Установка cowsay:

sudo apt install cowsay

Запуск cowsay:

/usr/games/cowsay “Hello”

/usr/games/cowthink “Hello”

vi Dockerfile:

FROM ubuntu:22.10 (исходный или родительский слой)

RUN apt-get update && \ (обновляем существующие пакеты)

apt-get install -y cowsay && \ (и устанавливаем cowsay внутри контейнера)

ln -s /usr/games/cowsay /usr/bin/cowsay && \ (символьная ссылка на /usr/games/cowsay, ее

положим в /usr/bin/cowsay. Она нужна нужна для запуска cowsay без указания

полного пути)

rm -rf /var/lib/apt/lists/\* (удаляем все кэши, которые были добавлены на этапе update.

Обновление и очистка должны быть на одном слое!!!)

CMD ["cowsay"] (на вход передаем слово)

docker build -t cowsaytest . – собираем образ (cowsaytest – это имя образ, а точка – это текущая директория, здесь находится Dockerfile)

docker run –it cowsaytest bash – запуск контейнера в интерактивном режиме, мы попадаем внутрь контейнера. Внутри контейнера мы можем запустить наше приложение cowsay:

cowsay “Hi”

для выхода из контейнера: exit

для запуска нашего приложения без входа в контейнер:

docker run cowsaytest cowsay “Hi”

docker run -d cowsaytest – запуск контейнера в фоновом режиме

FROM alpine (легковесный базовый образ)

RUN apk update && \ (обновляем существующие пакеты)

apk add cowsay && \ (и устанавливаем cowsay внутри контейнера)

--update-cache \

--repository <https://alpine.global.ssl.fastly.net/alpine/edge/community> \

--repository [https://alpine.global.ssl.fastly.net/alpine/edge/main \](https://alpine.global.ssl.fastly.net/alpine/edge/main%20\)

--repository <https://dl-3.alpinelinux.org/alpine/edge/testing> && \

rm -rf /var/cache/apk/\* (удаляем все кэши, которые были добавлены на этапе update.

Обновление и очистка должны быть на одном слое!!!)

CMD ["cowsay"] (на вход передаем слово)

FROM alpine

RUN echo "123" > /test

RUN apk update && apk add cowsay \

--update-cache \

--repository https://alpine.global.ssl.fastly.net/alpine/edge/community \

--repository https://alpine.global.ssl.fastly.net/alpine/edge/main \

--repository https://dl-3.alpinelinux.org/alpine/edge/testing && \

rm -rf /var/cache/apk/\*

CMD ["cowsay"]

семинар

docker run -it sleep600 bash попали внутрь контейнера

vi Dockerfile

FROM ubuntu:22.10

COPY example.txt /

CMD sleep 6

2)

