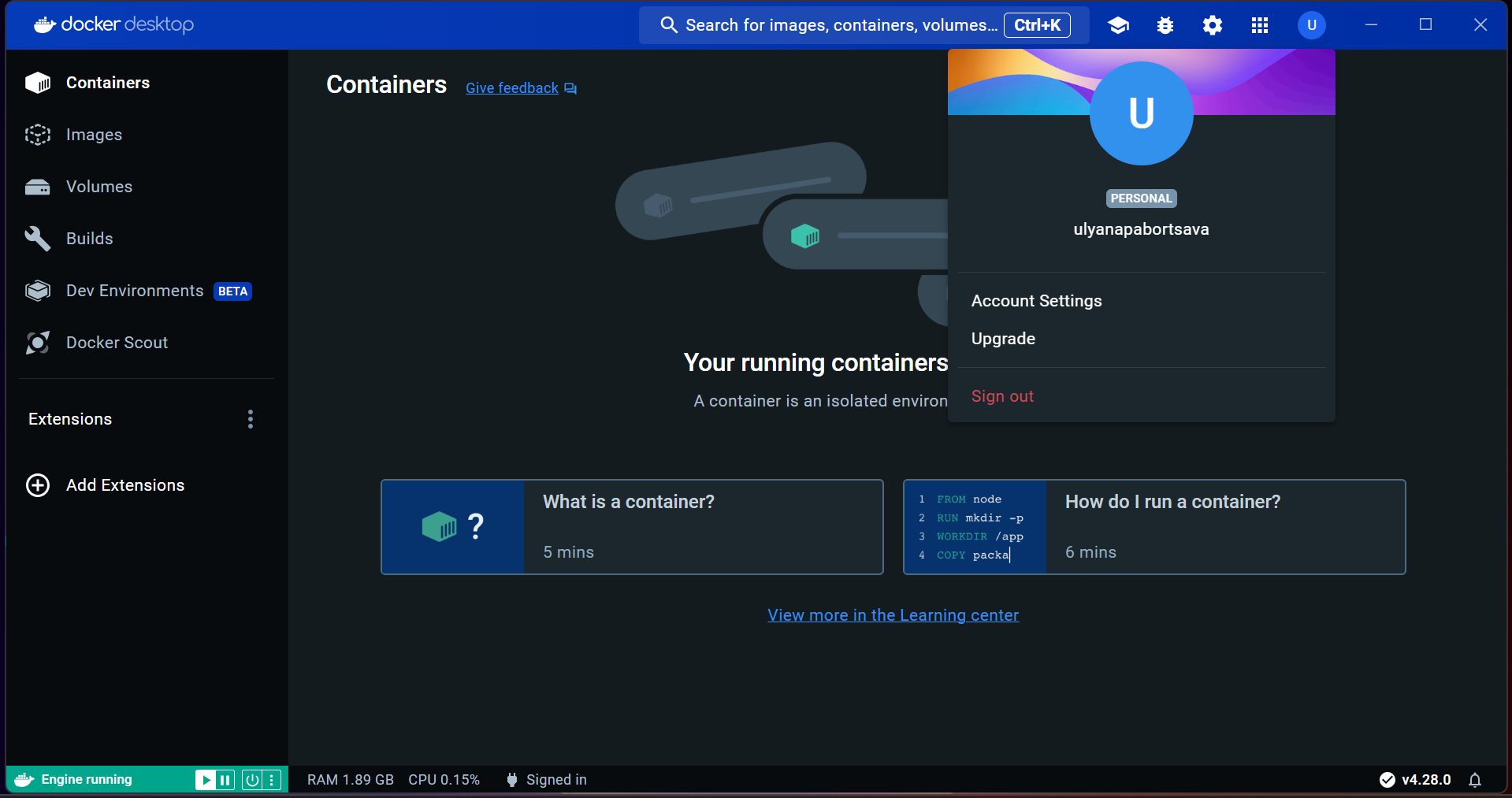
Лабораторная работа №2

**Тема**: Работа с Docker.

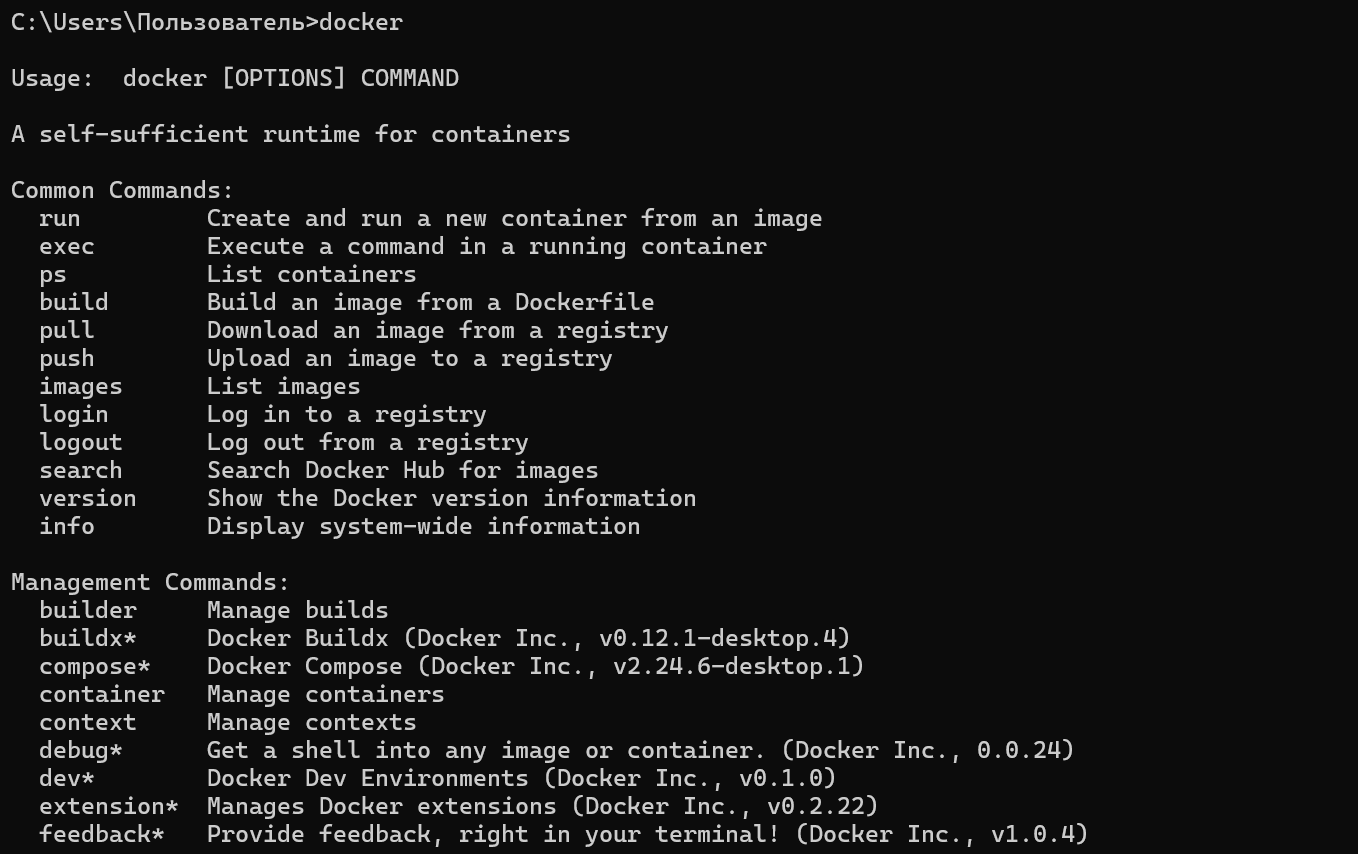
**Цель**: Познакомиться с возможностями и получить практические навыки работы с Docker.

1. Подготовьте рабочее окружение в соответствии с типом вашей операционной системы

