Департамент образования и науки города Москвы

Государственное автономное образовательное учреждение высшего образования города Москвы

«Московский городской педагогический университет»

Институт цифрового образования

Департамент информатики, управления и технологий

ДИСЦИПЛИНА:

Интеграция и развертывание программного обеспечения с помощью контейнеров  
**Лабораторная работа №3.1**

**Тема:**

«Docker Compose для мультиконтейнерных приложений»

Выполнил(а): st\_98, группа: АДЭУ-211

Преподаватель:

Москва

2025

**Цель работы:** освоить использование Docker Compose для управления многоконтейнерными приложениями

**Задачи:**

1. Создать файл docker-compose.yml для указанного многоконтейнерного приложения.
2. Запустить приложение с помощью Docker Compose.
3. Проверить работоспособность приложения и взаимодействие между контейнерами.
4. Выполнить индивидуальное задание

**Вариант 11 (st\_98):**

1. Создать файл docker-compose.yml для системы анализа социальных сетей (Python Flask + Neo4j).
2. Запустить приложение и проверить графовые запросы.
3. Добавить анализ популярности хештегов.

**Ход работы:**

1. В первую очередь на рисунке 1 посмотрим дерево файлов (каталог файлов)

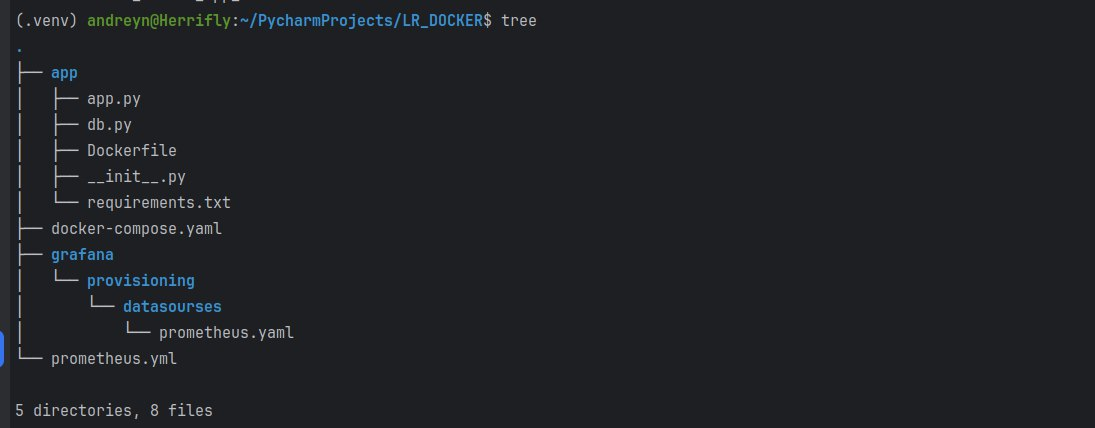


Рисунок 1 – Каталог файлов

1. Запускаем Docker compose (рисунок 2)

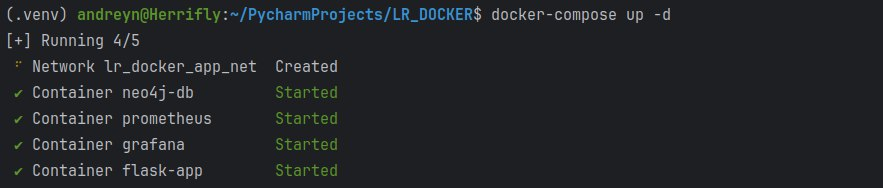


Рисунок 2 – Запуск Docker compose

1. Проверяем, что все необходимые контейнеры запущены (рисунок 3)



Рисунок 3 – Запущенные контейнеры

1. Напишем код, который сгенерирует нам посты с хештэгами и лайками (имитация поста в социальной сети). Код представлен на рисунке 4.



Рисунок 4 – Генеративная функция

1. Выполним подключение к Prometheus через Grafana, используя его ip из inspect контейнера. Процесс подключения отображен на рисунках 5-6.

