**ЛАБОРАТОРНАЯ РАБОТА №7**

**DOCKER: ОСНОВЫ РАБОТЫ С КОНТЕЙНЕРАМИ**

**1. Установка Docker**

Установите Docker на вашу систему.

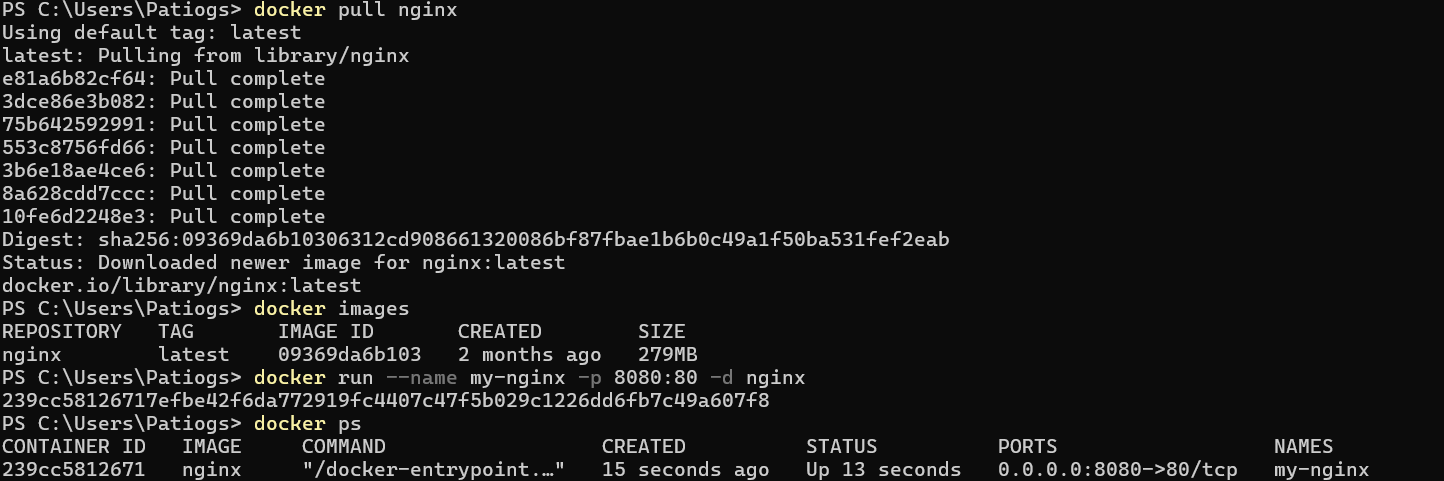
Проверьте корректность установки командой docker --version.