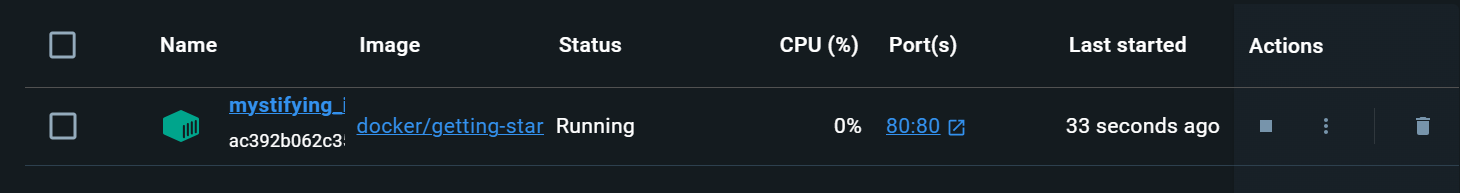
* Выполните базовую настройку

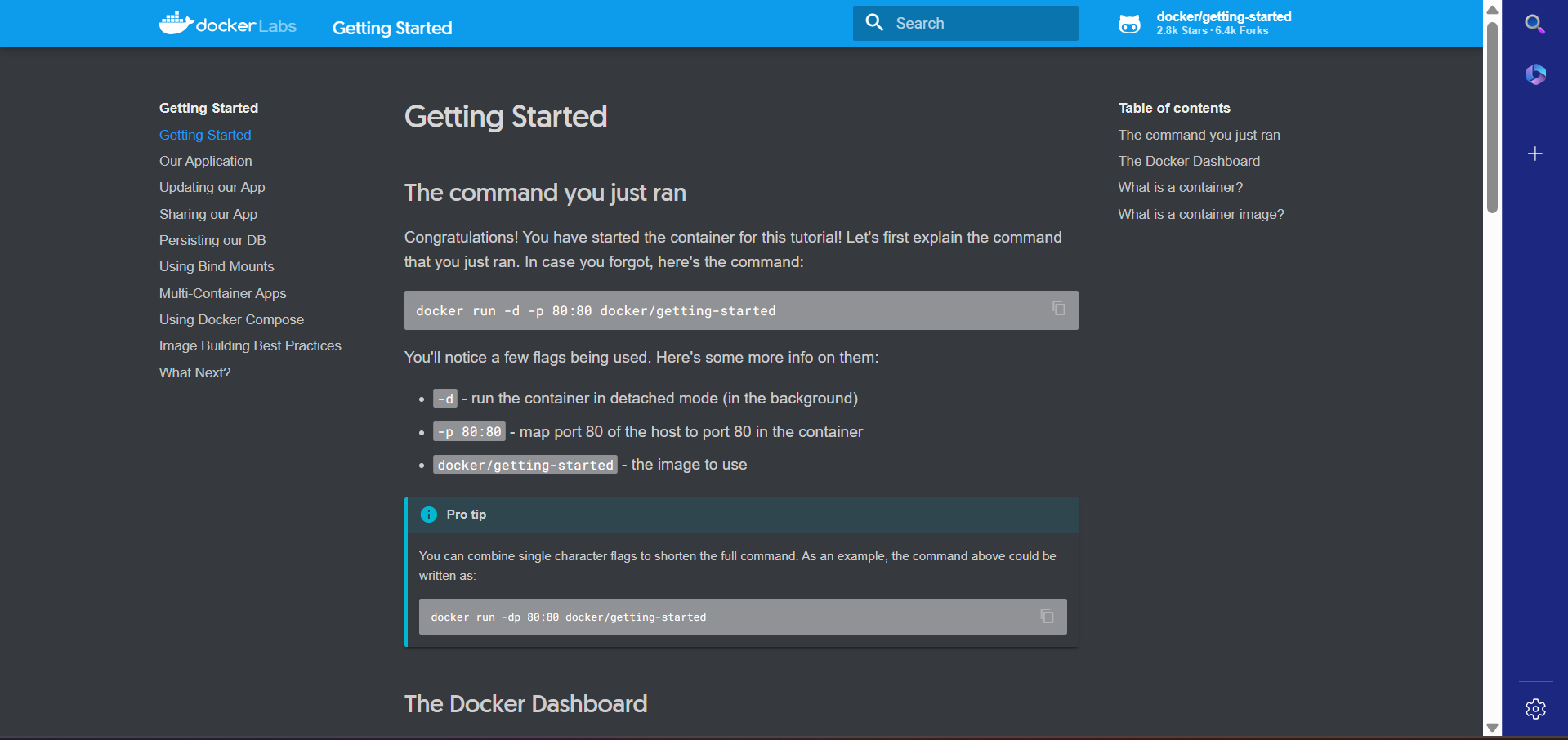


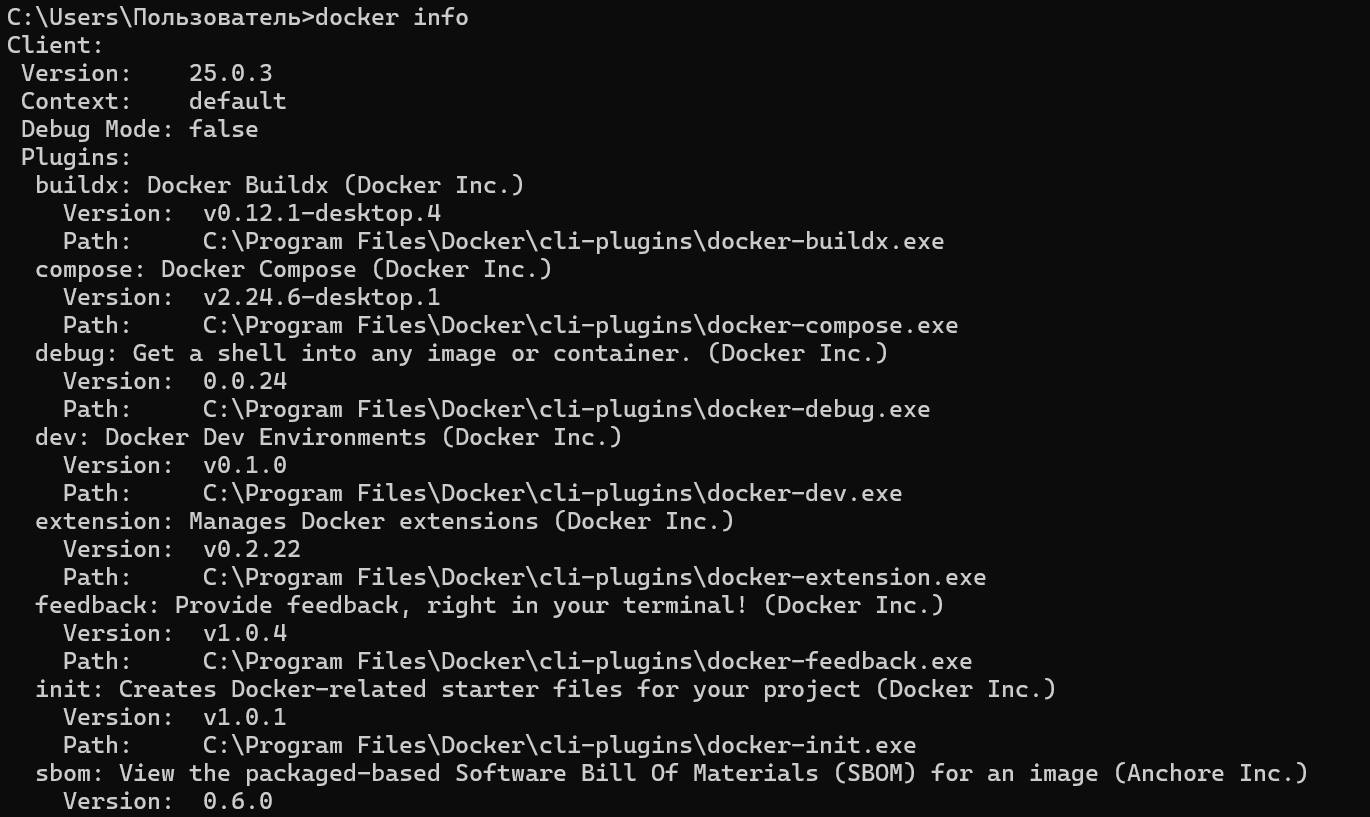
1. Изучите простейшие консольные команды и возможности Docker Desktop (см. лекцию), создать собственный контейнер docker/getting-started, открыть в браузере и изучить tutorial

