Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет

информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра Информатики

Дисциплина «Конструирование программ»

ОТЧЕТ

к лабораторной работе №2

на тему:

«Работа с Docker»

БГУИР 6-05-0612-02 23

|  |
| --- |
| Выполнил студент группы 353504  ВОЙНО Никита Андреевич |
|  |
| (дата, подпись студента) |
| Проверил доцент каф. Информатики  ЖВАКИНА Анна Васильевна |
|  |
| (дата, подпись преподавателя) |

Минск 2024

1. **ИНДИВИДУАЛЬНЫЕ ЗАДАНИЯ**
2. Подготовьте рабочее окружение в соответствии с типом вашей операционной системы

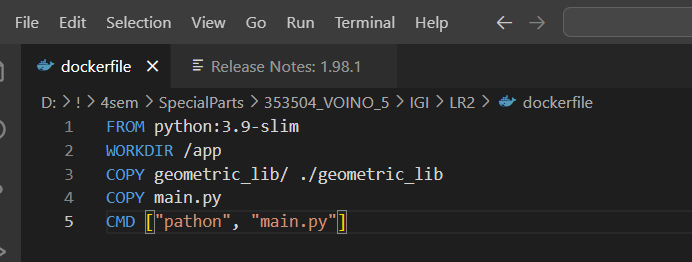
* Установите Docker
* Выполните базовую настройку

1. Изучите простейшие консольные команды и возможности Docker Desktop (см.лекцию), создать собственный контейнер docker/getting-started, открыть в браузере и изучить tutorial
2. Создайте docker image, который запускает скрипт с использованием функций из https://github.com/smartiqaorg/geometric\_lib.
3. Данные необходимые для работы скрипта передайте любым удобным способом (например: конфиг файл через docker volume, переменные окружения, перенаправление ввода). Изучите простейшие консольные команды для работы с docker(см. лекцию). Зарегистрируйтесь на DockerHub и выберите необходимые для проекта образы
4. Создать Dockerfile для реализации сборки собственных Docker образов
5. Использовать его для создания контейнера. Протестировать использование контейнера
6. Скачать любой доступный проект с GitHub с произвольным стеком технологий(пример – см. индивидуальное задание) или использовать свой, ранее разработанный. Создать для него необходимый контейнер, используя Docker Compose для управления многоконтейнерными приложениями. Запустить проект в контейнере.( Примеры Images:https://hub.docker.com/\_/phpmyadmin, https://hub.docker.com/\_/mysql, https://hub.docker.com/\_/postgres)
7. Настроить сети и тома для обеспечения связи между контейнерами и сохранения данных (исходные данные, логин, пароль и т.д.)
8. Разместите результат в созданный репозиторий в DockerHub
9. Выполните следующие действия с целью изучить особенности сетевого взаимодействия:

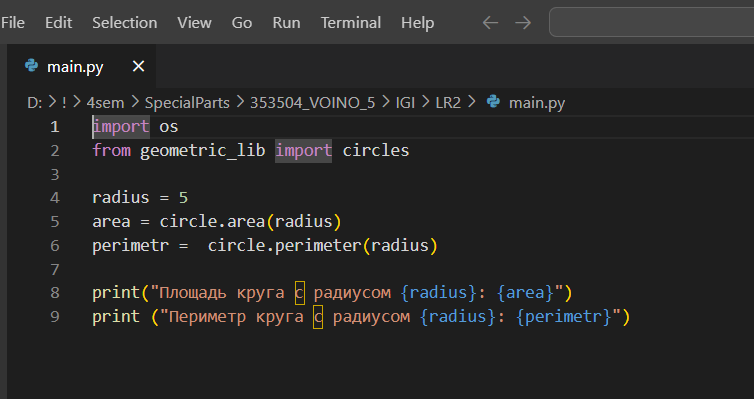
* Получить информацию о всех сетях, работающих на текущем хосте и подробности о каждом типе сети
* Создать свою собственную сеть bridge, проверить, создана ли она, запустить Docker-контейнер в созданной сети, вывести о ней всю информацию(включая IP-адрес контейнера), отключить сеть от контейнера
* Создать еще одну сеть bridge, вывести о ней всю информацию, запустить в ней три контейнера, подключиться к любому из контейнеров и пропинговать два других из оболочки контейнера, убедиться, что между контейнерами происходит общение по IP-адресу
* 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.

