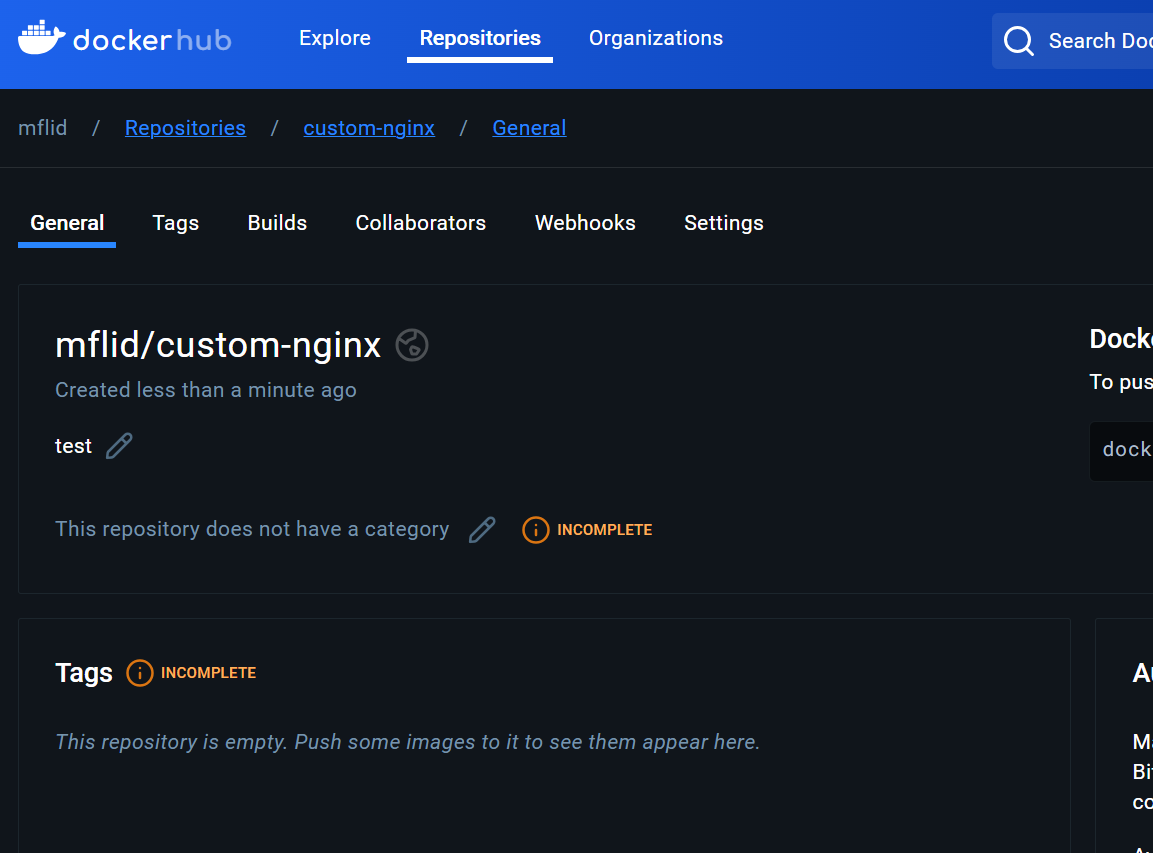
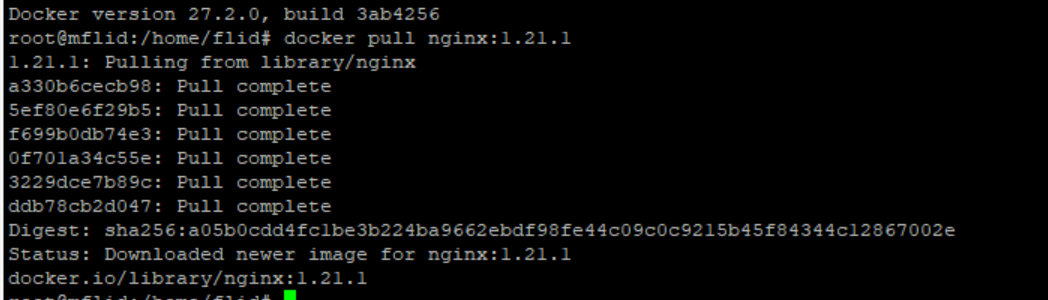
**Домашнее задание к занятию 4 «Оркестрация группой Docker контейнеров на примере Docker Compose»**