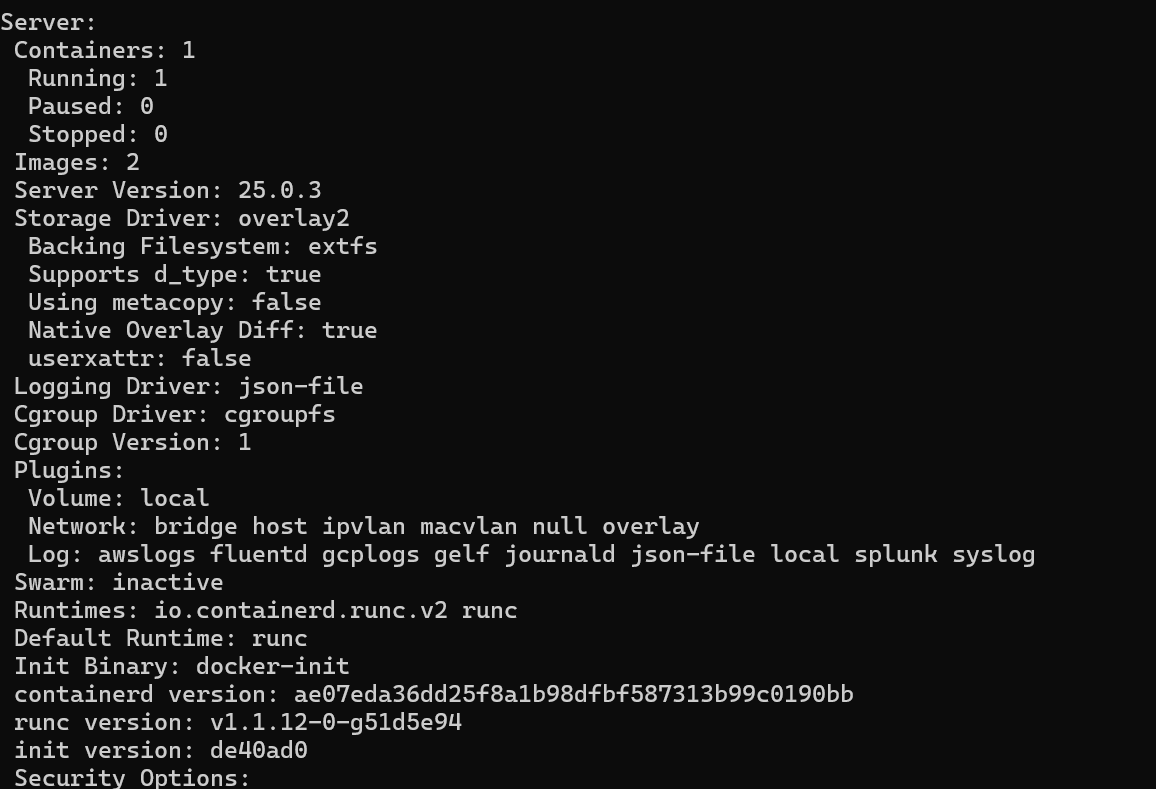


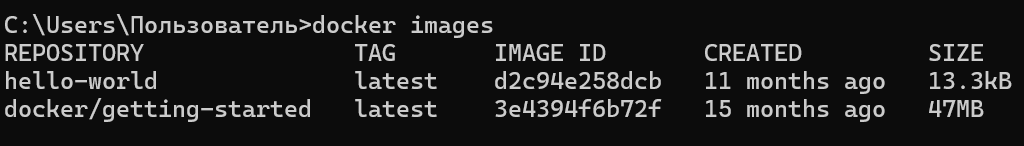


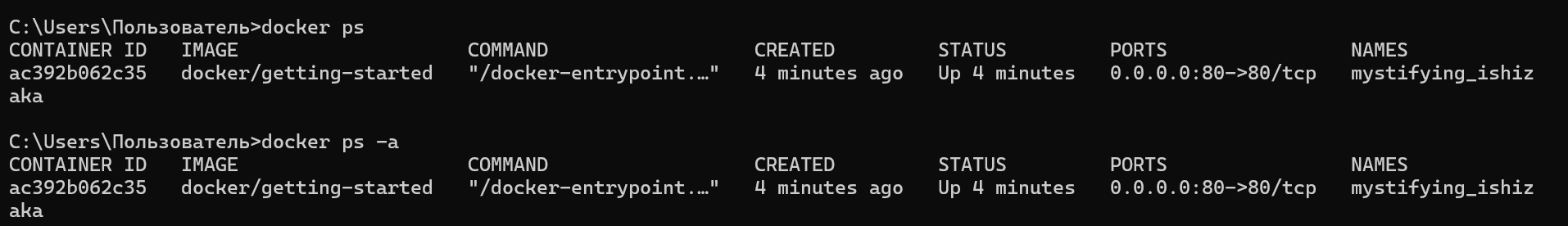


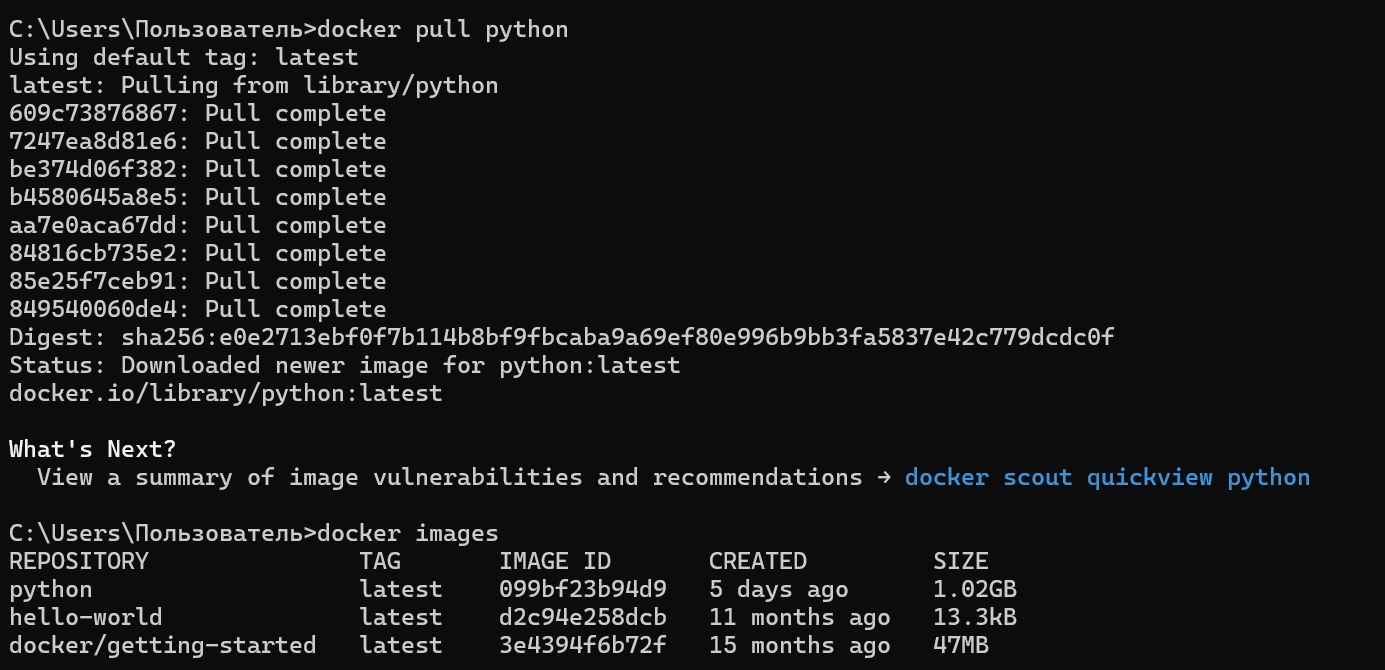


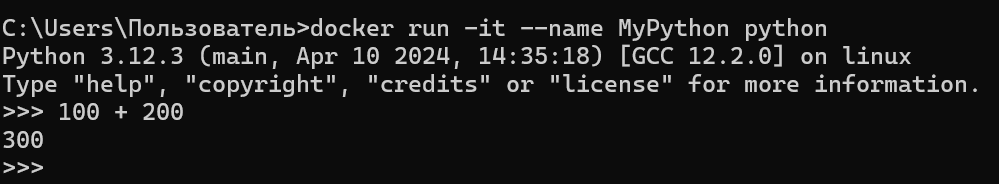


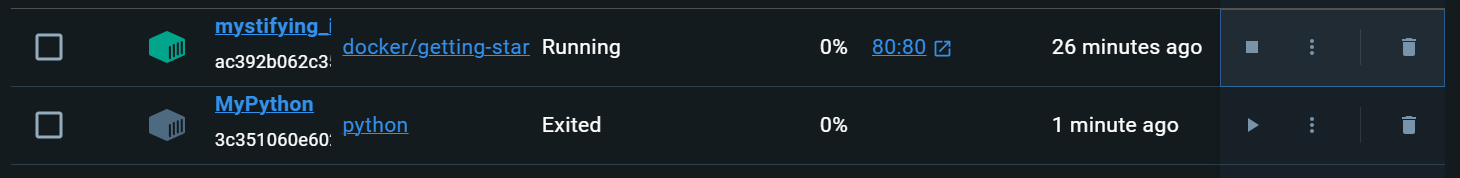




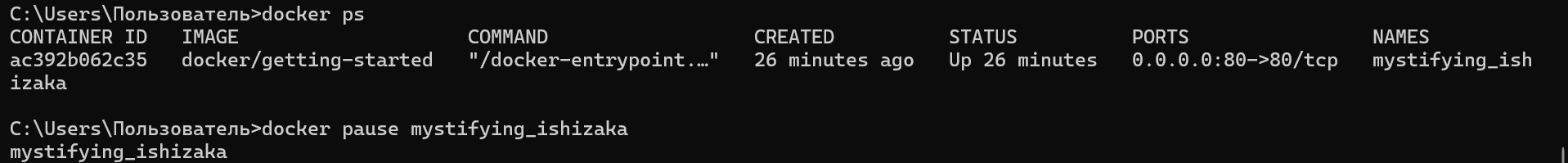


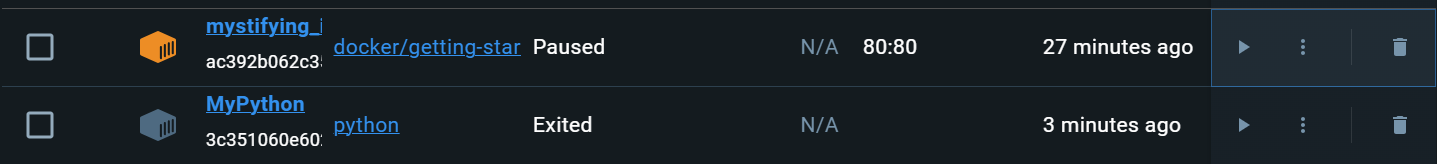


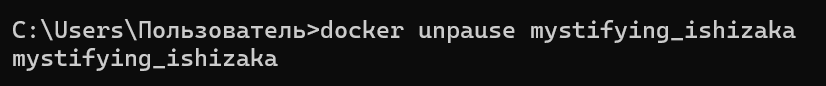


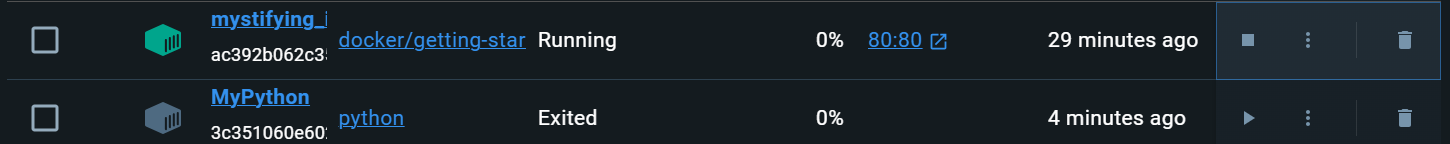


Docker pause/unpause

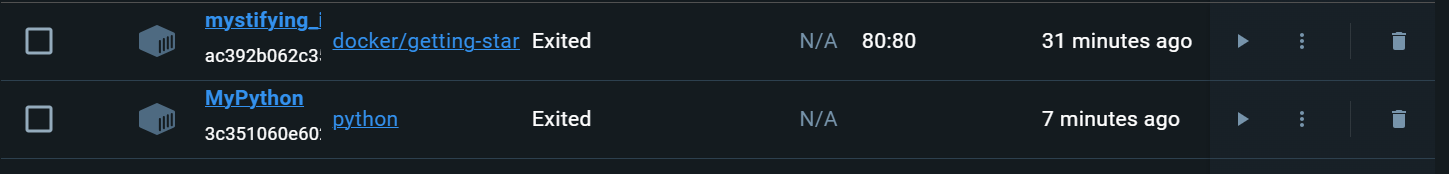


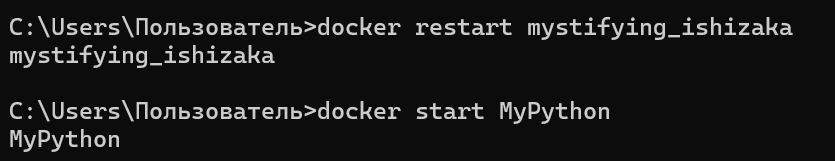