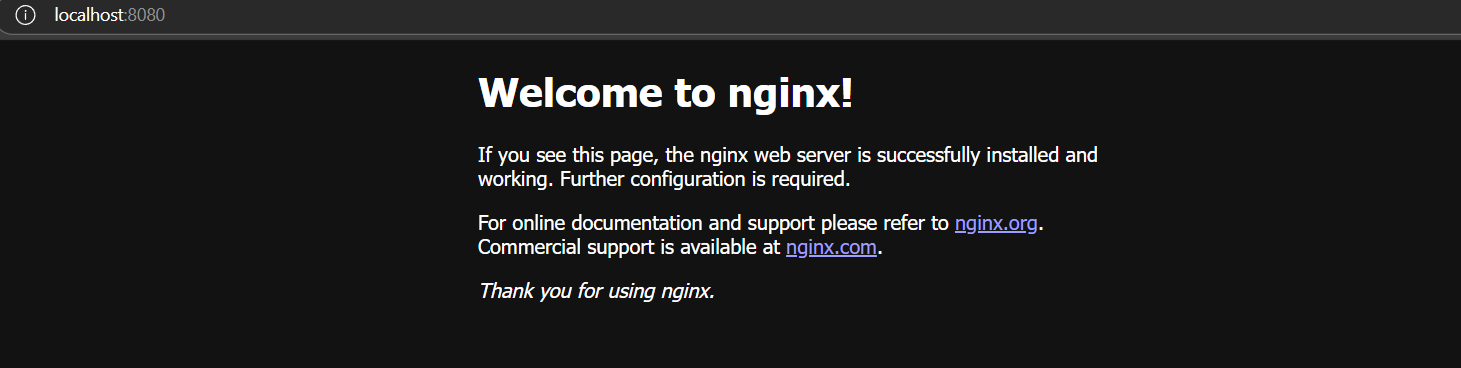


**2. Работа с образами**

Загрузите образ nginx из Docker Hub.

Запустите контейнер на основе этого образа и откройте веб-страницу Nginx в браузере.

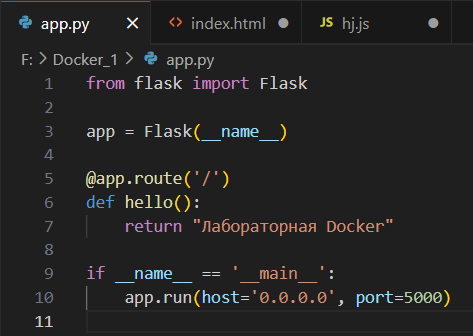
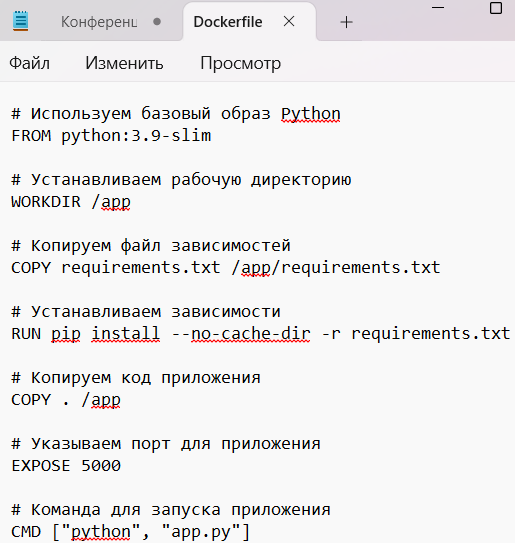
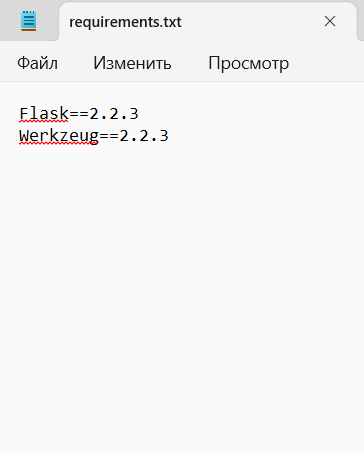


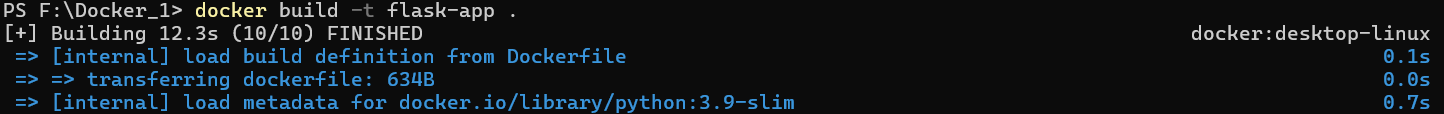


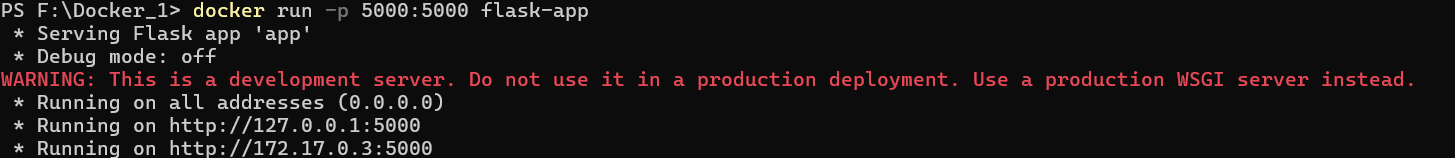
**3. Создание пользовательского образа**

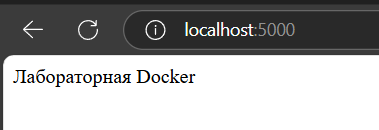
Напишите Dockerfile для создания образа с простым веб-приложением на Python (например, Flask).

