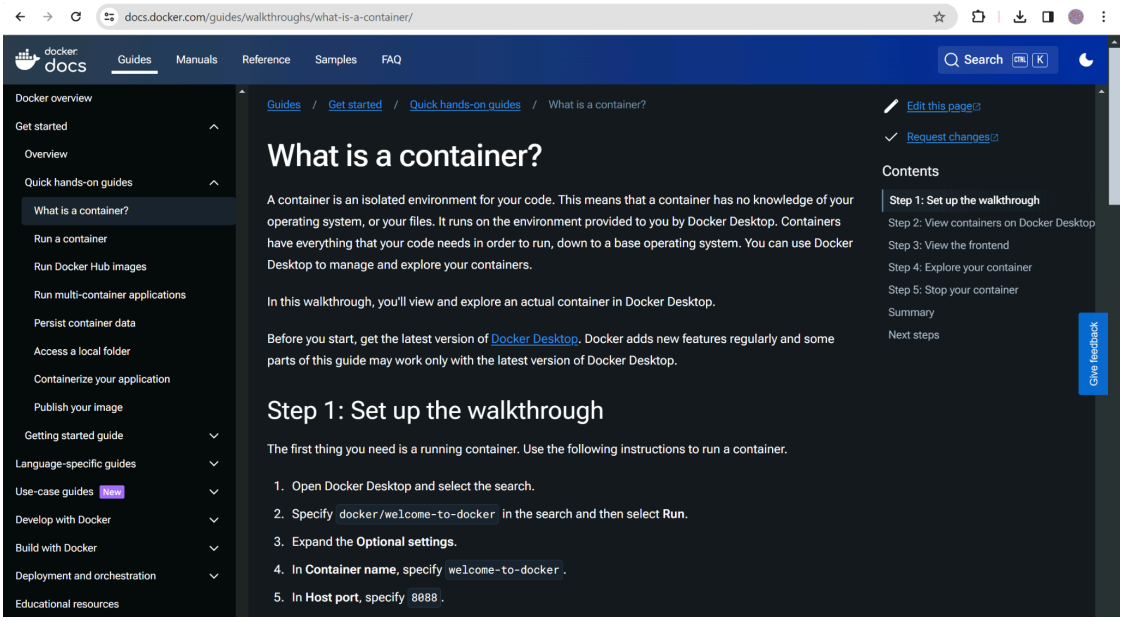
Лабораторная работа №2

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

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

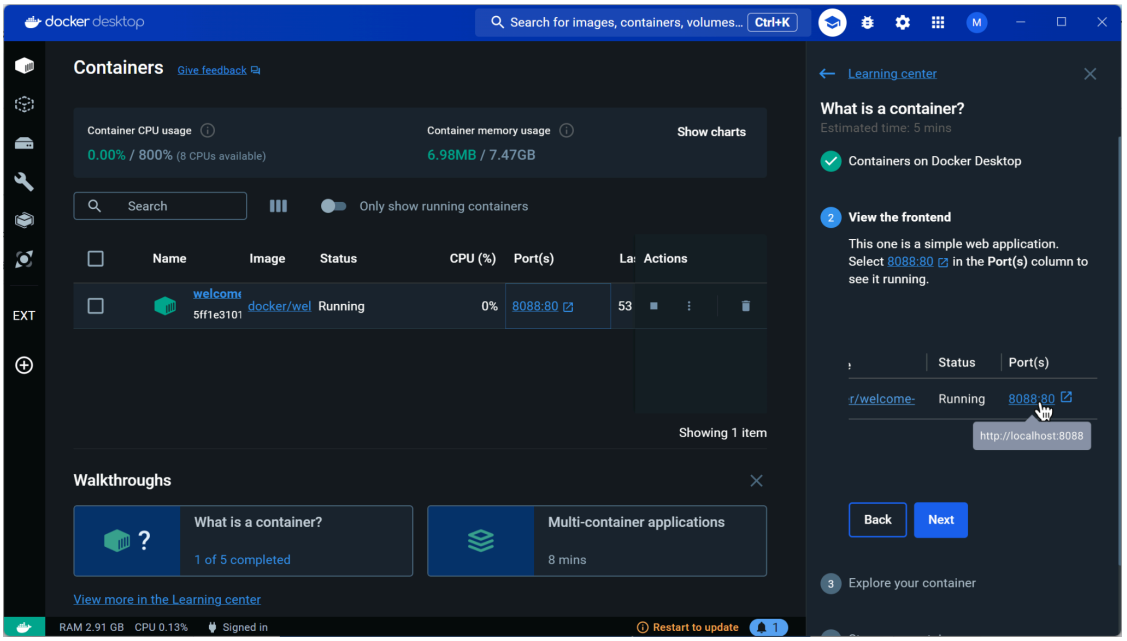
- Установите Docker
- Выполните базовую настройку