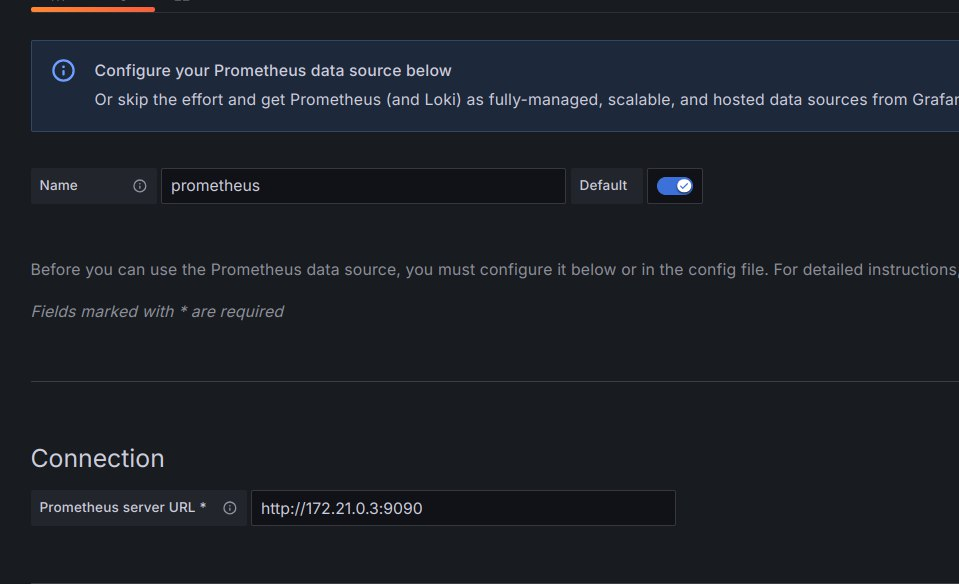


Рисунок 5 – url для подключения

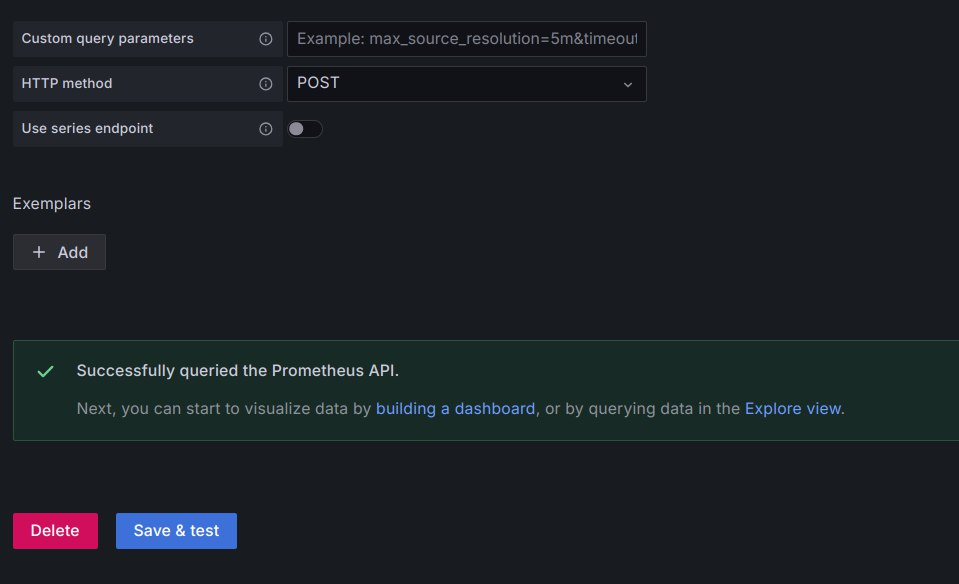


Рисунок 6 – Успешный результат подключения

1. Проверим зависимость лайков постов от хэштегов с помощью дашборда Grafana на рисунке 7.