Соберите образ и запустите контейнер.

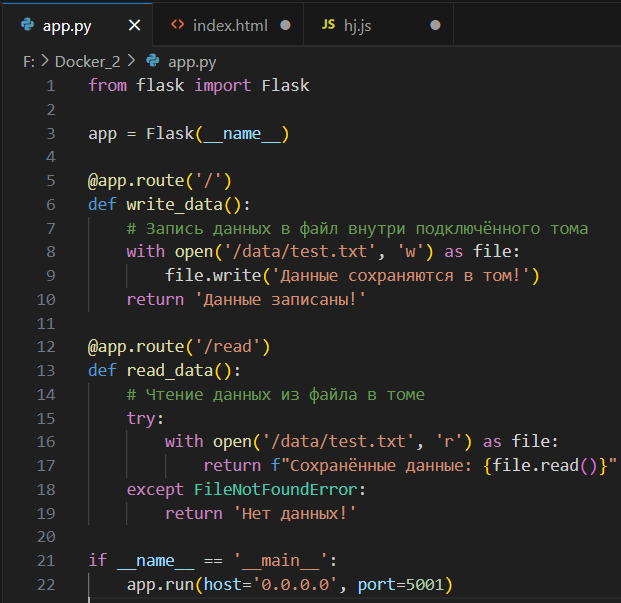
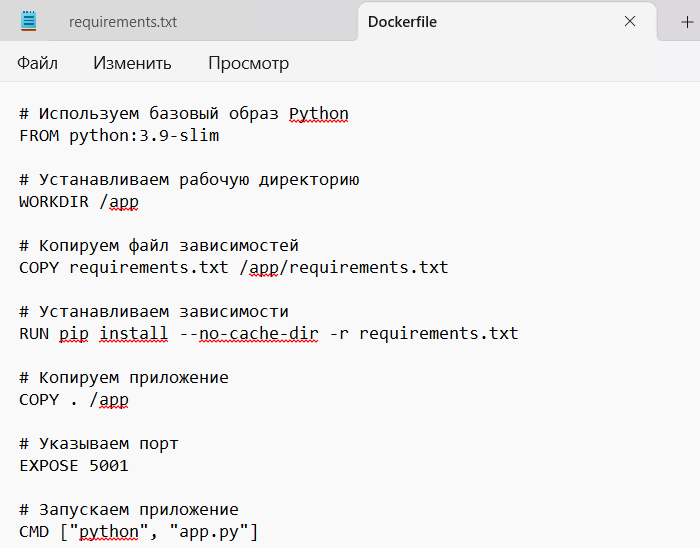




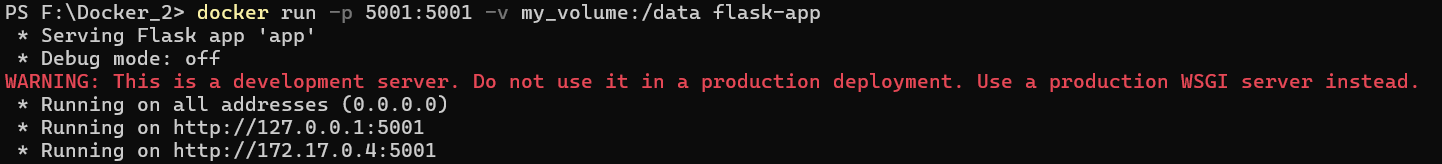


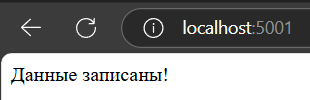
**4. Работа с томами (Volumes)**

Создайте том и подключите его к контейнеру для сохранения данных.

