

Docker Labs

lab 0

problem 1

- sudo docker run node
- sudo docker run -v ./etc/node node
- sudo docker cp <source path><container>:<destination path>
sudo docker cp <container>:<source path><destination path>
- sudo docker exec -it my_container python /app/myscript.py
- sudo docker run -d -p 80:80 nginx
- sudo docker logs my-nginx-container
- sudo docker images
- sudo docker ps -a
- sudo docker stop my-container
- sudo docker container prune
- sudo docker image inspect my-image
- A Docker container is a runnable instance of a Docker image.
sudo docker run my-image

lab 1

problem 1

- sudo docker run hello-world
- sudo docker ps -a
- sudo docker start my-container
- sudo docker rm my-container
- sudo docker rmi my-image

problem 2

- sudo docker run -it <ubuntu OR centos>
- sudo docker run -it ubuntu echo docker
- sudo touch hello-docker
- sudo docker stop my-container
sudo docker rm my-container

problem 3

- sudo docker run -it --name=nginx --mount source=data,destination=/data nginx
- sudo docker run -d -p 3000:80 --name=nginx -v /data:/data/html nginx
- sudo docker rm nginx
- sudo docker run -d -p 9898:80 --name=my-nginx-new-container --mount
source=data,destination=/usr/share/nginx/html nginx

problem 4

- sudo docker run -d -p 8080:80 --name=my-nginx-no-volume nginx
- sudo docker cp [html_file_path] my-nginx-no-volume:/usr/share/nginx/html/

- sudo docker commit my-nginx-no-volume my-nginx
- nano Dockerfile

```
- FROM nginx:latest
COPY ./[html_file] /usr/share/nginx/html
EXPOSE 80
```

- sudo docker build -t my-nginx-image .

problem 5

- sudo docker volume create mysql_data
- sudo docker run -d --name app-database -e MYSQL_ROOT_PASSWORD=P4sSw0rd0! -v mysql_data:/var/lib/mysql mysql:latest

lab 2

problem 1

- mkdir my-nginx-image
 - cd my-nginx-image
 - nano Dockerfile
- ```
- FROM ubuntu:latest
RUN apt-get update && apt-get install -y nginx
RUN rm /var/www/html/index.nginx-debian.html
COPY ./index.html /var/www/html
EXPOSE 80
CMD ["nginx", "-g", "daemon off;"]
```
- sudo docker build -t my-nginx-image .
  - sudo docker run -d -p 3000:80 my-nginx-image

### ### problem 2

- npx create-react-app my-react-app
  - cd my-react-app
  - nano Dockerfile
- ```
- FROM node:alpine AS development
ENV NODE_ENV development
WORKDIR /react-app
COPY ./package.json /react-app
RUN npm install
COPY . .
CMD npm start
```

- nano .dockerignore
- node_modules
- npm-debug.log
- build
- .git
- *.md
- .gitignore
- sudo docker build -t my-react-app .
- sudo docker run -d -p 3000:3000 my-react-app

problem 3

- mkdir flask
- cd flask
- nano app.py
- from flask import Flask
- from redis import Redis
- app = Flask(__name__)
- redis = Redis(host='redis', port=6379)
- @app.route('/')
 - def count_visits():
 - visits = redis.incr('visits')
 - return f'Number of visits: {visits}'
- if __name__ == '__main__':
 - app.run(host='0.0.0.0', port=80)
- nano requirements.txt
- Flask==2.1.1
- Werkzeug==2.1.1
- redis==4.5.2
- nano Dockerfile
- FROM python:3.9-slim
- WORKDIR /app
- COPY requirements.txt .
- RUN pip install --trusted-host pypi.python.org -r requirements.txt
- COPY . .

EXPOSE 80

CMD ["python", "app.py"]

- nano docker-compose.yml

- version: '3'

services:

web:

build: .

ports:

- "8080:80"

redis:

image: "redis:latest"

- sudo docker compose up --build