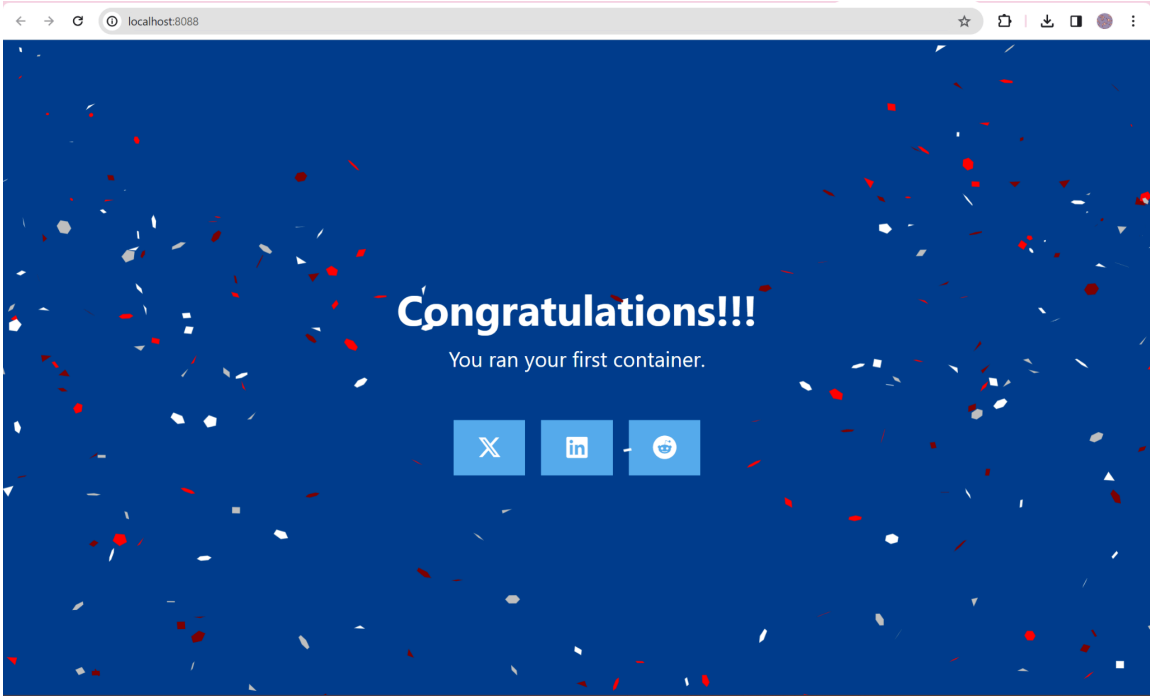
Установим Docker и ознакомимся с документацией



Задание 2. Изучите простейшие консольные команды и возможности Docker Desktop (см. лекцию), создать собственный контейнер docker/getting-started, открыть в браузере и изучить tutorial