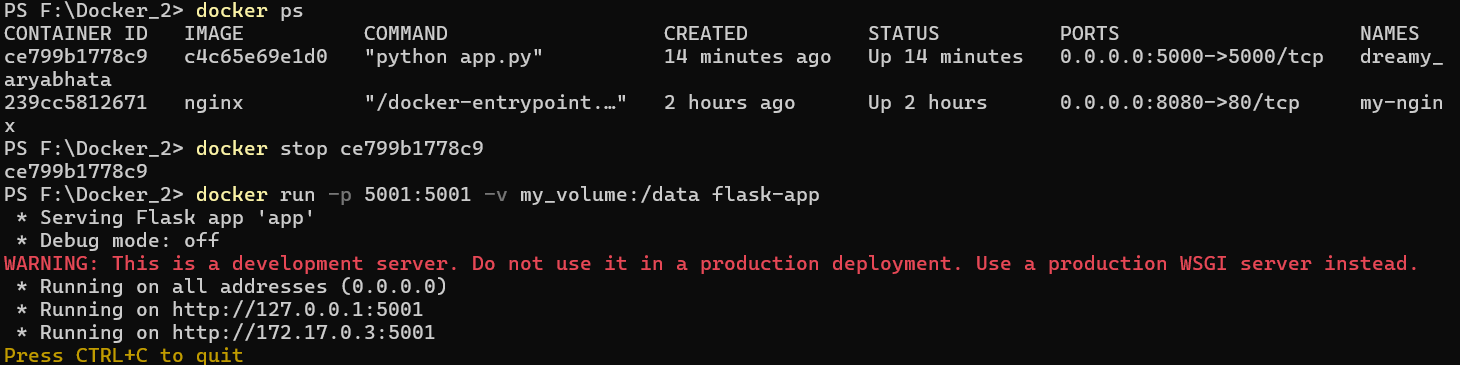
 

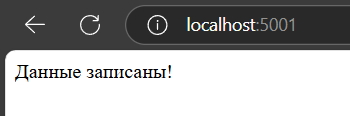






Убедитесь, что данные сохраняются после перезапуска контейнера.

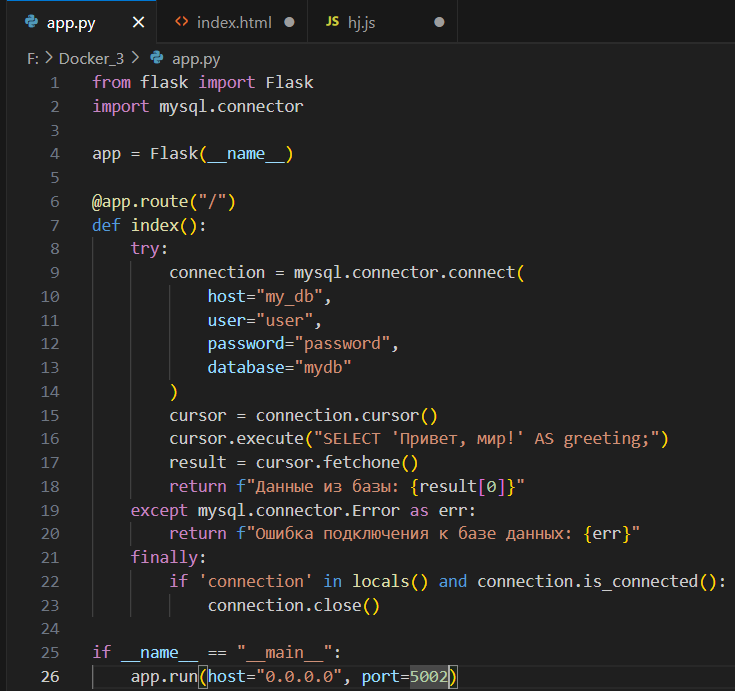
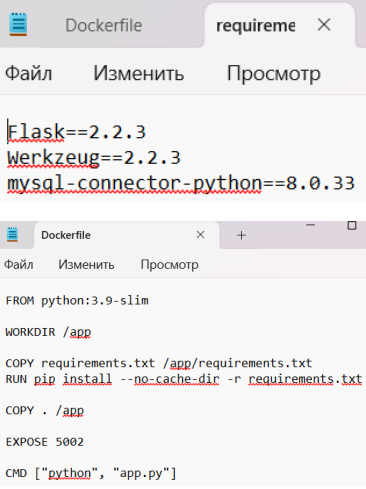