Задача 1

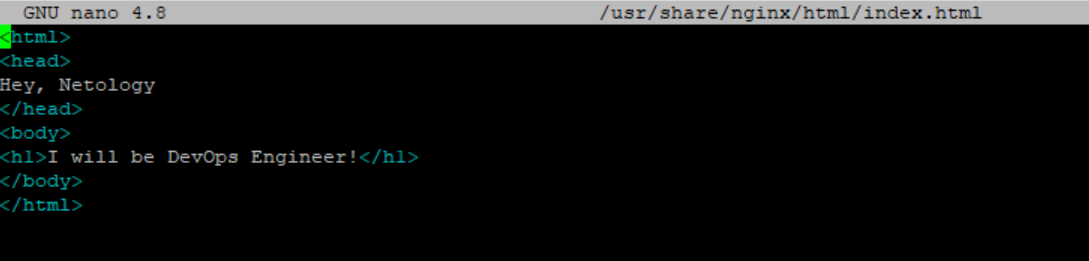
Зарегистрируйтесь и создайте публичный репозиторий с именем "custom-nginx" на [https://hub.docker.com](https://hub.docker.com/) (ТОЛЬКО ЕСЛИ У ВАС ЕСТЬ ДОСТУП);

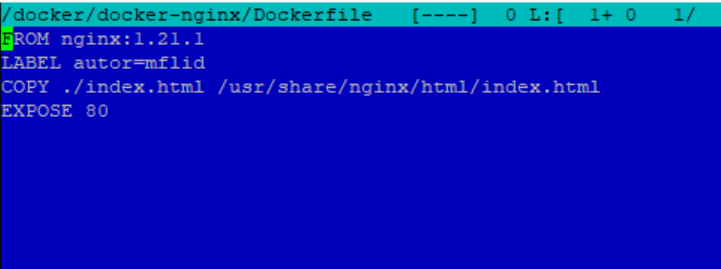


скачайте образ nginx:1.21.1;