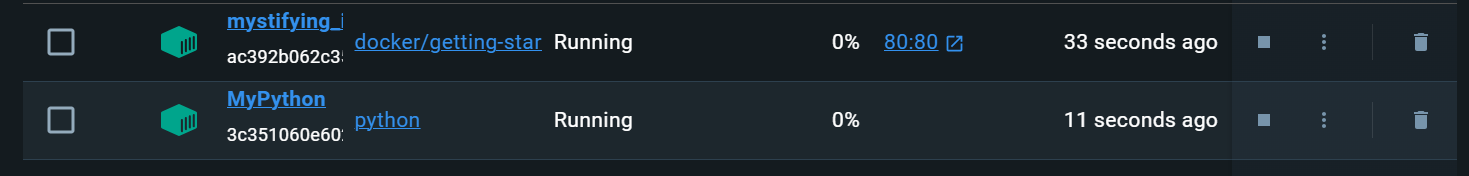




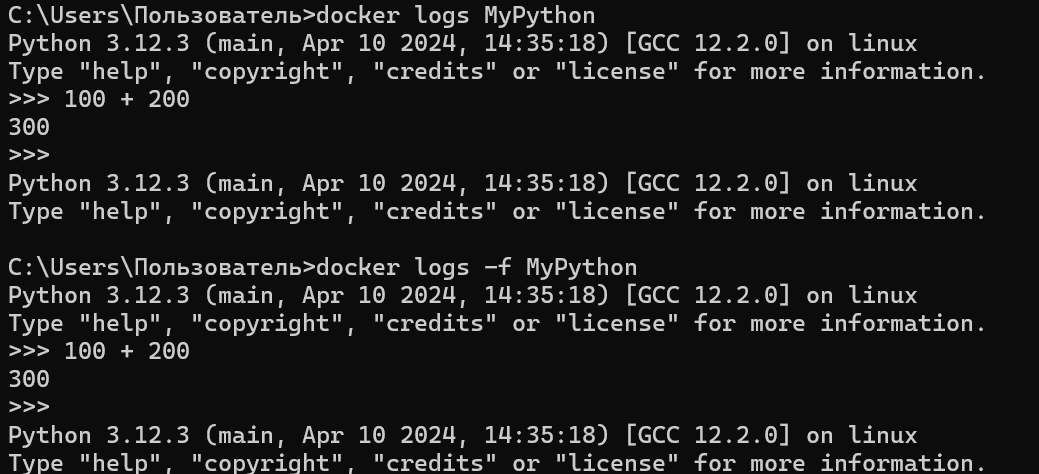
docker stop/start/restart

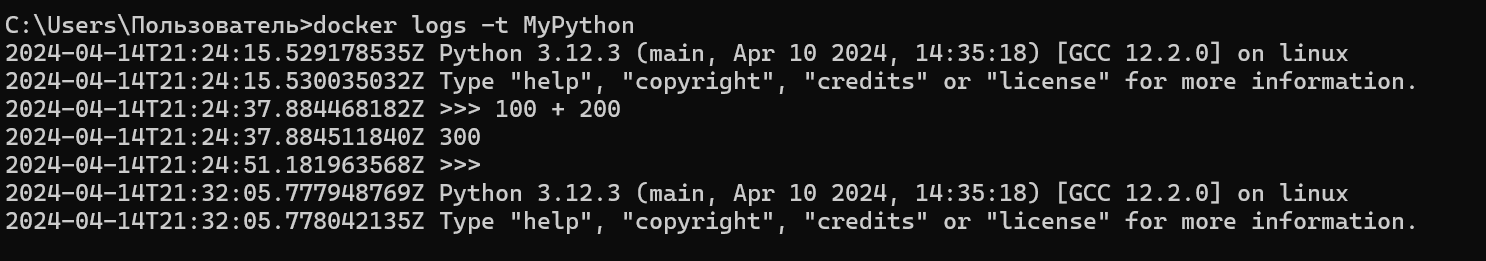


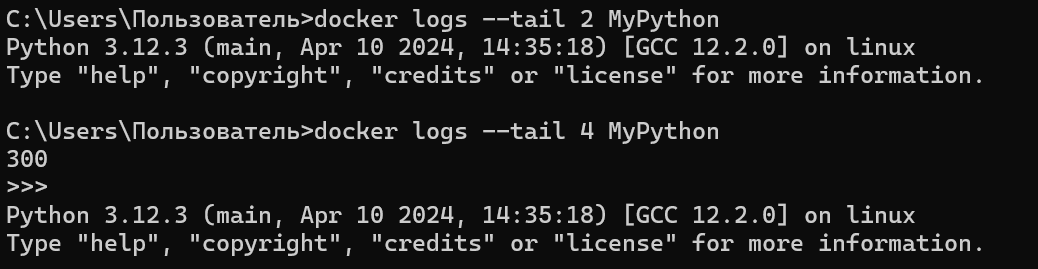




Docker logs

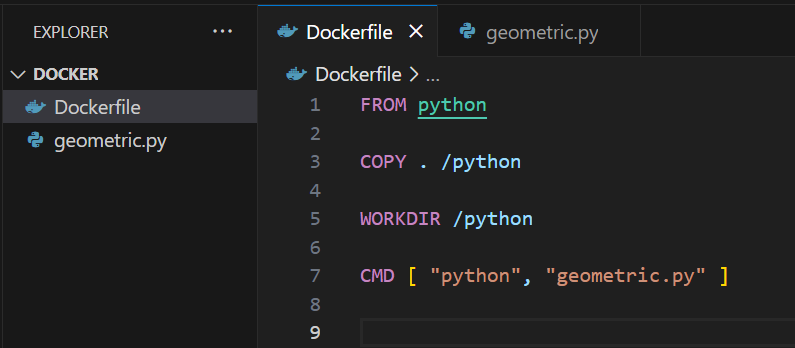


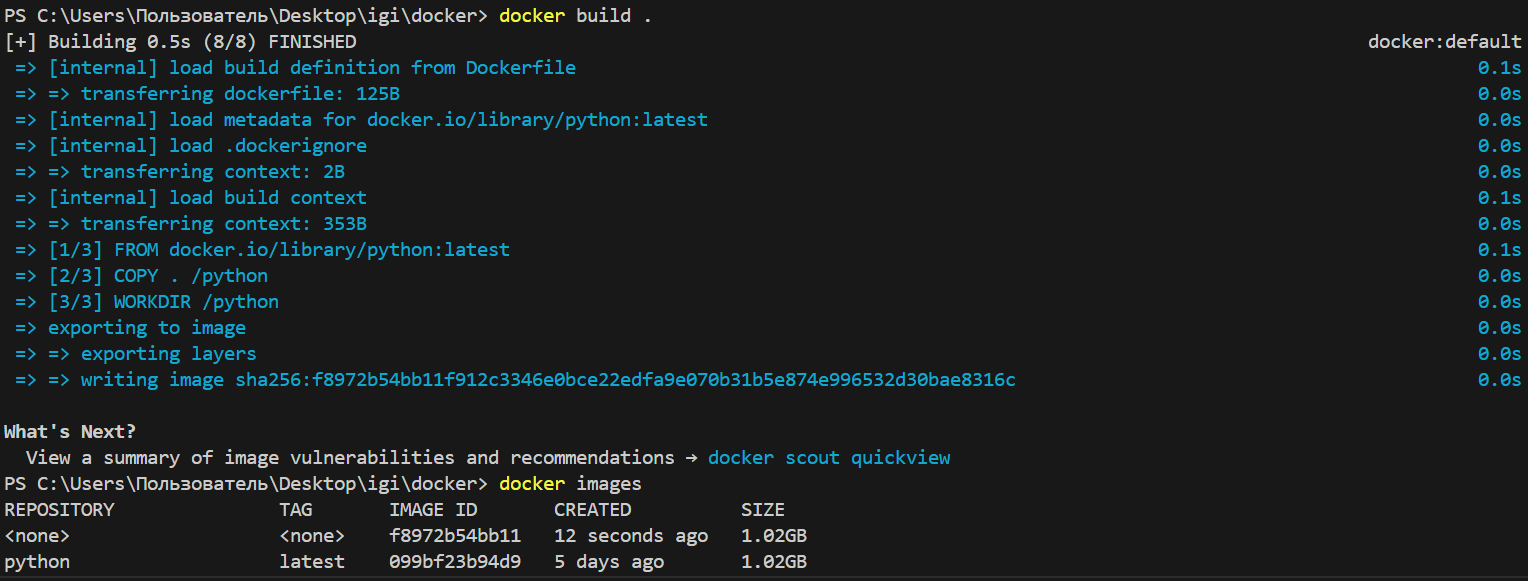


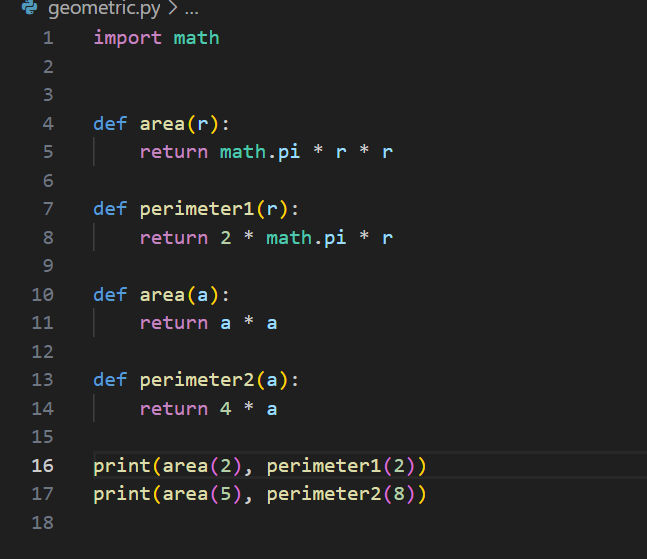


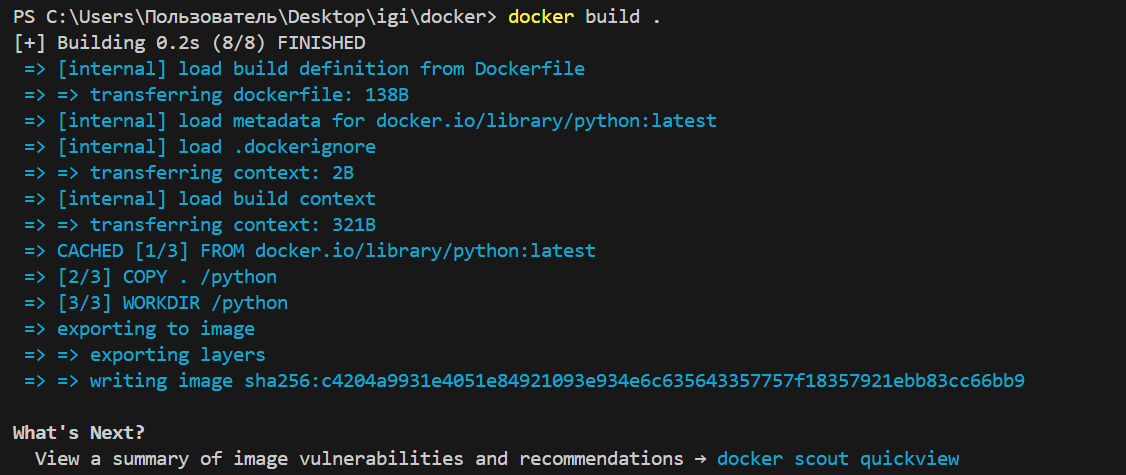
3. Создайте docker image, который запускает скрипт с использованием функций из <https://github.com/smartiqaorg/geometric_lib>.