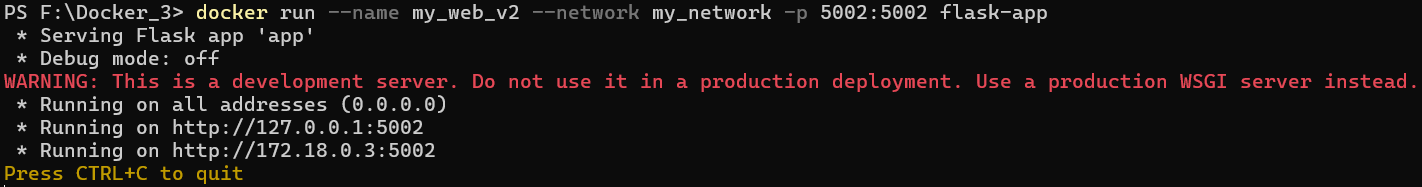
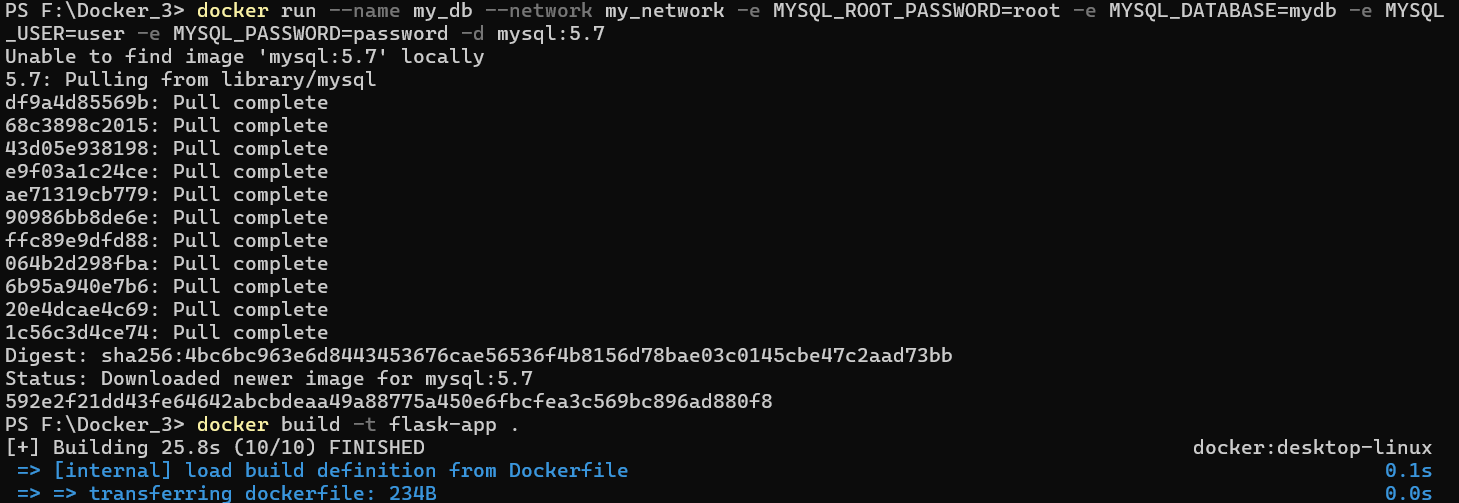


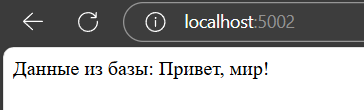
**5. Сеть в Docker**

Создайте два контейнера (например, веб-сервер и базу данных).