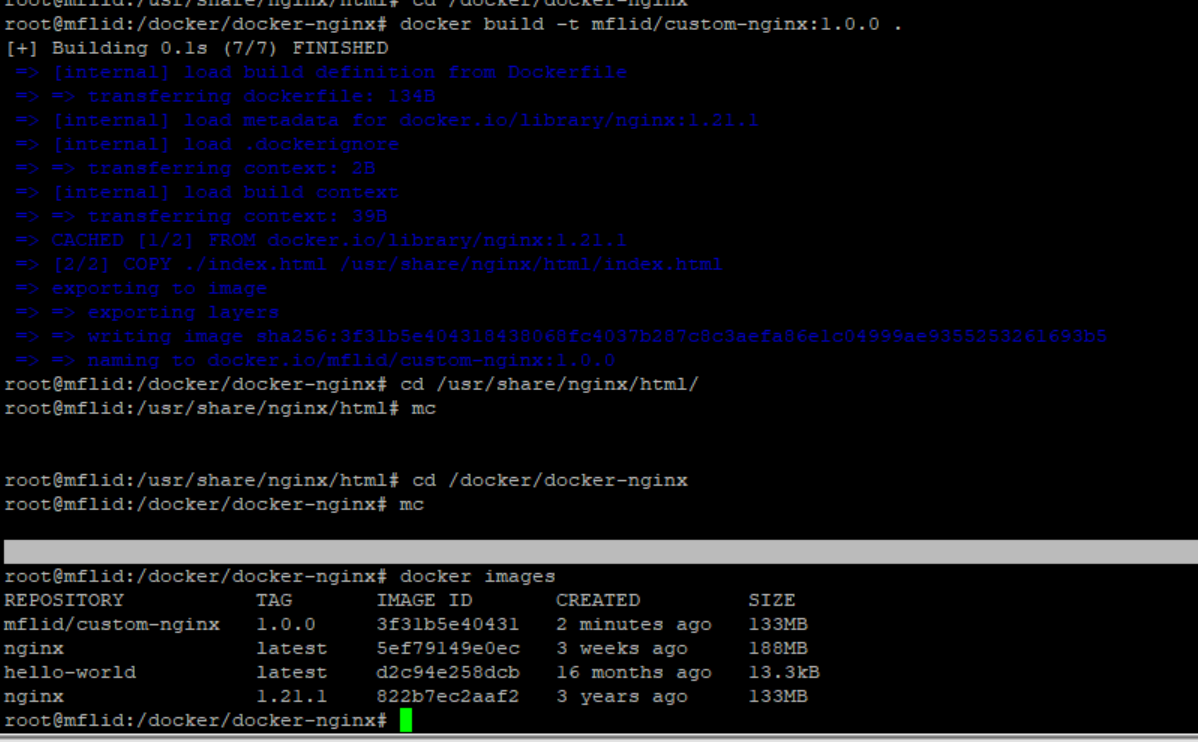


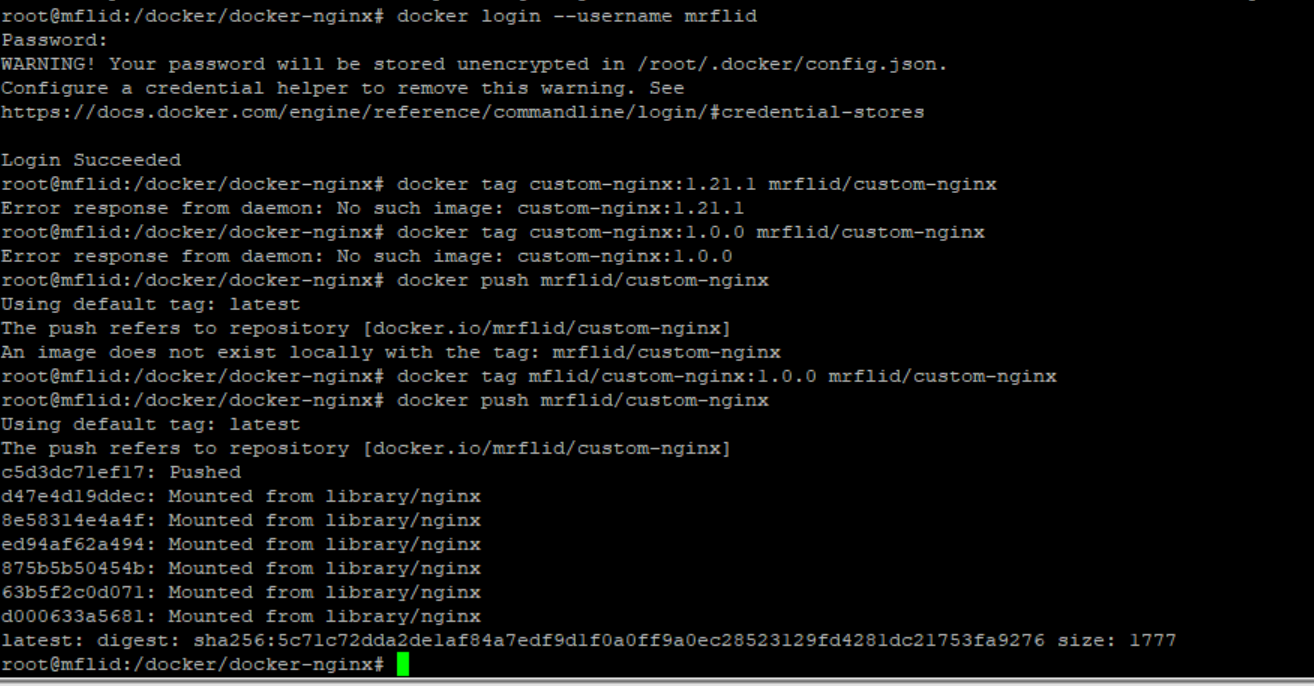
Создайте Dockerfile и реализуйте в нем замену дефолтной индекс-страницы(/usr/share/nginx/html/index.html), на файл index.html с содержимым:

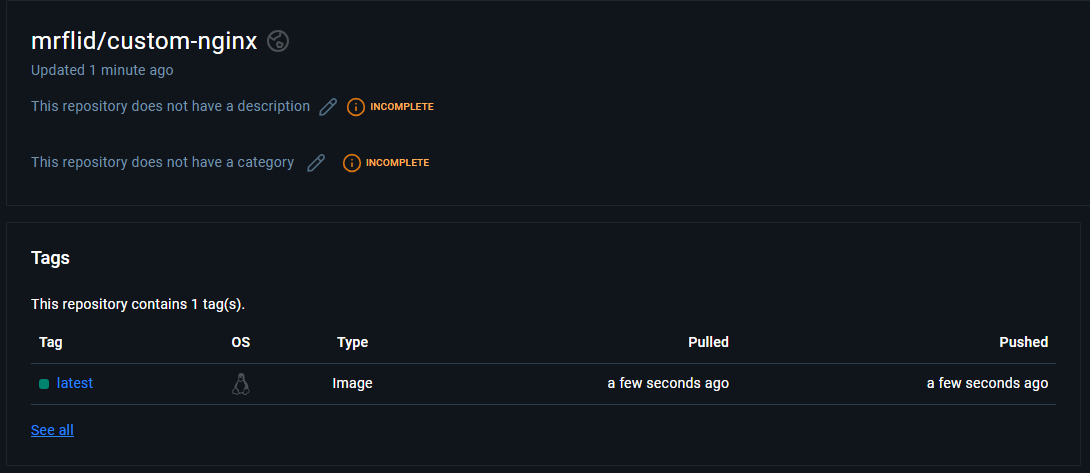




Соберите и отправьте созданный образ в свой dockerhub-репозитории c tag 1.0.0 (ТОЛЬКО ЕСЛИ ЕСТЬ ДОСТУП).







Ссылка - https://hub.docker.com/repository/docker/mrflid/custom-nginx/general

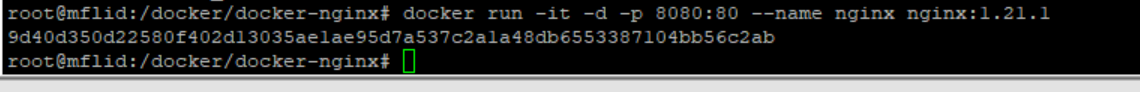
Задание 2

Запустите ваш образ custom-nginx:1.0.0 командой docker run в соответвии с требованиями:

имя контейнера "ФИО-custom-nginx-t2"

контейнер работает в фоне

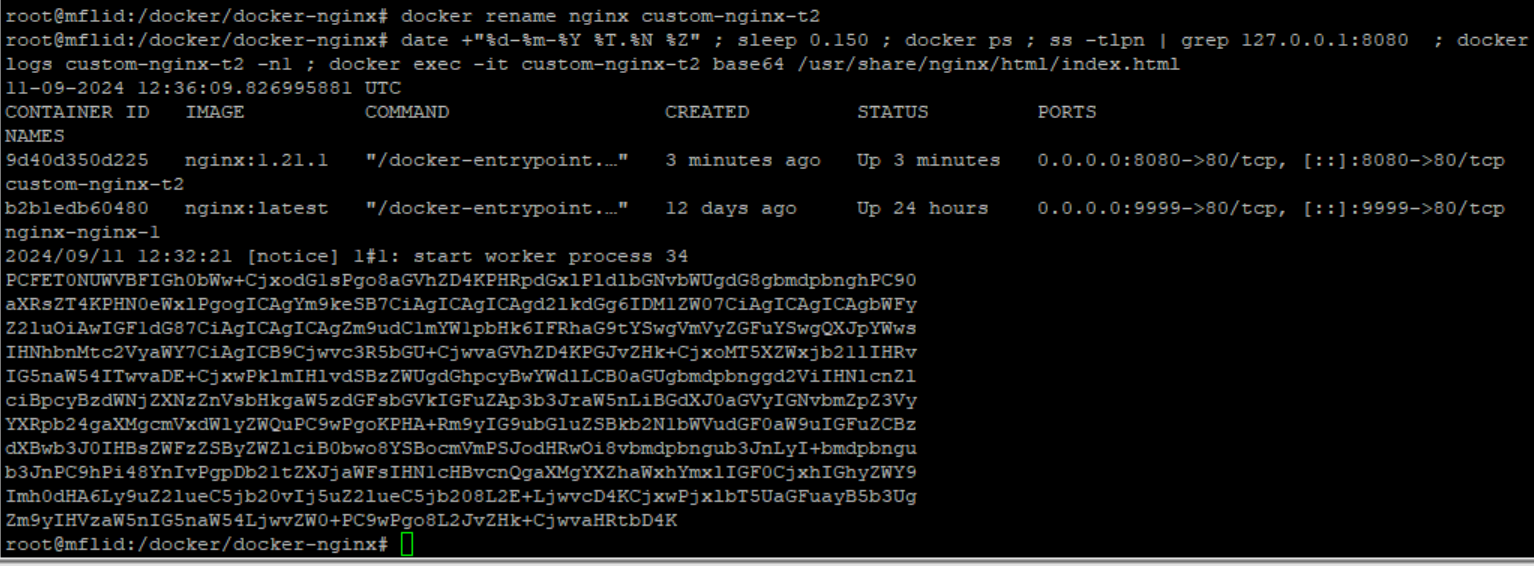
контейнер опубликован на порту хост системы 127.0.0.1:8080