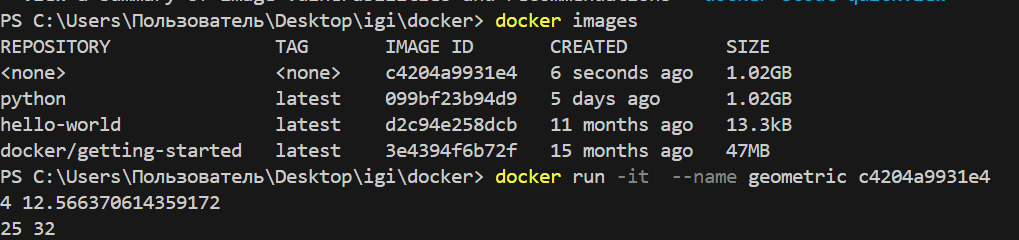
1. Способ передачи данных – переменные окружения.
2. Создать Dockerfile для реализации сборки собственных Docker образов
3. Использовать его для создания контейнера. Протестировать использование контейнера

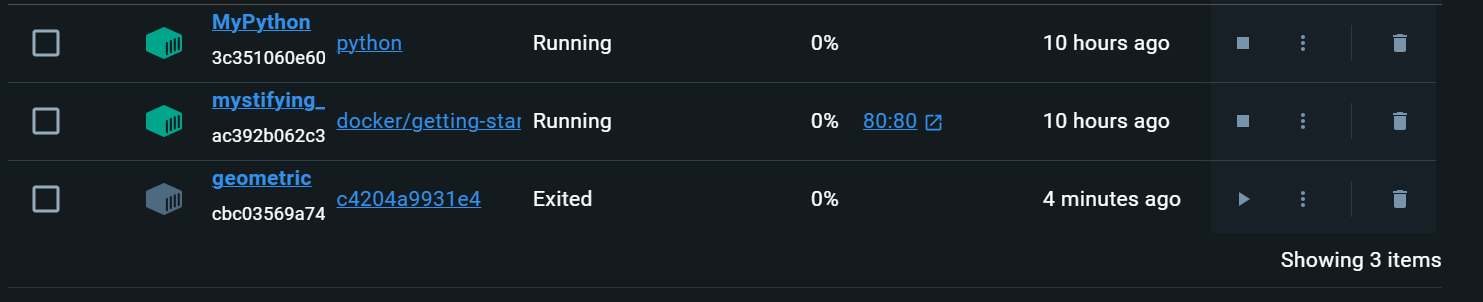










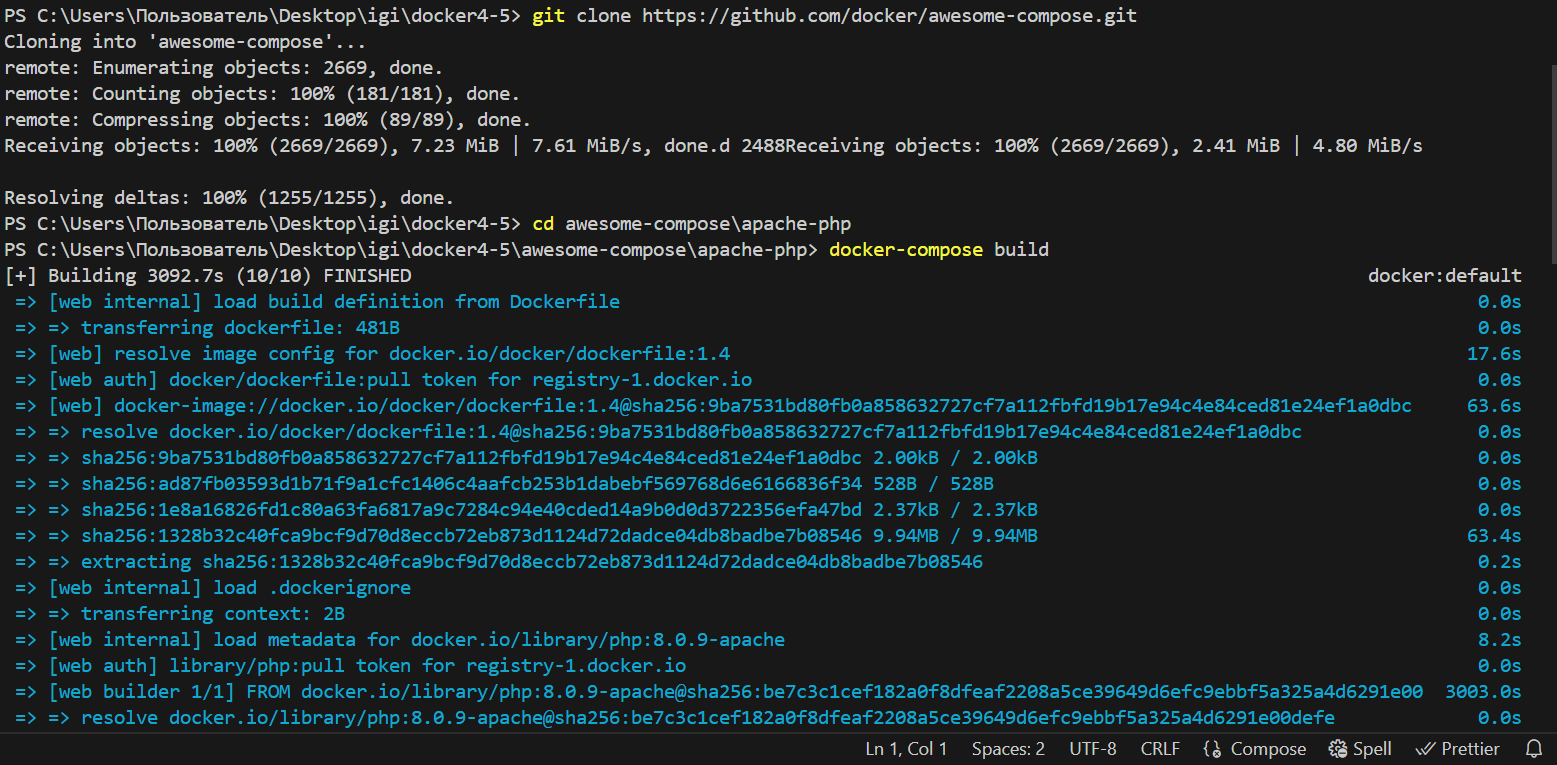


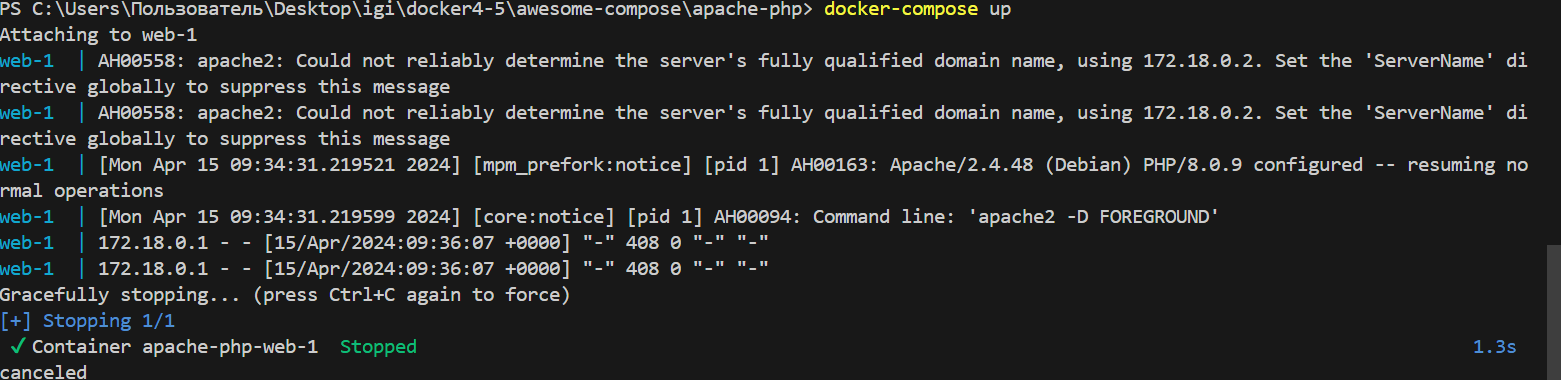
4-5. Скачать любой доступный проект с GitHub с произвольным стеком технологий или использовать свой, ранее разработанный. Создать для него необходимый контейнер, используя Docker Compose для управления многоконтейнерными приложениями. Запустить проект в контейнере. Настроить сети и тома для обеспечения связи между контейнерами и сохранения данных.

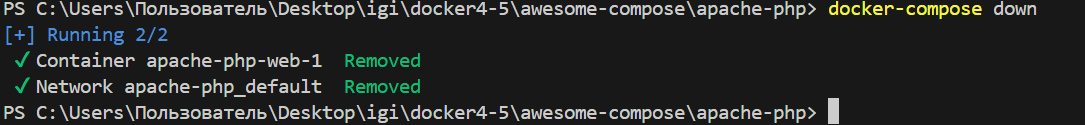
Ссылка на проект GitHub:

<https://github.com/docker/awesome-compose/tree/master>

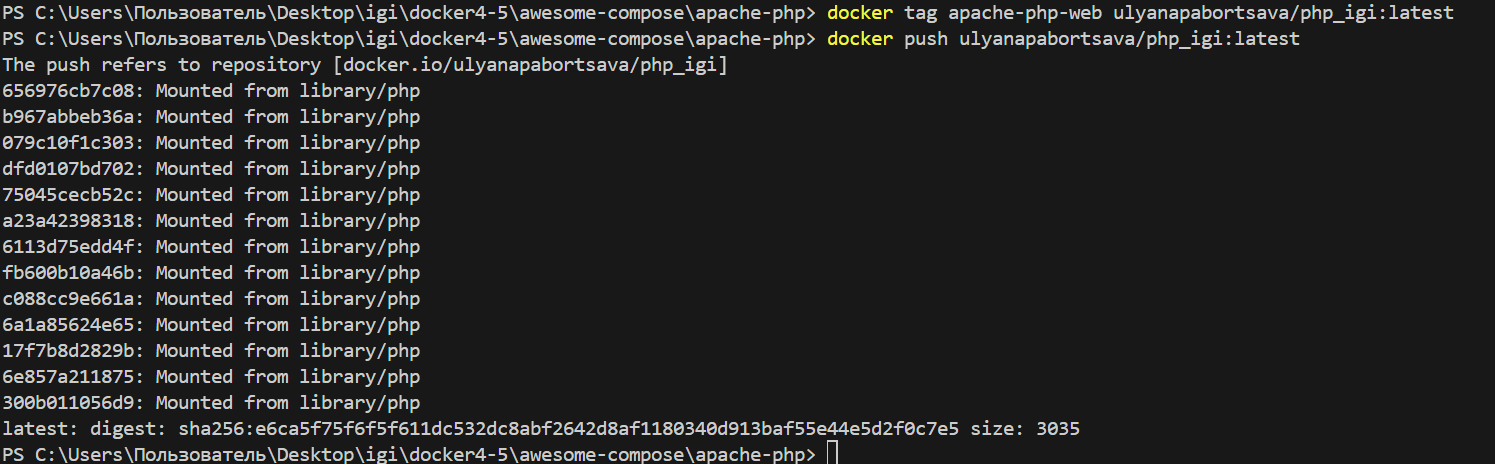
Скачан проект с GitHub со стеком: PHP. Создан контейнер , используя Docker Compose для управления многоконтейнерными приложениями. Настроены volumes, networks.