Настройте сеть для взаимодействия между ними.

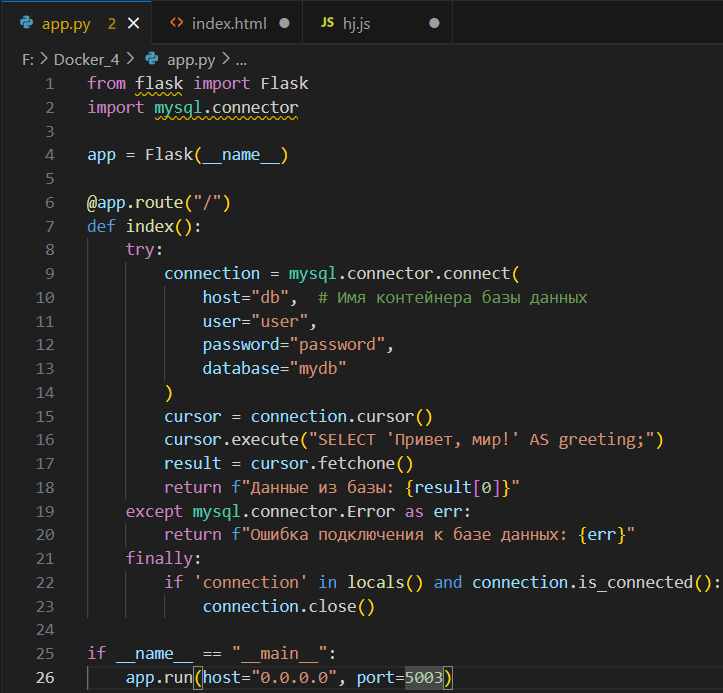
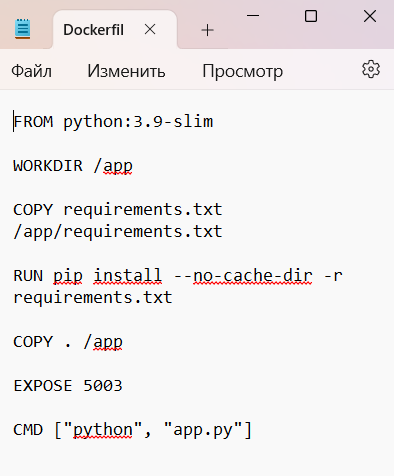