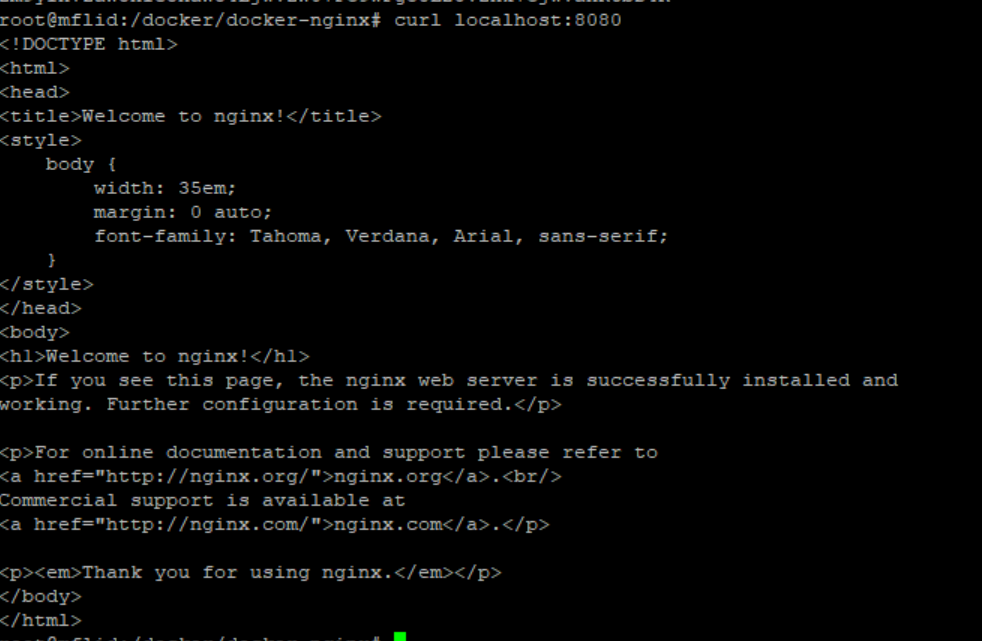
Не удаляя, переименуйте контейнер в "custom-nginx-t2"

Выполните команду date +"%d-%m-%Y %T.%N %Z" ; sleep 0.150 ; docker ps ; ss -tlpn | grep 127.0.0.1:8080 ; docker logs custom-nginx-t2 -n1 ; docker exec -it custom-nginx-t2 base64 /usr/share/nginx/html/index.html

Убедитесь с помощью curl или веб браузера, что индекс-страница доступна.



Убедитесь с помощью curl или веб браузера, что индекс-страница доступна.



Задание 3

Воспользуйтесь docker help или google, чтобы узнать как подключиться к стандартному потоку ввода/вывода/ошибок контейнера "custom-nginx-t2".

Подключитесь к контейнеру и нажмите комбинацию Ctrl-C.

Выполните docker ps -a и объясните своими словами почему контейнер остановился.   
Ответ: ctrl+c это сочетание клавишдля завершения процесса

Перезапустите контейнер

Зайдите в интерактивный терминал контейнера "custom-nginx-t2" с оболочкой bash.

Установите любимый текстовый редактор(vim, nano итд) с помощью apt-get.