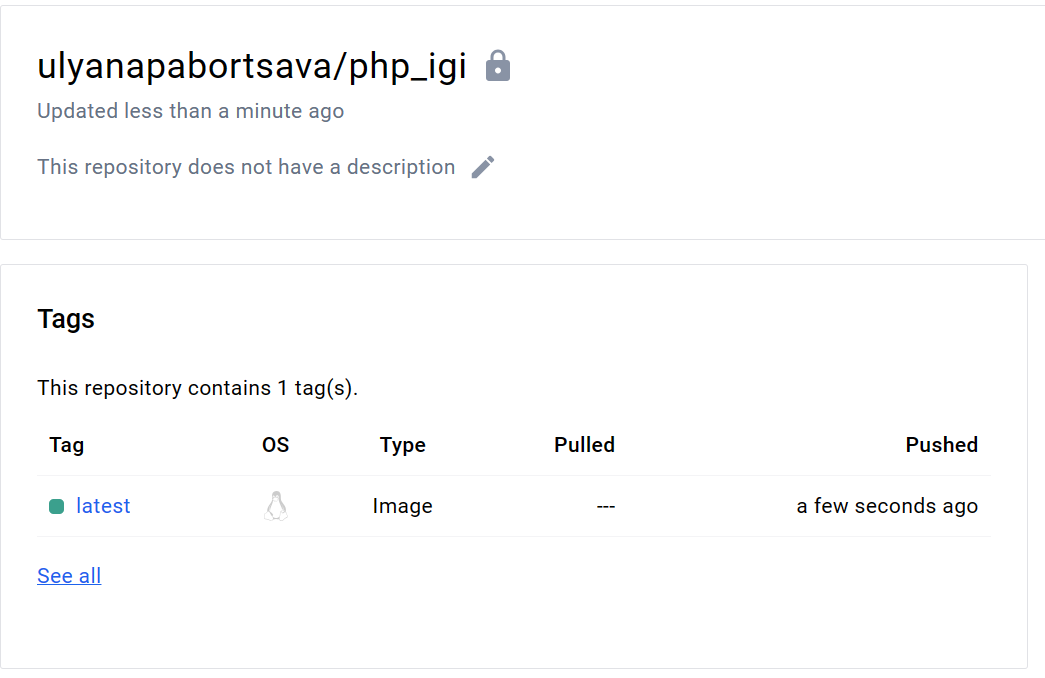






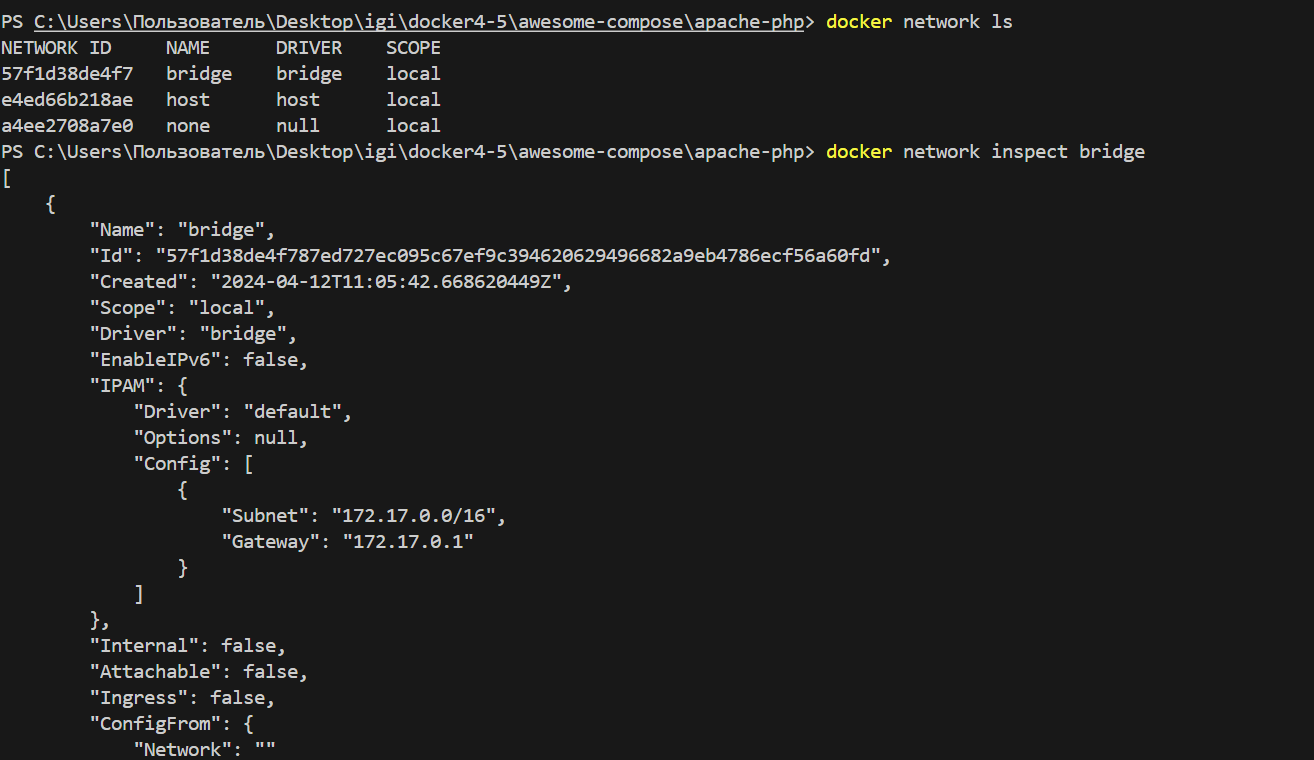
6. Разместите результат на DockerHub

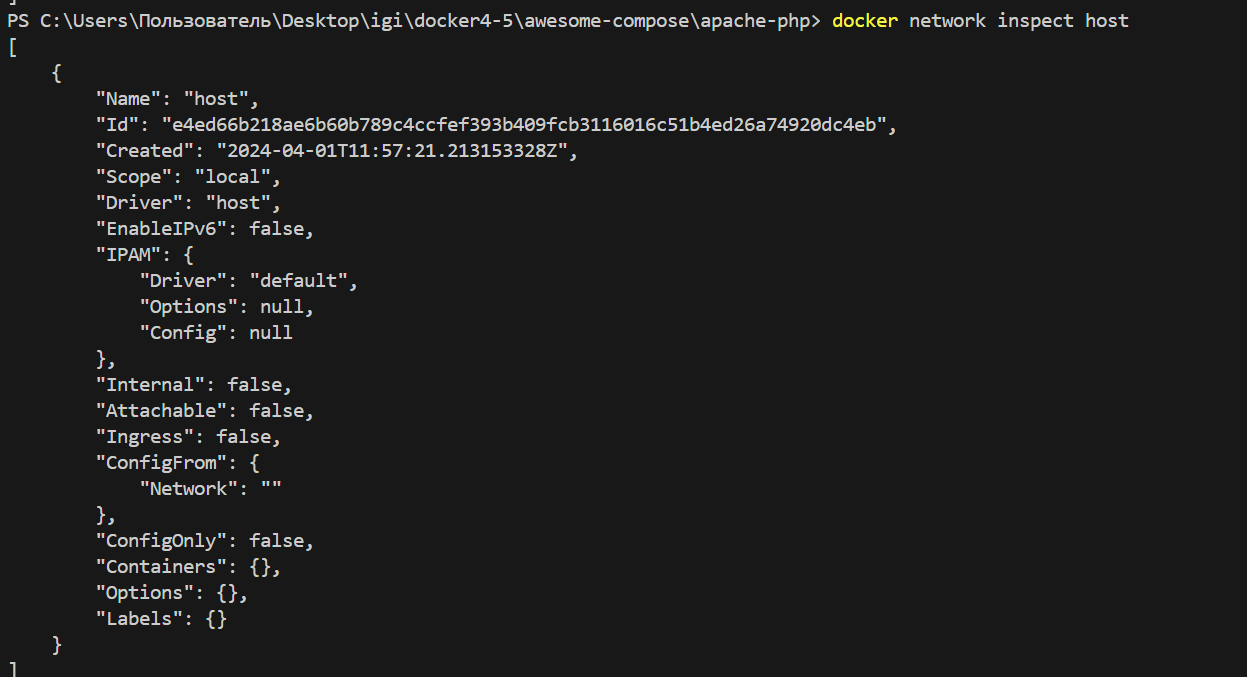




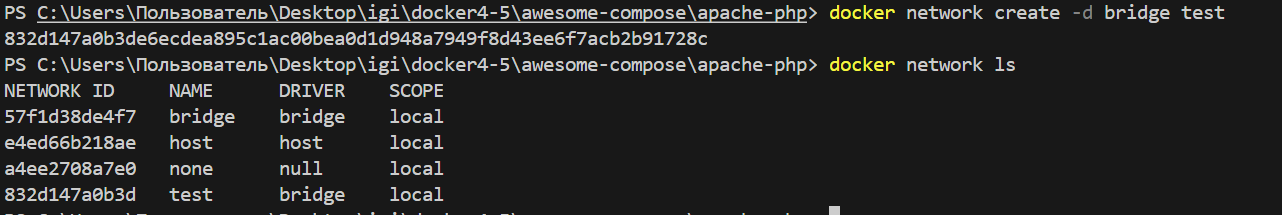
7.Выполните следующие действия с целью изучить особенности сетевого взаимодействия:

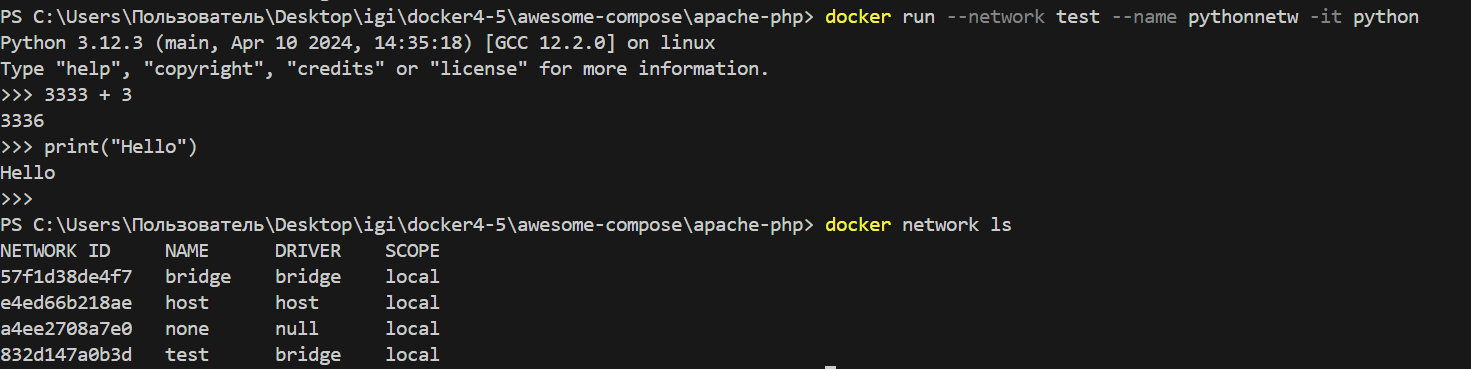
* Получить информацию о всех сетях, работающих на текущем хосте и подробности о каждом типе сети

