Docker Labs
lab 0
problem 1

- How do you run a Node.js application using Docker, using the official Node.js image?
- What command would you use to mount a local directory into a Node.js Docker container?
- What is the command to copy files from your local machine into a running Python Docker container and vice versa?
- How can you execute a Python script inside a running Docker container?
- How do you start a Nginx Docker container and expose it on port 80?
- What command can you use to inspect the Nginx container's logs?
- How can you list all Docker images on your system?
- What is the purpose of the docker ps command, and how can you see all containers, including stopped ones?
- How can you stop a running Docker container gracefully?
- What command would you use to remove all stopped containers from your system?
- How can you view detailed information about a Docker image, including its layers?
- What is the difference between a Docker image and a Docker container, and how do you create a container from an image?

lab 1
problem 1

- Run the container hello-world
- Check the container status
- Start the stopped container
- Remove the container
- Remove the image

problem 2

- Run container centos or ubuntu in an interactive mode
- Run the following command in the container “echo docker ”
- Open a bash shell in the container and touch a file named hello-docker
- Stop the container and remove it. Write your comment about the file hello-docker
-

problem 3

- Run a container nginx with name nginx and attach a volume to the container
- Volume for containing static html file
- Remove the container
- Run a new container with the following:
  - Attach the volume that was attached to the previous container
  - Map port 80 to port 9898 on you host machine
  - Access the html files from your browser

problem 4

- Run the image nginx again without attaching any volumes
- Add html static files to the container and make sure they are accessible
- Commit the container with image name my nginx
- Create a dockerfile for ngnix and build the image from this dockerfile

problem 5

- Create a volume called mysql_data, then:
  - deploy a MySQL database called app-database.
  - use the mysql latest image
  - use the -e flag to set MYSQL_ROOT_PASSWORD to P4sSw0rd0!.
  - mount the mysql_data volume to /var/lib/mysql.
  - the container should run in the background.

lab 2

problem 1

- Create your own nginx docker image based on ubuntu “NEVER USE FROM nginx”
  - Install nginx
  - index.html one as file
  - Expose
  - Start
  - Port mapping

problem 2

- Create react app docker container "using Dockerfile"

problem 3

- Create flask app to count number of visits to browser:
  - Create new directory called flask then add app.py and requirements.txt files
  - Create Dockerfile for the python app
  - Create docker-compose for the app and use Redis as temp DB.