vi Dockerfile

FROM ubuntu:22.10

RUN apt update && \

apt install -y nginx && \

apt install -y cowsay && \

ln -s /usr/games/cowsay /usr/bin/cowsay && \

rm -rf /var/lib/apt/lists/\*

RUN echo "Hello, world!" \

> /usr/share/nginx/html/index.html

EXPOSE 80

docker run --name hello -d hello

3) работа с nginx

FROM ubuntu:22.10

RUN apt update && \

apt install -y nginx && \

apt install -y cowsay && \

ln -s /usr/games/cowsay /usr/bin/cowsay && \

rm -rf /var/lib/apt/lists/\*

RUN echo "Hello, world!" \

> /usr/share/nginx/html/index.html

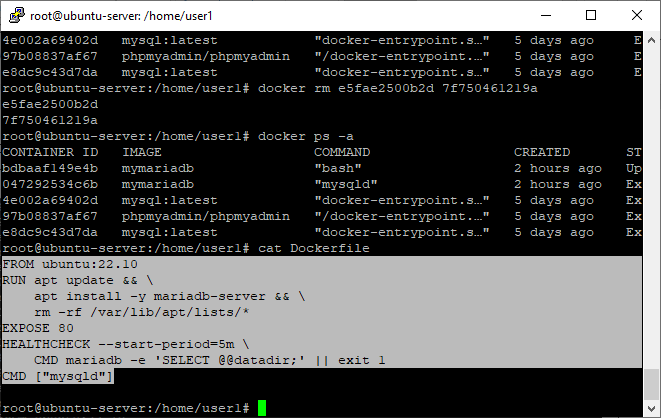
EXPOSE 80

CMD sleep 600

sudo docker run –d –p 8081:80 hello

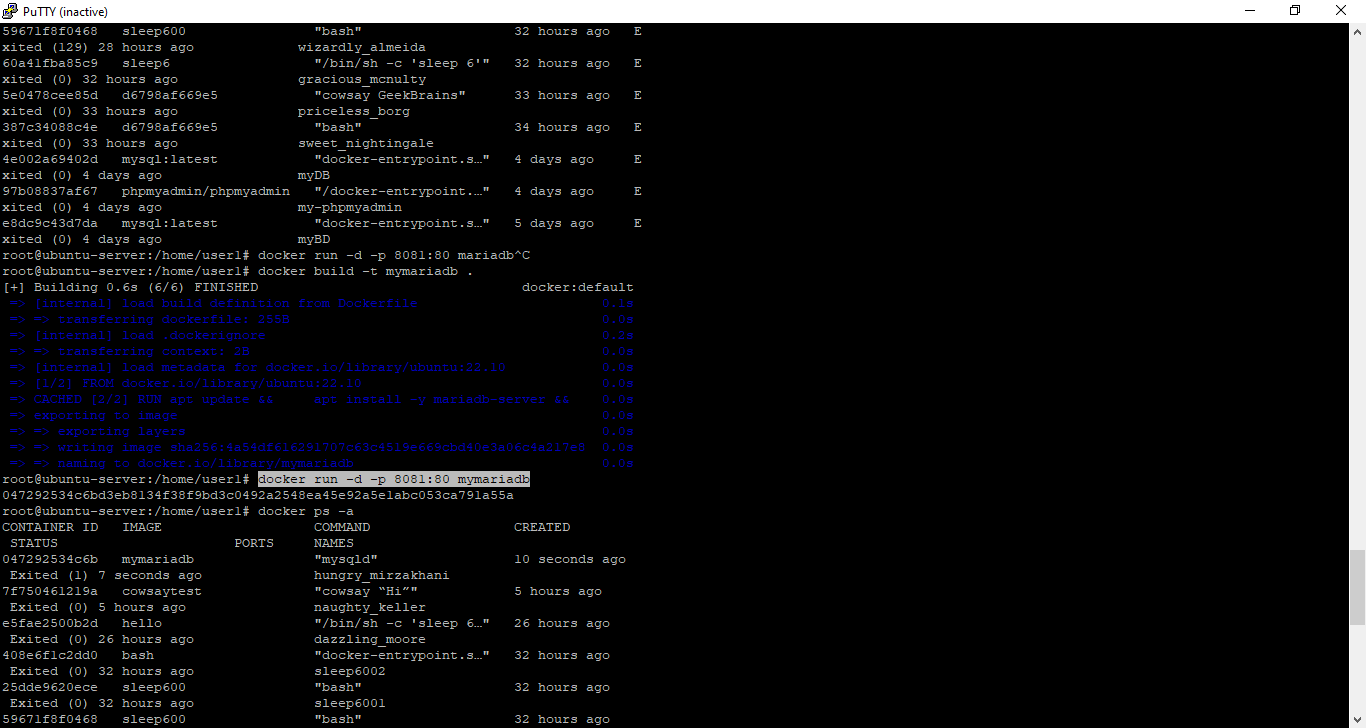
sudo curl 0.0.0.0:8081

ДОМАШНЕЕ ЗАДАНИЕ

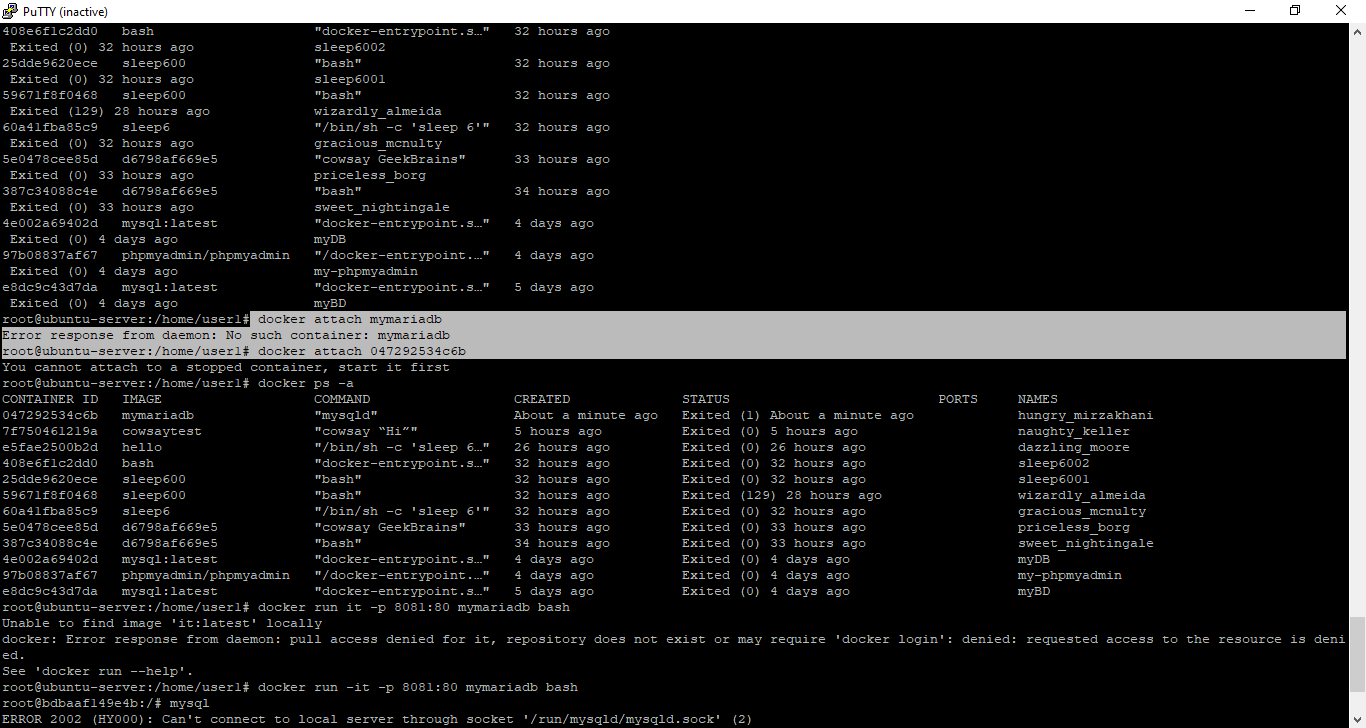


docker build -t mymariadb .