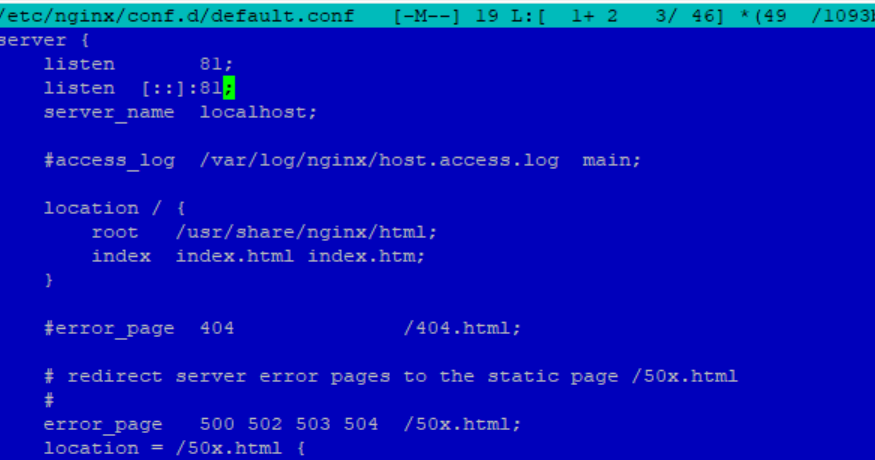


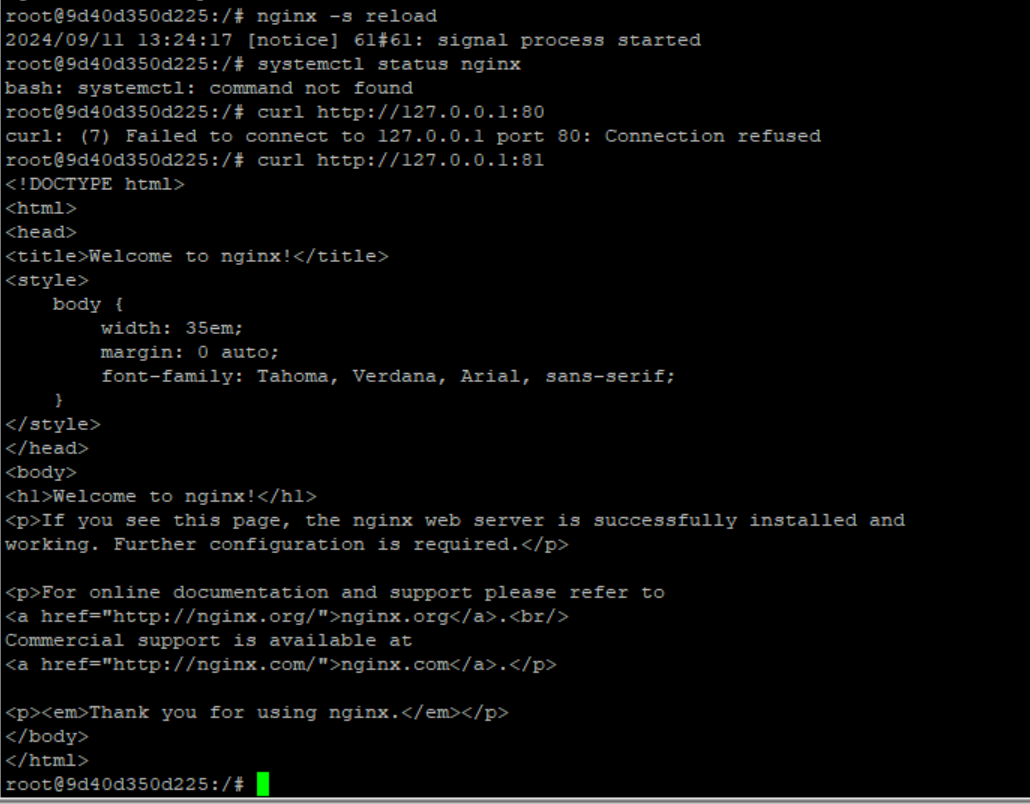
Отредактируйте файл "/etc/nginx/conf.d/default.conf", заменив порт "listen 80" на "listen 81".

