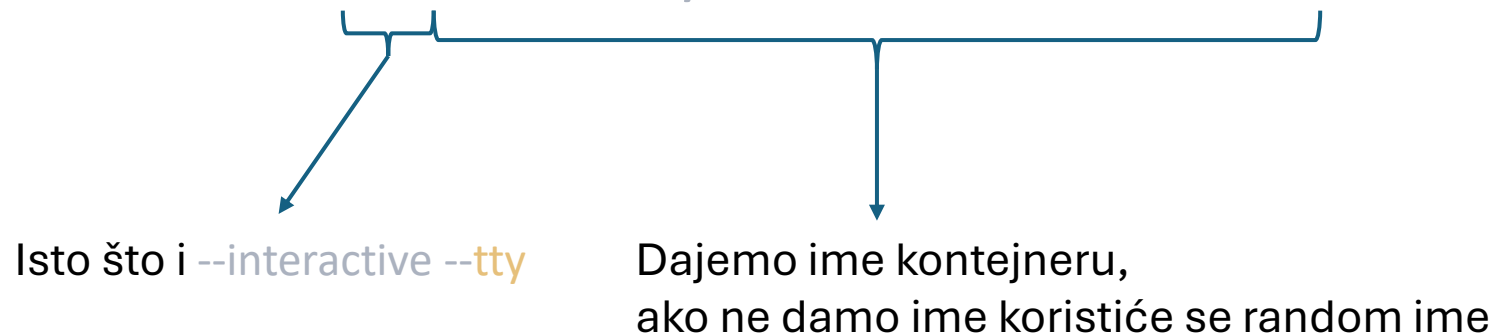
# Fajl sistem kontejnera

- Kontejner se kreira iz slike
- Sve što je u slici se tretira kao read-only
- Dodaje se samo novi sloj po kojem će da se piše dok je kontejner aktivan



# Kontejner koji se čuva i nakon što se zaustavi

- `docker run -it --name my-ubuntu-container ubuntu:22.04`



- `docker stop my-ubuntu-container`
- `docker container ps -a | grep my-ubuntu-container`
- `docker container inspect my-ubuntu-container`
- `docker start my-ubuntu-container`
- `docker attach my-ubuntu-container`

# Kreiranje docker slike sa instaliranim softverom

- Možemo da kriramo novu docker sliku iz postojeće i da joj dodamo sve što je nama potrebno.
- `docker build --tag my-ubuntu-image -<<EOF FROM ubuntu:22.04 RUN apt update && apt install iputils-ping --yes EOF`
- *# Run a container based on that image*
- `docker run -it --rm my-ubuntu-image`
- *# Confirm that ping was pre-installed*
- `ping google.com -c 1`

# Podaci u docker kontejneru

- Svi podaci koje kreiramo u docker kontejneru se brišu nakon zaustavljanja kontejnera
  - Briše se ContainerLayer koji služi za pisanje dok je container aktivan
- Potrebno je da bindujemo neki folder iz host os tako da se on prikazuje u dokcer containeru

# Podaci u docker kontejneru

- *# Create a container from the ubuntu image*
- `docker run -it --rm ubuntu:22.04`
- *# Make a directory and store a file in it*
- `mkdir my-data`
- `echo "Hello from the container!" > /my-data/hello.txt`
- *# Confirm the file exists*
- `cat my-data/hello.txt`
- `exit`



# Podaci u docker kontejneru

- *# Create a container from the ubuntu image*
- `docker run -it --rm ubuntu:22.04`
- *# Check if the file exists*
- `cat my-data/hello.txt`
- *# Produces error: `cat: my-data/hello.txt: No such file or directory`*

---

To Be  
Continued

---