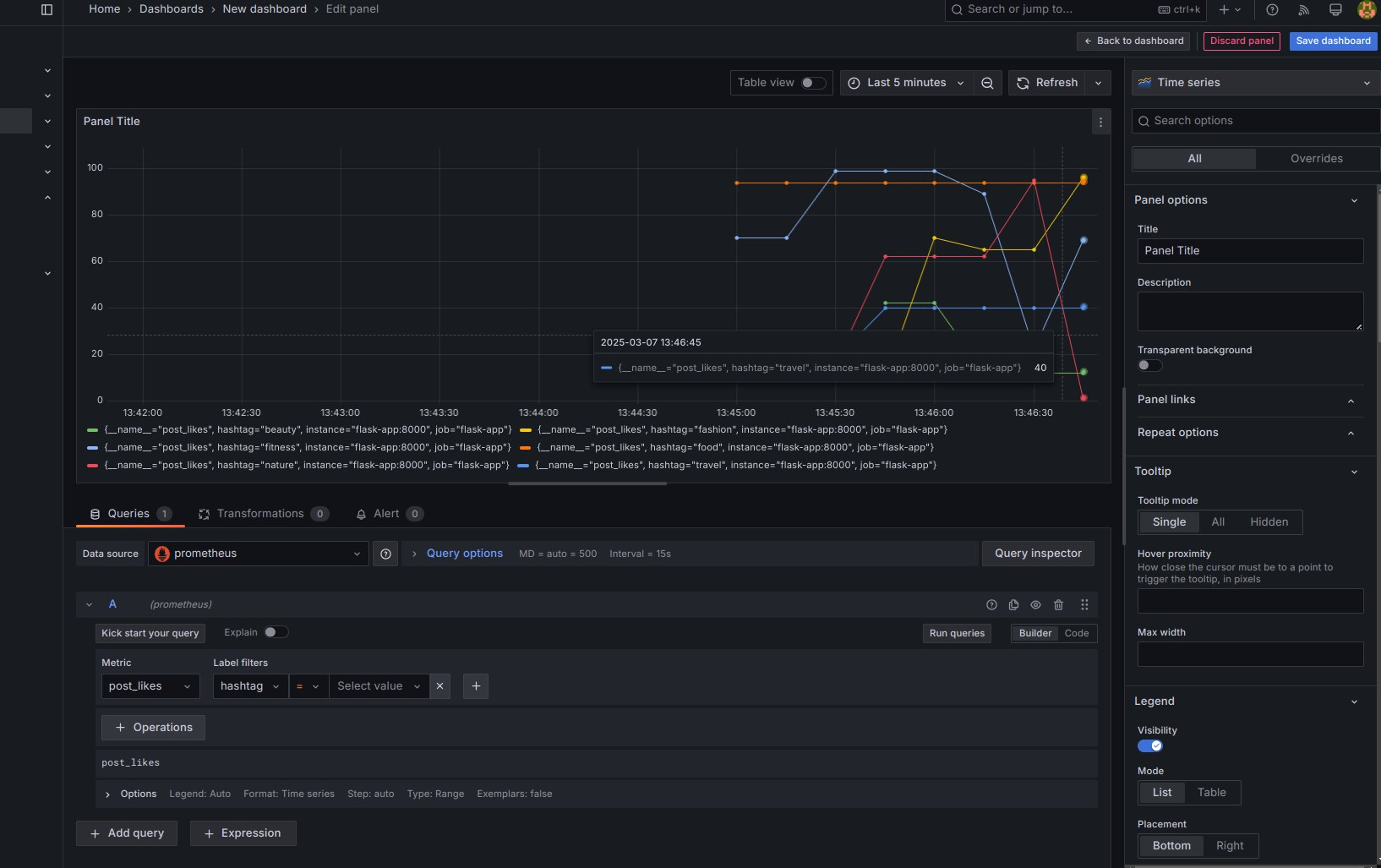


Рисунок 7 – Дашборд Grafana

**Вывод:**

В ходе выполнения задания была изучена технология, которая позволяет работать с множеством сервисов одновременно.

В рамках работы было подключено 4 сервиса:

1. Backend на Python Flask
2. Neo4j
3. Prometheus
4. Grafana
5. Был настроен просмотр метрик и их зависимостей в Grafana

**Контрольные вопросы:**

1. Что такое Docker Compose и для чего он используется?

Docker Compose — это инструмент для управления многоконтейнерными Docker приложениями. Он позволяет описывать конфигурацию всех контейнеров в одном YAML-файле и управлять ими с помощью простых команд. Используется для автоматизации развертывания и управления сервисами, что особенно полезно при разработке и тестировании сложных приложений.

2. Основные преимущества использования Docker Compose:

* Позволяет запускать несколько контейнеров одной командой (docker-compose up).
* Описание всех сервисов в docker-compose.yml делает проект более структурированным.
* Все контейнеры в одном файле автоматически подключаются к общей сети.
* Можно легко изменять параметры, добавлять новые сервисы и масштабировать их.
* Легко воссоздавать окружение разработки, тестирования и продакшена.

3. Основные разделы и директивы в docker-compose.yml:

Пример типового docker-compose.yml:

version: "3.8"

services:

app:

image: my-app:latest

build: . # Сборка из Dockerfile

ports:

- "8000:8000"

environment:

- DATABASE\_URL=mysql://user:password@db:3306/mydb

depends\_on:

- db

db:

image: mysql:latest

restart: always

environment:

MYSQL\_ROOT\_PASSWORD: root

MYSQL\_DATABASE: mydb

volumes:

- db\_data:/var/lib/mysql

volumes:

db\_data:

*Основные директивы:*

version – версия синтаксиса Docker Compose.

services – список контейнеров.

image – используемый образ.

build – путь к Dockerfile.

ports – проброс портов.

environment – переменные окружения.

depends\_on – порядок запуска зависимых сервисов.

volumes – определение и подключение томов.

4. Как запустить многоконтейнерное приложение с помощью Docker Compose?

Запуск выполняется командой:

docker-compose up -d

Если нужно пересобрать образы перед запуском:

docker-compose up --build -d

5. Как остановить и удалить контейнеры, запущенные с помощью Docker Compose?

Остановка контейнеров:

docker-compose down

Если нужно также удалить тома:

docker-compose down -v

Если нужно остановить контейнеры, но оставить их данные:

docker-compose stop