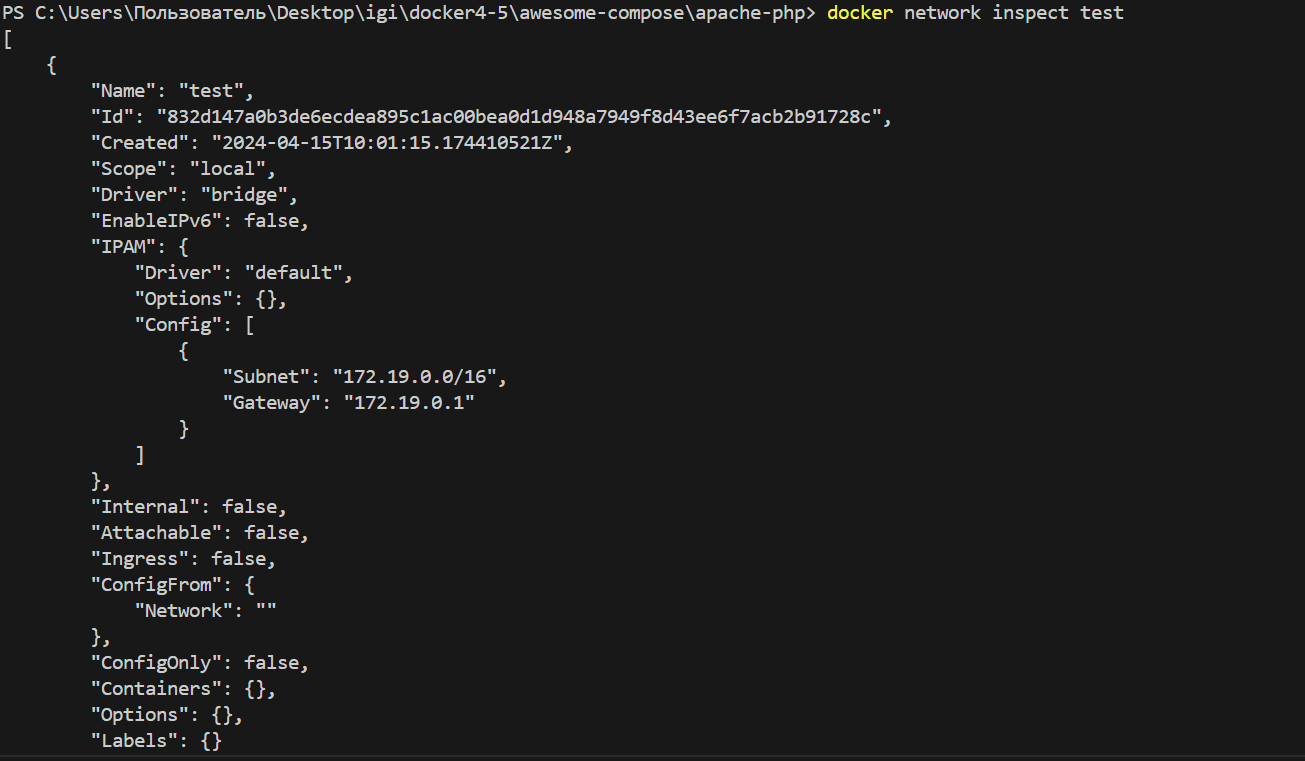


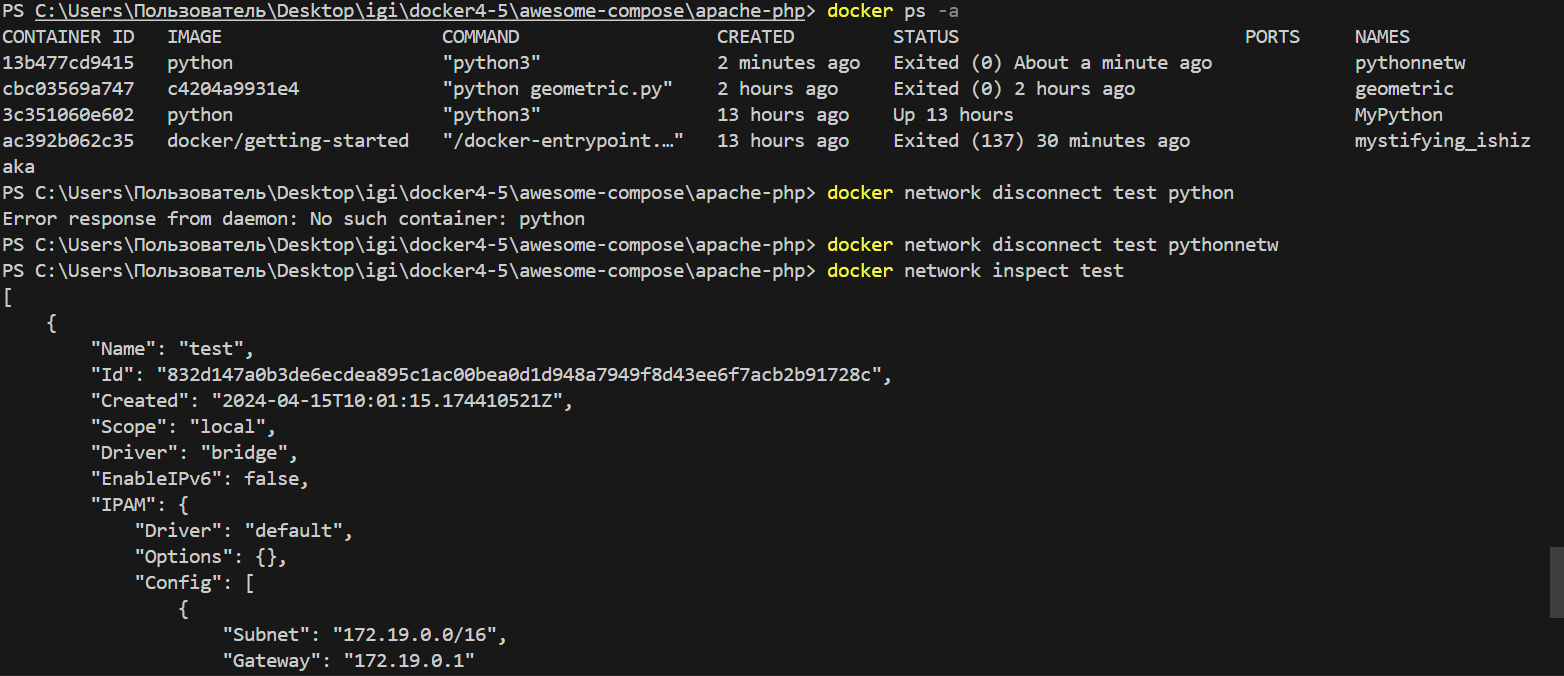


* Создать свою собственную сеть bridge, проверить, создана ли она, запустить Docker-контейнер в созданной сети, вывести о ней всю информацию(включая IP-адрес контейнера), отключить сеть от контейнера