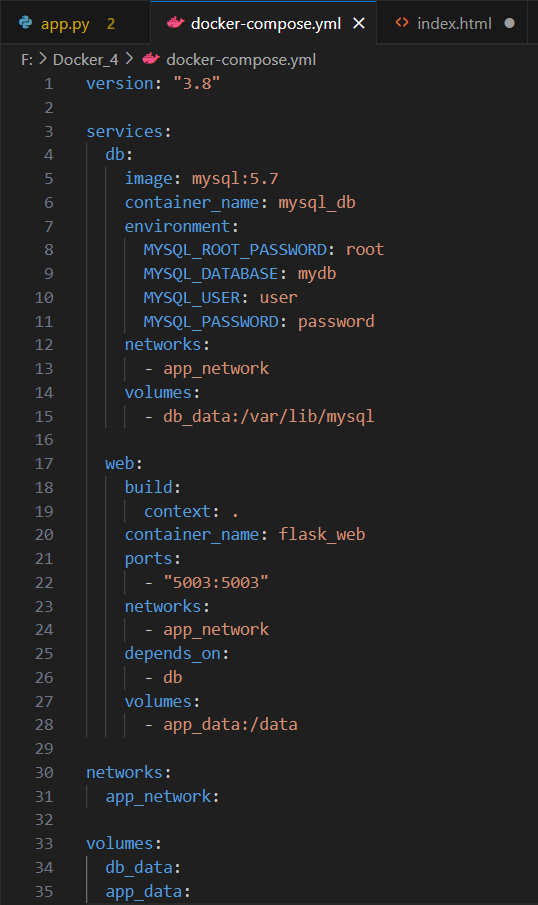


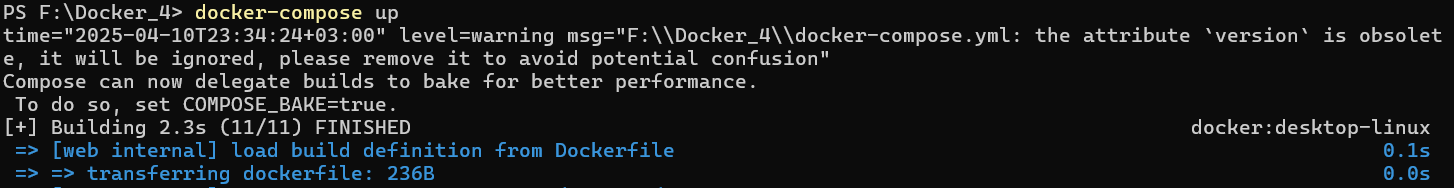
**6. Docker Compose**

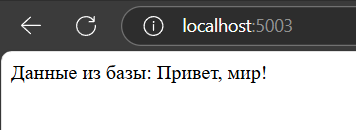
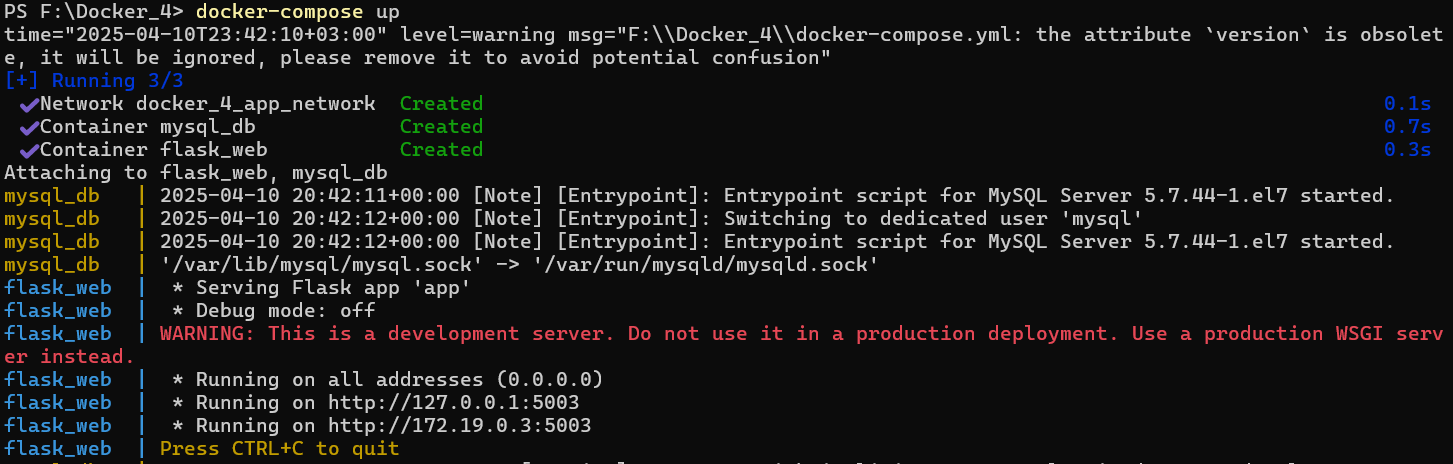
Напишите файл docker-compose.yml для развёртывания многоконтейнерного приложения (например, веб-сервер + БД).

Запустите приложение с помощью Docker Compose.







**7. Очистка ресурсов**

Удалите все созданные контейнеры, образы и тома.