**2 ХОД РАБОТЫ**

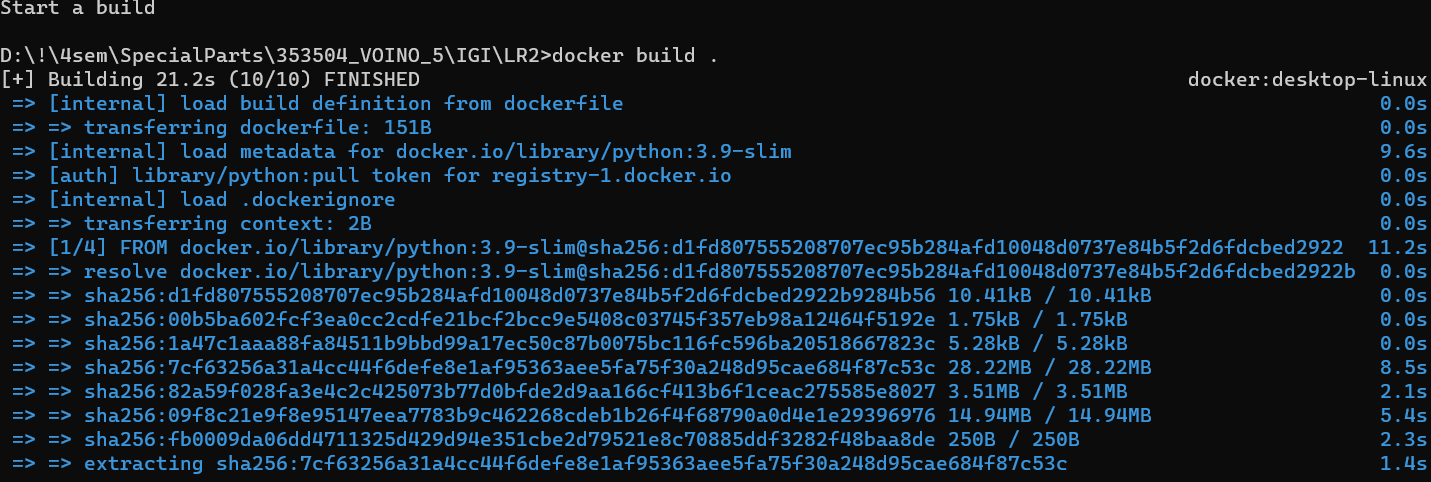
Создания dockerfile для создания образа репозитория:

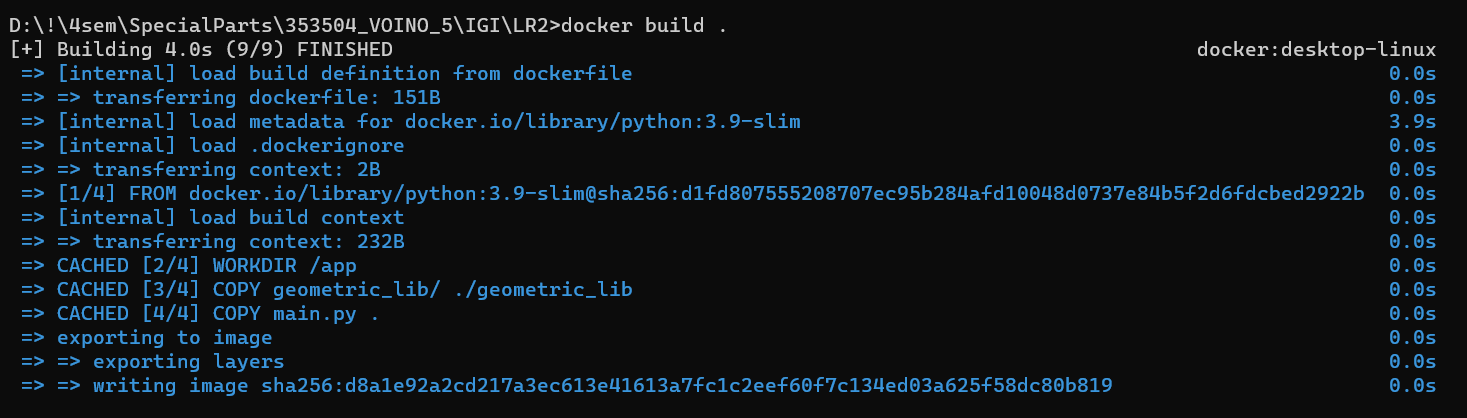


Создание main.py:

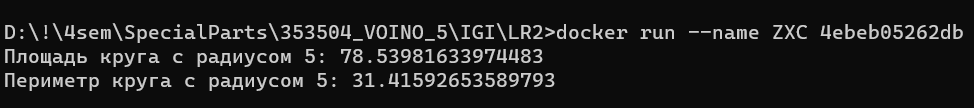


Билд и запуск контейнера:

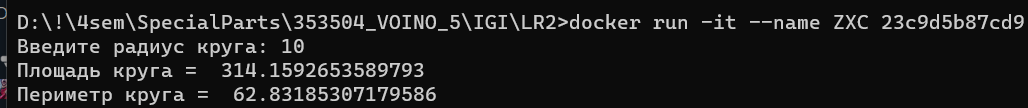




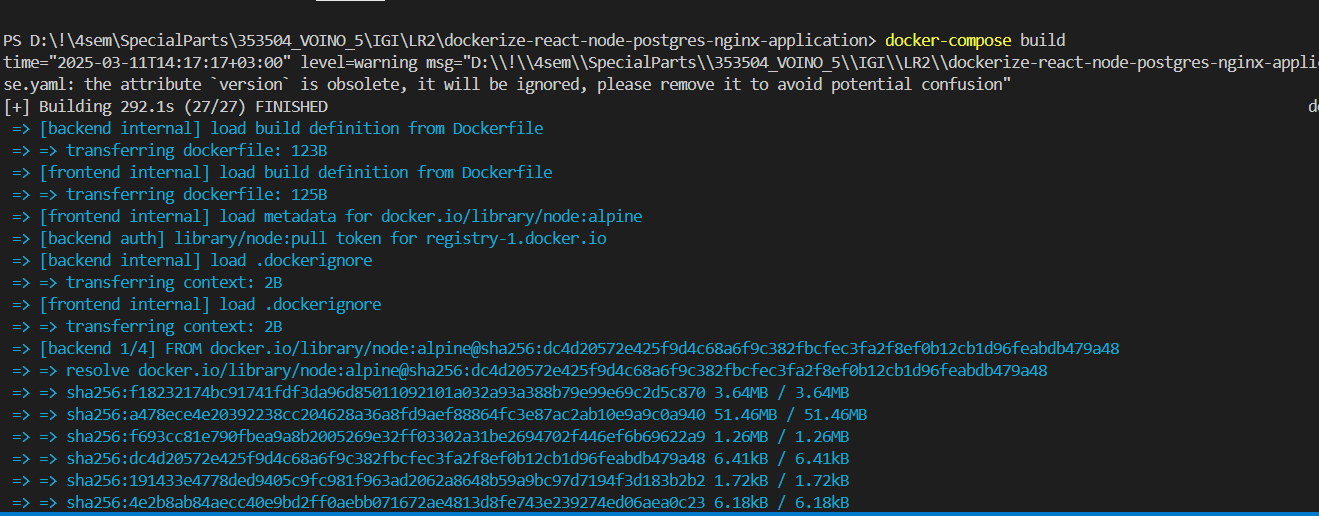
Вывод программы:



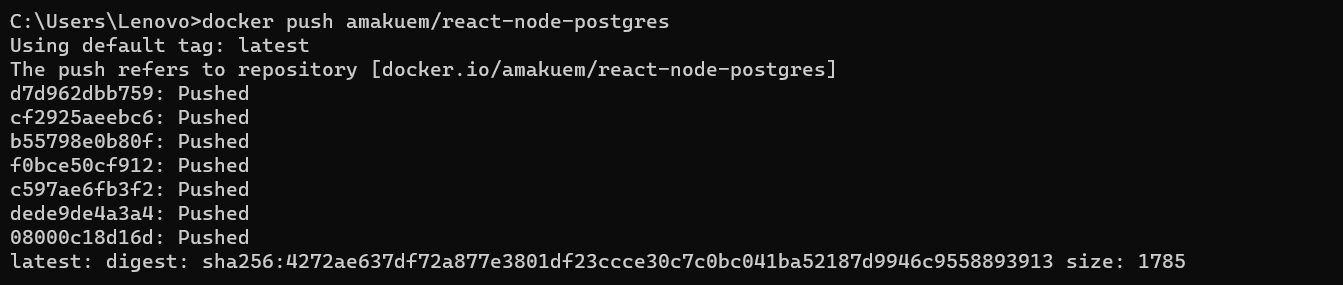
Изменение кода и задания радиуса через терминал:



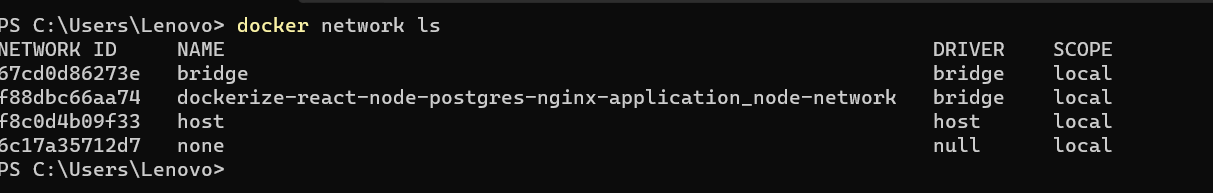
Билд docker-compose:



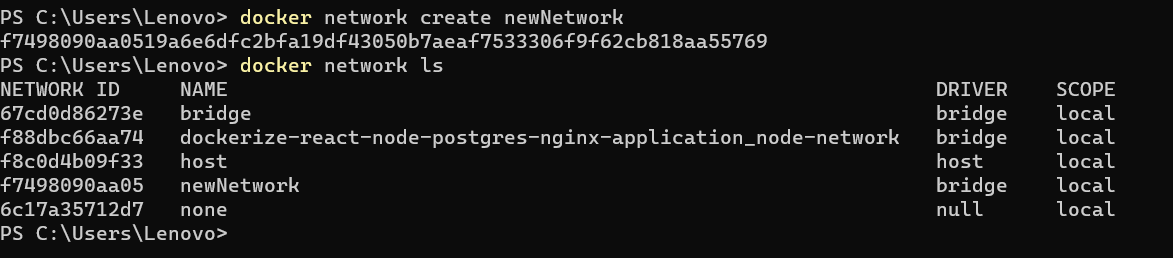
После проверки работы созданного контейнера пуш на DockerHub:



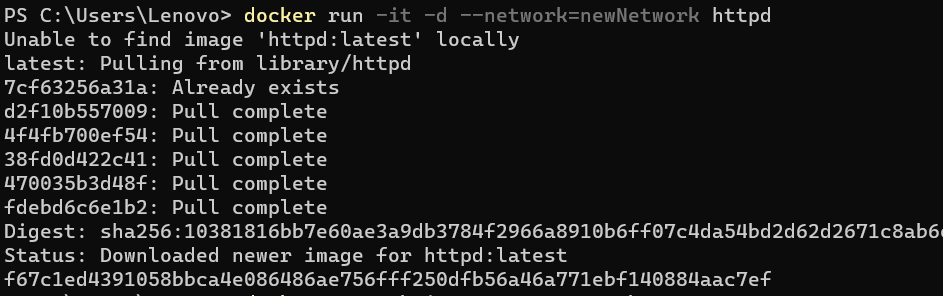
Проверка всех сетей:

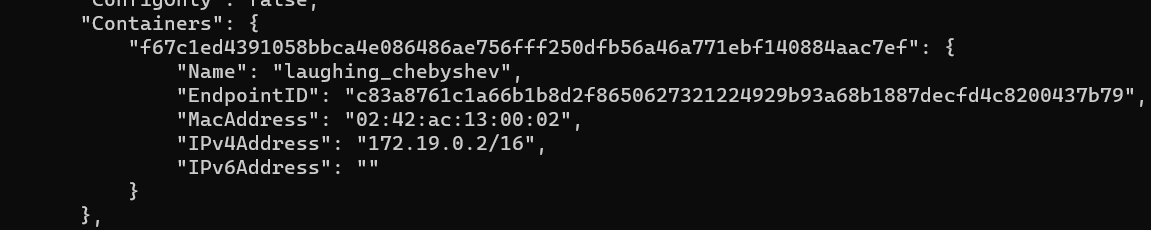


Создание сетки bridge:

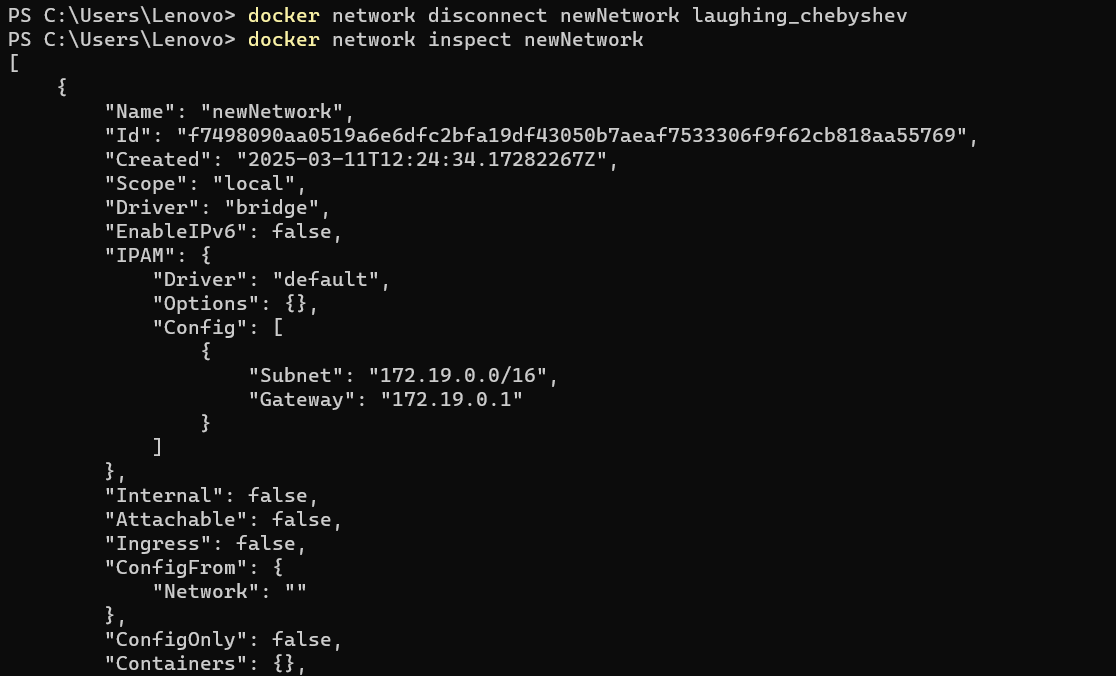


Запуск контейнера в созданной сети:





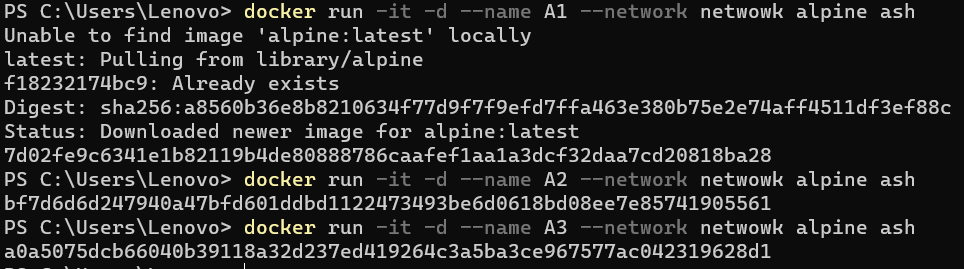
Отключении сети от контейнера:

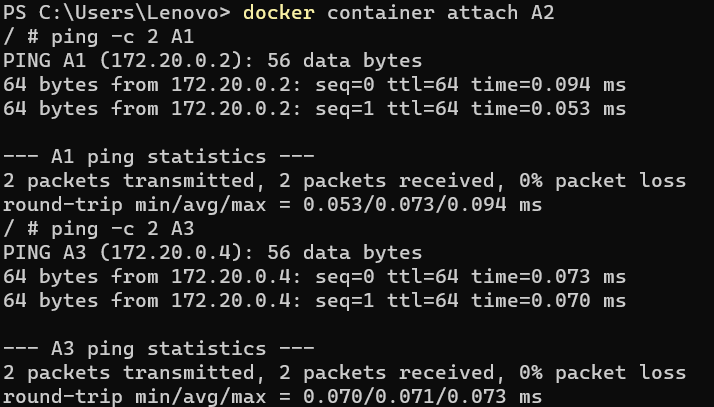


Создание новой bridge сети:



Создание там трех контейнеров и пропинговка из одного остальных:





**ВЫВОД**

В ходе данной лабораторной работы были изучены основные команды Docker и работа с ним.