Откроем Docker Desktop и запустим по-умолчанию созданный контейнер



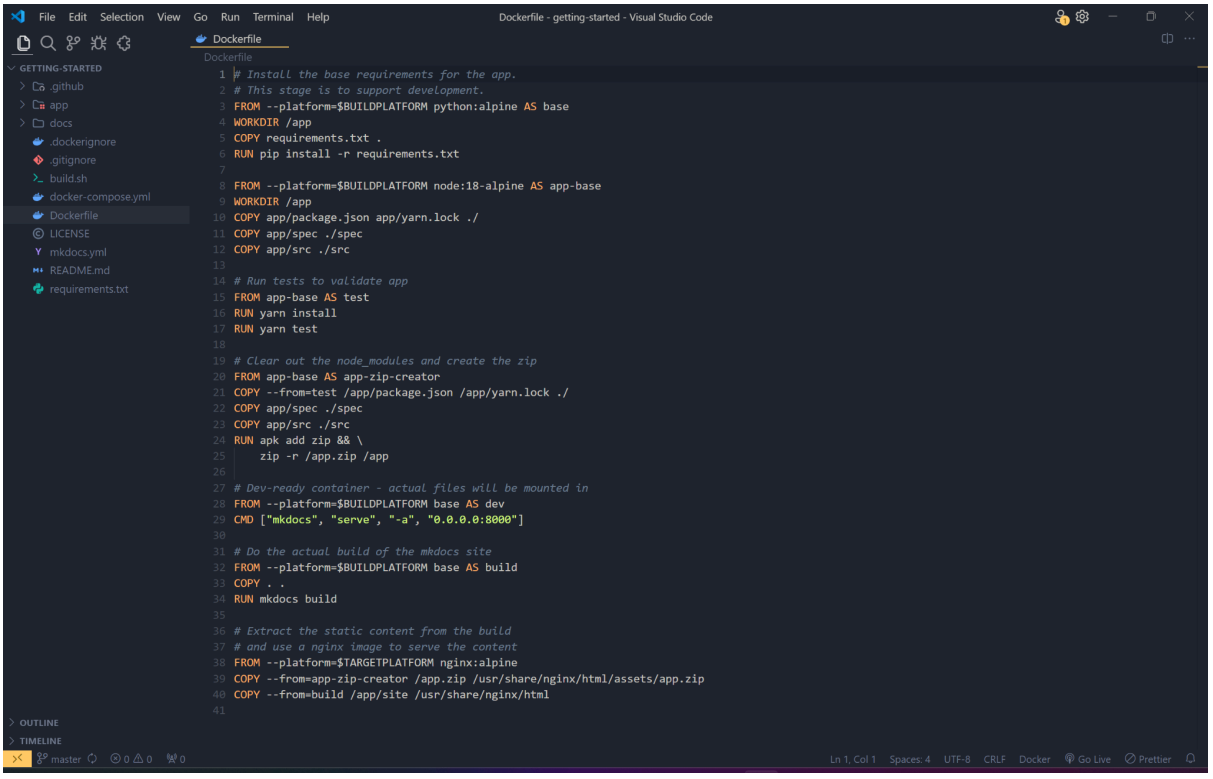


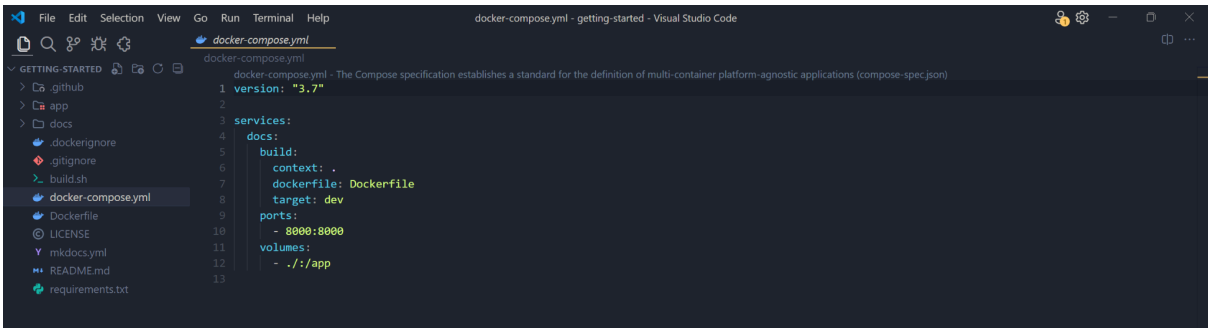
Клонируем репозиторий с GitHub для запуска следующего контейнера:

```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> cd getting-started
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\getting-started> ls


    Каталог: D:\IGI\253502_KRASYOV_11\IGI\LR2\getting-started


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----         3/4/2024   7:52 PM                .github
d-----         3/4/2024   7:52 PM                app
d-----         3/4/2024   7:52 PM                docs
-a----         3/4/2024   7:52 PM             28 .dockerignore
-a----         3/4/2024   7:52 PM             28 .gitignore
-a----         3/4/2024   7:52 PM            267 build.sh
-a----         3/4/2024   7:52 PM            179 docker-compose.yml
-a----         3/4/2024   7:52 PM           1223 Dockerfile
-a----         3/4/2024   7:52 PM          11556 LICENSE
-a----         3/4/2024   7:52 PM           2076 mkdocs.yml
-a----         3/4/2024   7:52 PM           1739 README.md
-a----         3/4/2024   7:52 PM            110 requirements.txt
```