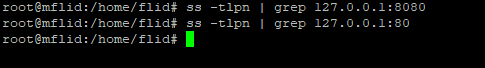
Запомните(!) и выполните команду nginx -s reload, а затем внутри контейнера curl http://127.0.0.1:80 ; curl http://127.0.0.1:81.

Выйдите из контейнера, набрав в консоли exit или Ctrl-D.

Проверьте вывод команд: ss -tlpn | grep 127.0.0.1:8080 , docker port custom-nginx-t2, curl http://127.0.0.1:8080. Кратко объясните суть возникшей проблемы.



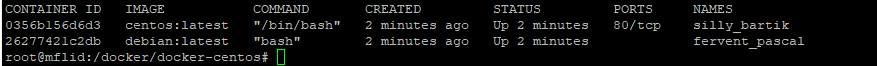




Задача №4

Запустите первый контейнер из образа ***centos*** c любым тегом в фоновом режиме, подключив папку текущий рабочий каталог $(pwd) на хостовой машине в /data контейнера, используя ключ -v.

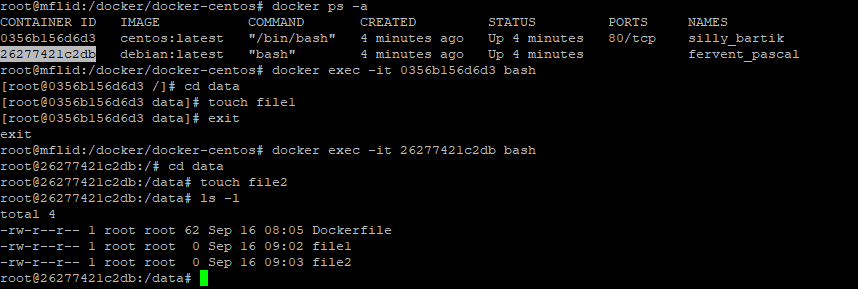
Запустите второй контейнер из образа ***debian*** в фоновом режиме, подключив текущий рабочий каталог $(pwd) в /data контейнера.



Подключитесь к первому контейнеру с помощью docker exec и создайте текстовый файл любого содержания в /data.

Добавьте ещё один файл в текущий каталог $(pwd) на хостовой машине.

Подключитесь во второй контейнер и отобразите листинг и содержание файлов в /data контейнера.



Задача 5

Создайте отдельную директорию(например /tmp/netology/docker/task5) и 2 файла внутри него. "compose.yaml" с содержимым:

*version: "3"*

*services:*

*portainer:*

*image: portainer/portainer-ce:latest*

*network\_mode: host*

*ports:*

*- "9000:9000"*

*volumes:*

*/var/run/docker.sock:/var/run/docker.sock*

*"docker-compose.yaml" с содержимым:*

*version: "3"*

*services:*

*registry:*

*image: registry:2*

*network\_mode: host*

*ports:*

*- "5000:5000"*

И выполните команду "docker compose up -d". Какой из файлов был запущен и почему? (подсказка: <https://docs.docker.com/compose/compose-application-model/#the-compose-file> )

Отредактируйте файл compose.yaml так, чтобы были запущенны оба файла. (подсказка: <https://docs.docker.com/compose/compose-file/14-include/>)

Выполните в консоли вашей хостовой ОС необходимые команды чтобы залить образ custom-nginx как custom-nginx:latest в запущенное вами, локальное registry. Дополнительная документация: <https://distribution.github.io/distribution/about/deploying/>

Откройте страницу "[https://127.0.0.1:9000](https://127.0.0.1:9000/)" и произведите начальную настройку portainer.(логин и пароль адмнистратора)

Откройте страницу "<http://127.0.0.1:9000/#!/home>", выберите ваше local окружение. Перейдите на вкладку "stacks" и в "web editor" задеплойте следующий компоуз:

- "9090:80"

Перейдите на страницу "<http://127.0.0.1:9000/#!/2/docker/containers>", выберите контейнер с nginx и нажмите на кнопку "inspect". В представлении <> Tree разверните поле "Config" и сделайте скриншот от поля "AppArmorProfile" до "Driver".

Удалите любой из манифестов компоуза(например compose.yaml). Выполните команду "docker compose up -d". Прочитайте warning, объясните суть предупреждения и выполните предложенное действие. Погасите compose-проект ОДНОЙ(обязательно!!) командой.