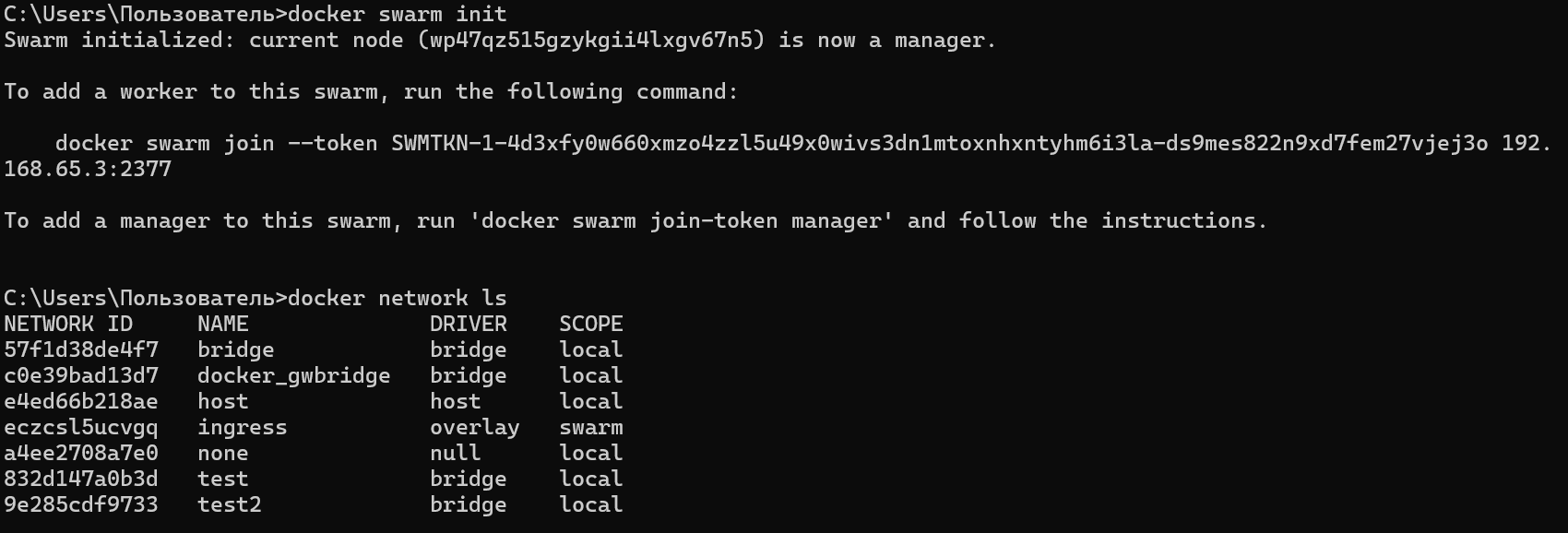


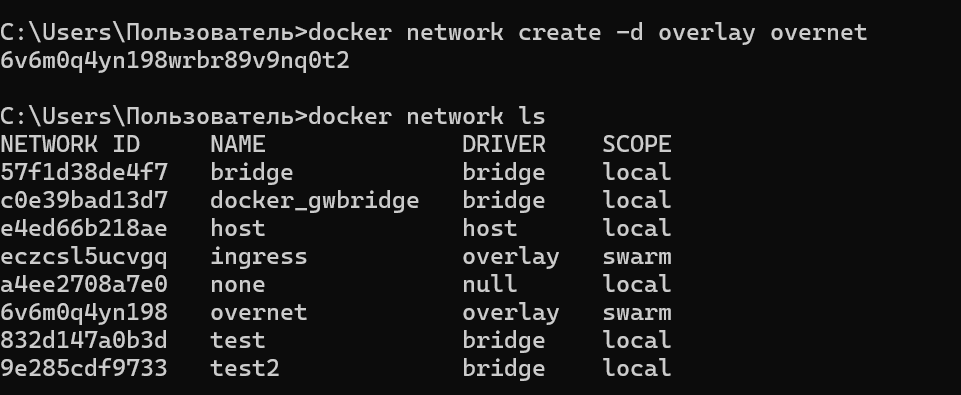


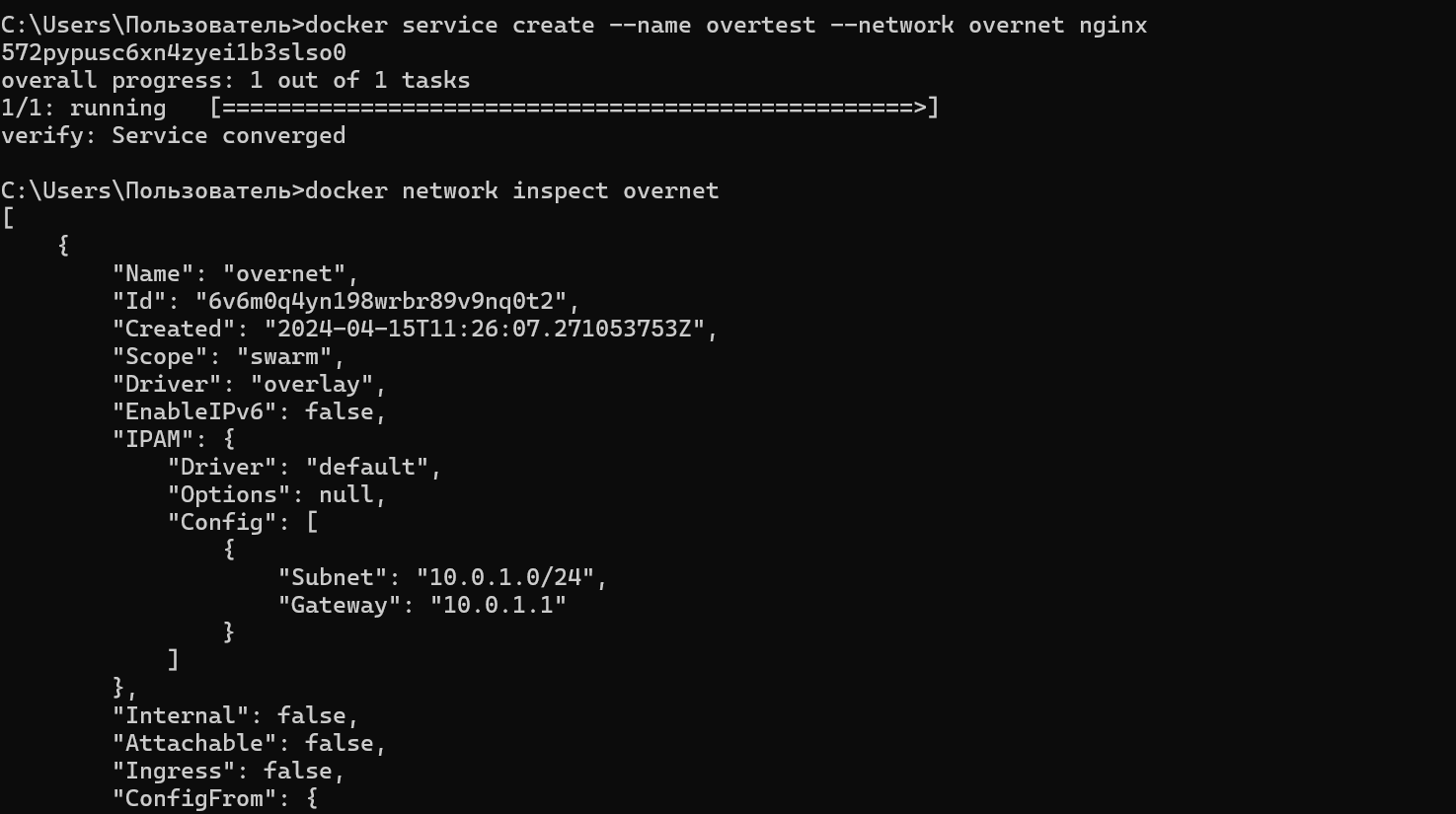


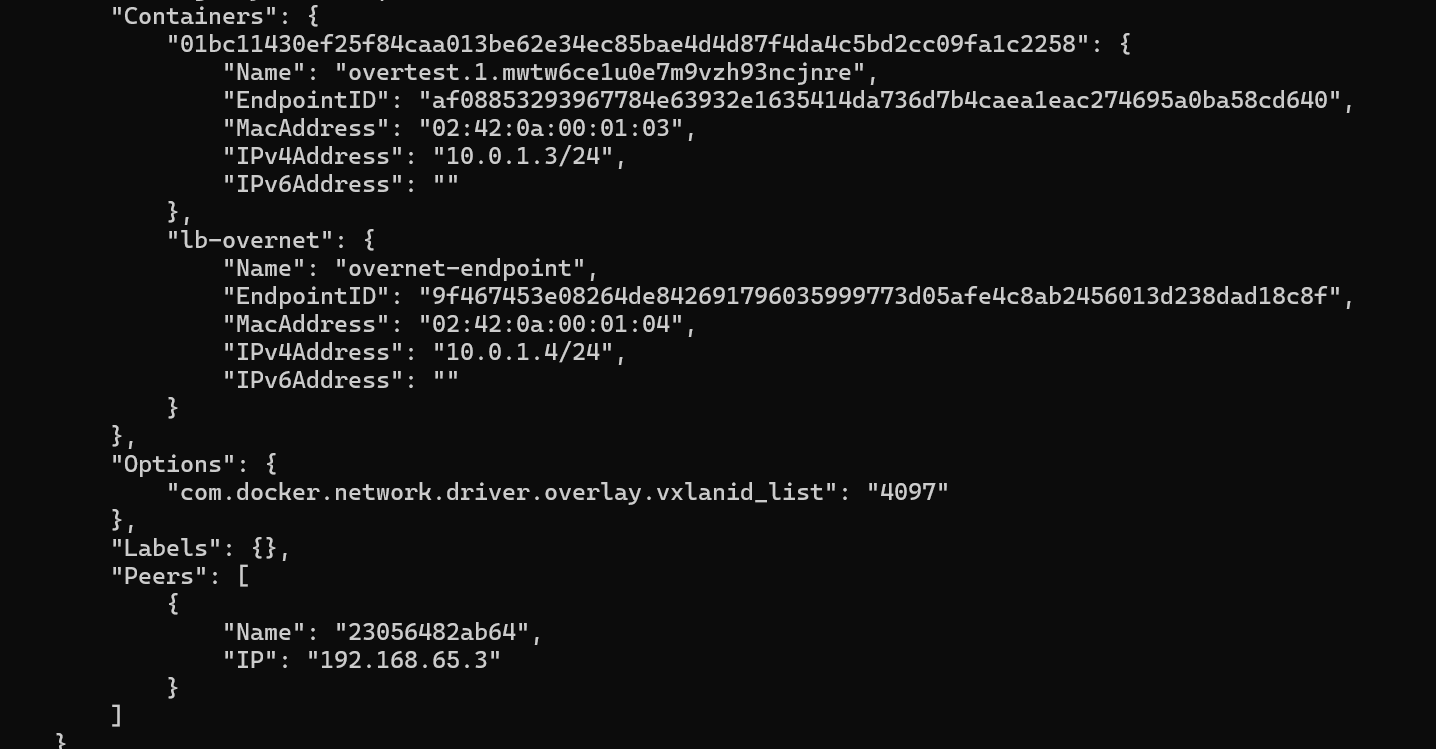


* Создать свою собственную сеть overlay, проверить, создана ли она, вывести о ней всю информацию

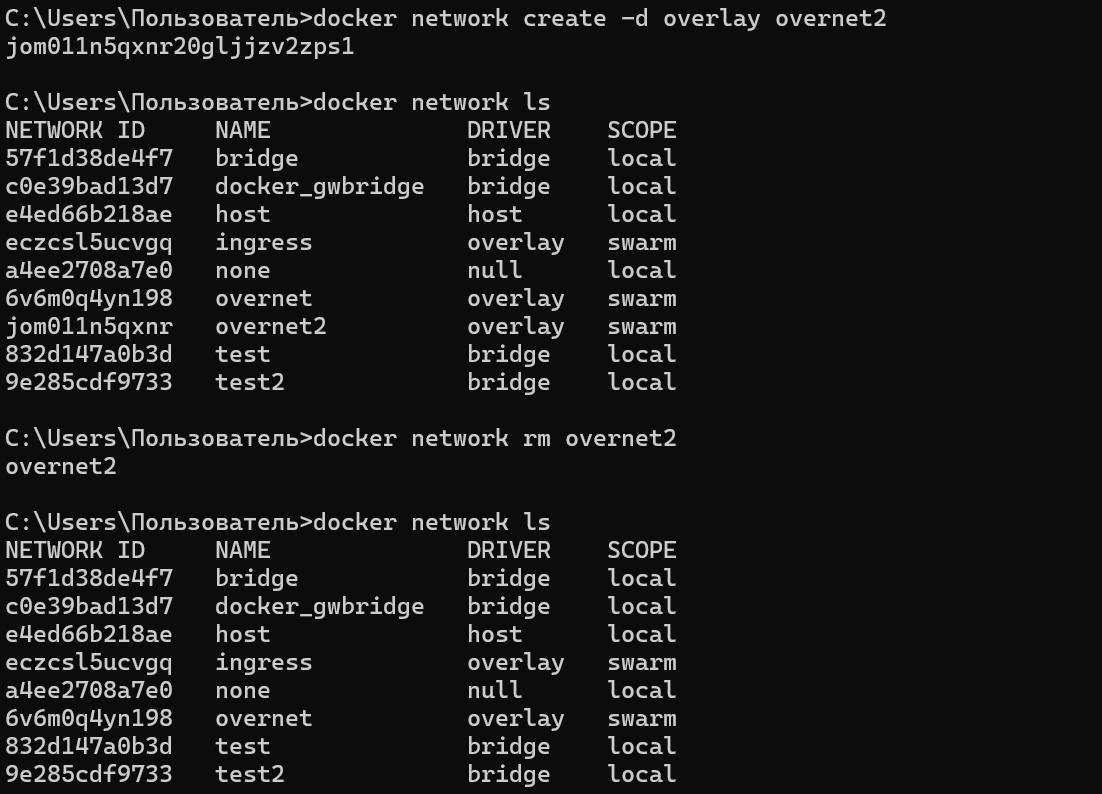








* Создать еще одну сеть overlay, проверить, создана ли она, вывести о ней всю информацию, удалить сеть



* Попробовать создать сеть host, сохранить результат  в отчет.