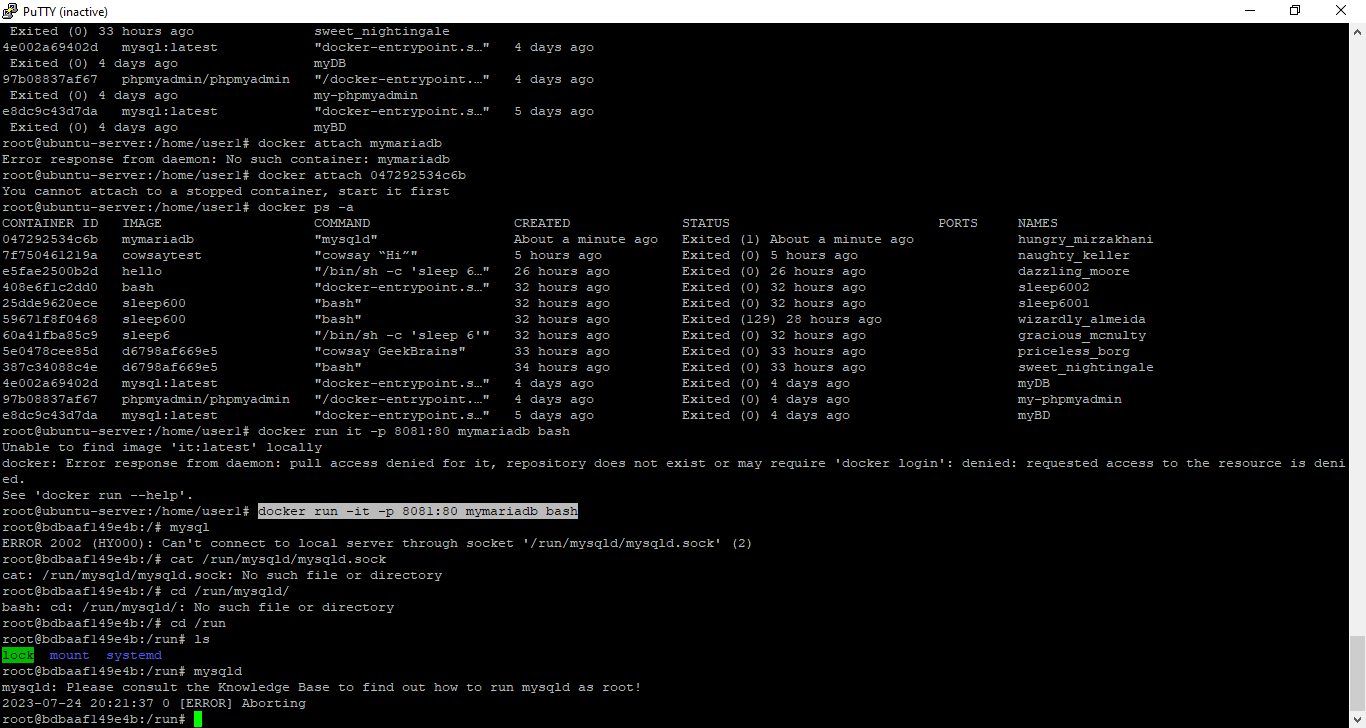
docker run -d -p 8081:80 mymariadb

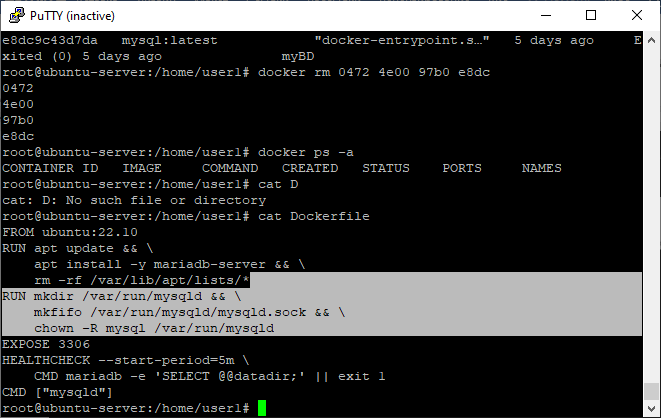


docker attach 047292534c6b



docker run -it -p 8081:80 mymariadb bash





FROM ubuntu:22.10

RUN apt update && \

apt install -y mariadb-server && \

rm -rf /var/lib/apt/lists/\* && \

mkdir /var/run/mysqld && \

mkfifo /var/run/mysqld/mysqld.sock && \

chown -R mysql /var/run/mysqld

EXPOSE 3306

HEALTHCHECK --start-period=5m \

CMD mariadb -e 'SELECT @@datadir;' || exit 1

CMD ["mysqld"]