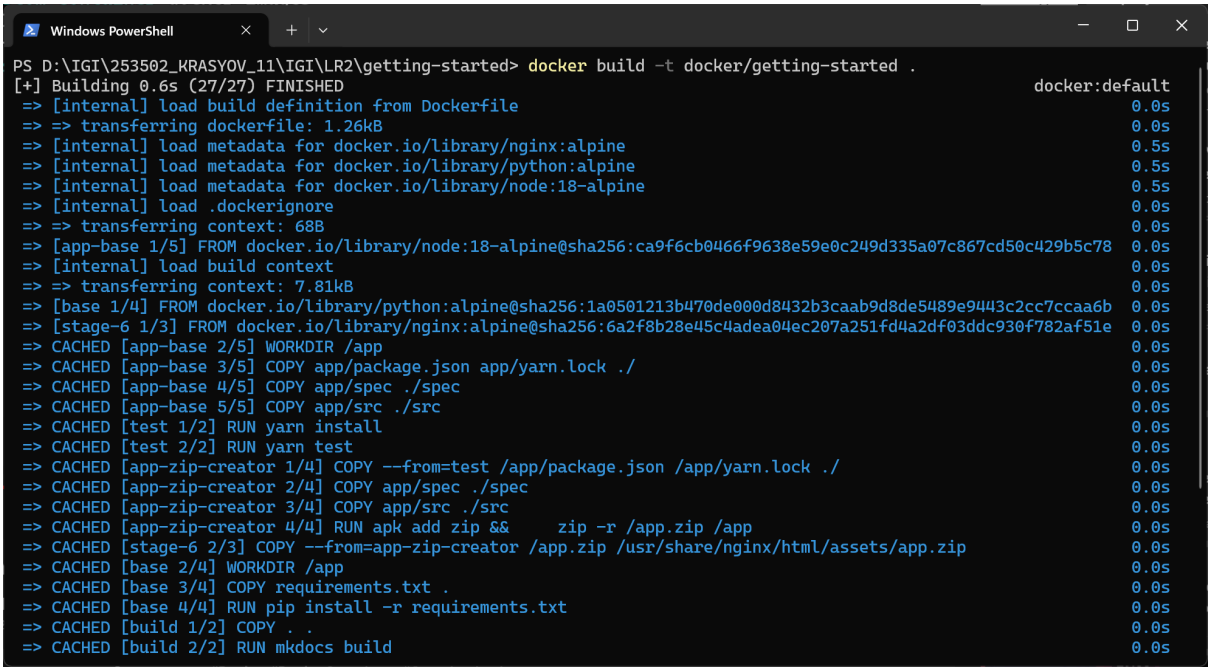
Изучим его Dockerfile:

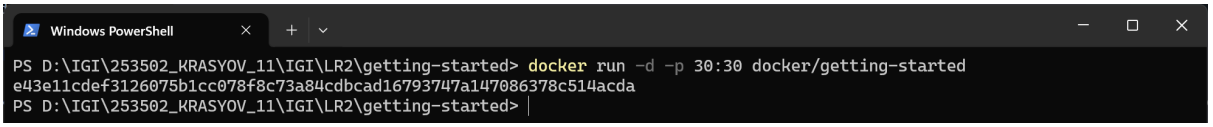
```
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\getting-started> code . Dockerfile
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\getting-started>
```

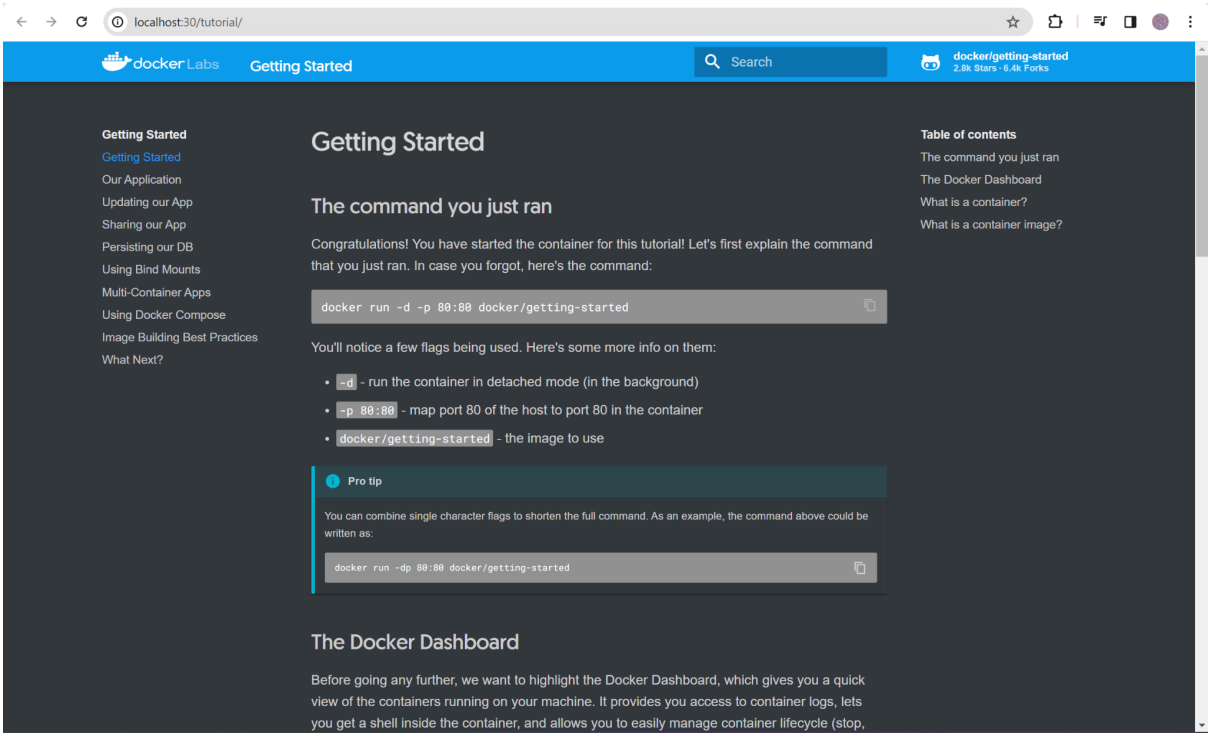




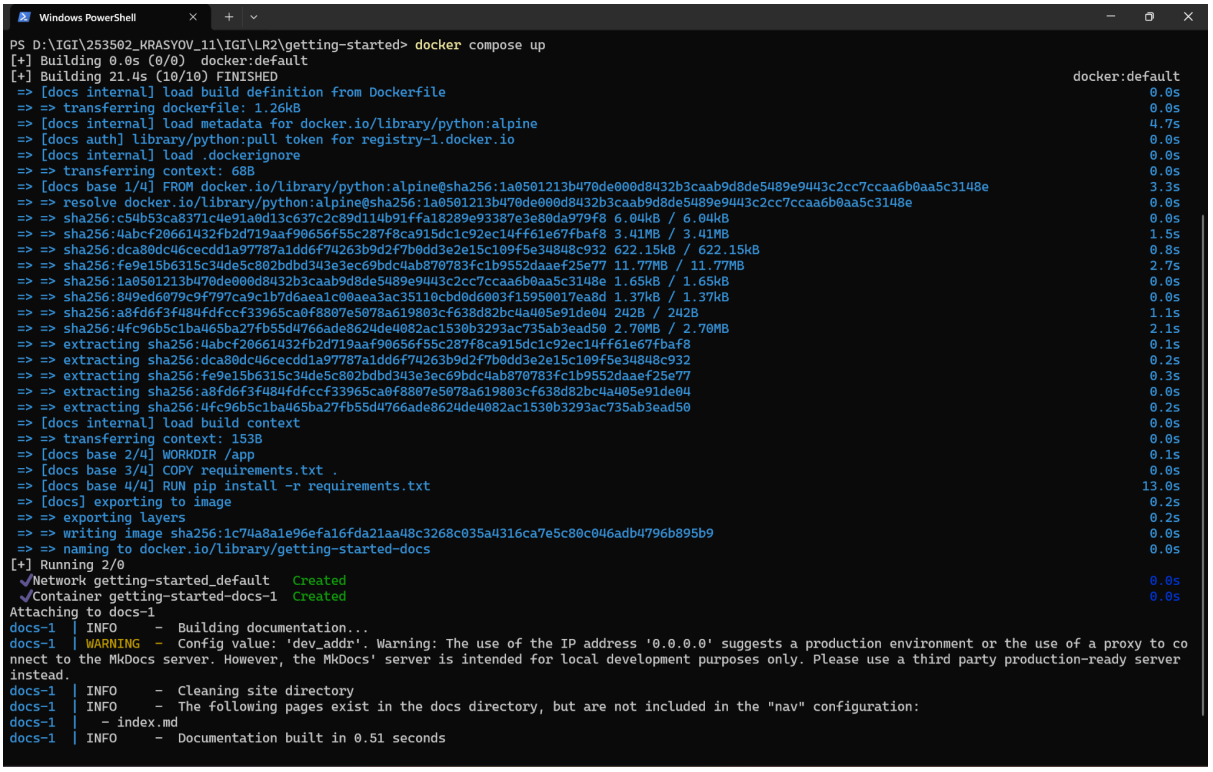
Соберём образ и запустим данный контейнер

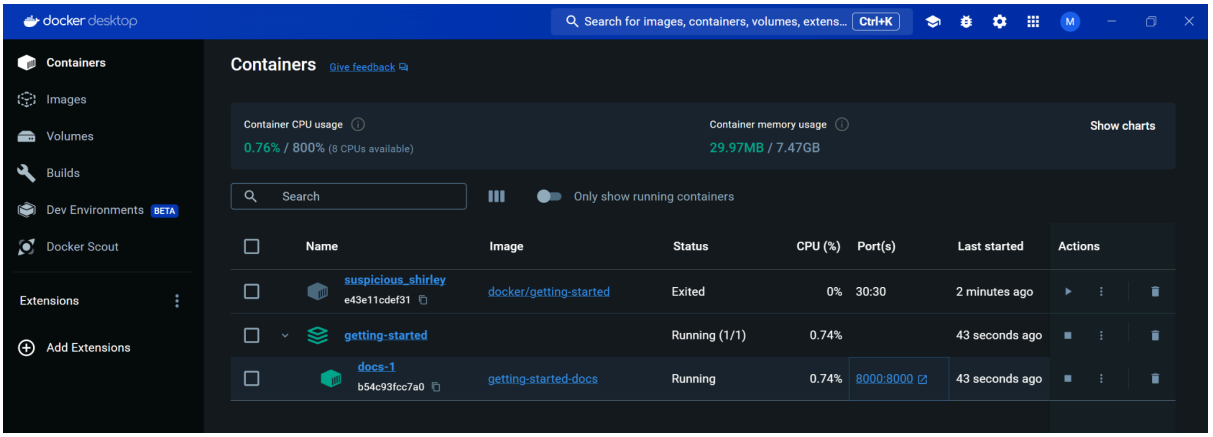


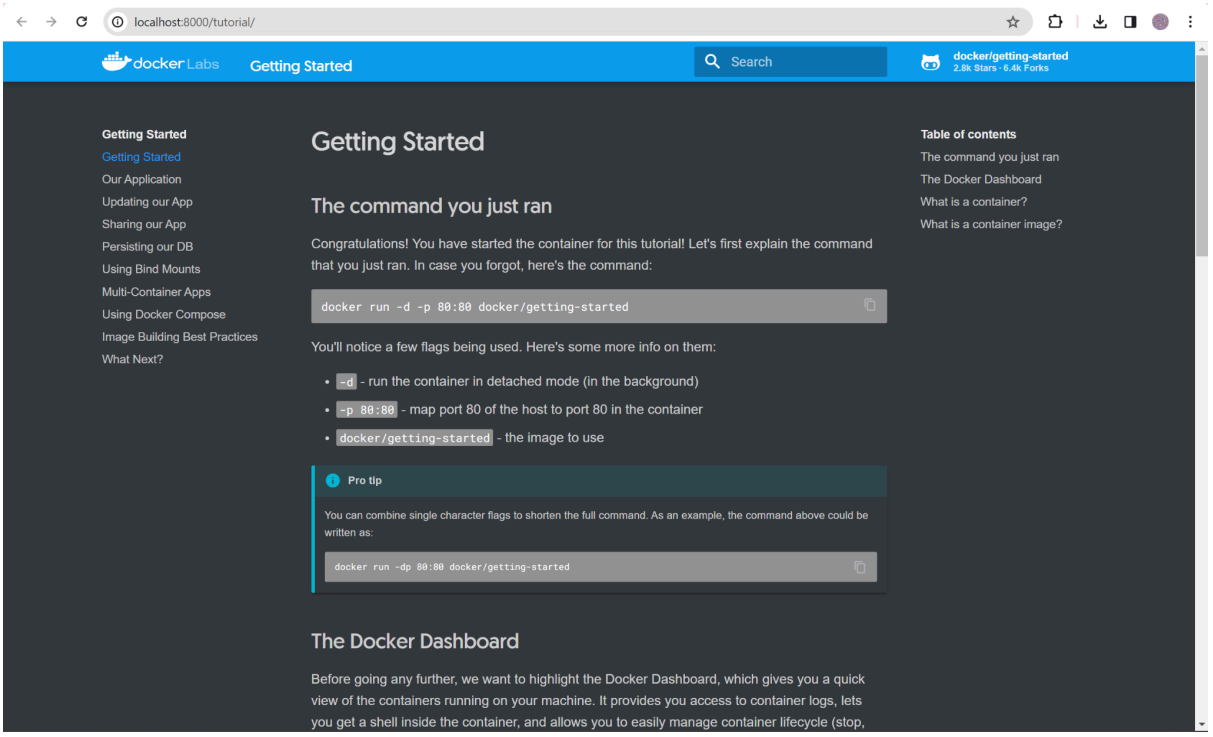




Используем compose чтобы собрать все контейнеры приложения вместе:


Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\getting-started> docker compose up
[+] Building 0.0s (0/0) docker:default
[+] Building 21.4s (10/10) FINISHED docker:default
=> [docs internal] load build definition from Dockerfile 0.0s
=> => transferring dockerfile: 1.26kB 0.0s
=> [docs internal] load metadata for docker.io/library/python:alpine 4.7s
=> [docs auth] library/python:pull token for registry-1.docker.io 0.0s
=> [docs internal] load .dockerignore 0.0s
=> => transferring context: 68B 0.0s
=> [docs base 1/4] FROM docker.io/library/python:alpine@sha256:1a0501213b470de000d8432b3caab9d8de5489e9443c2cc7ccaa6b0aa5c3148e 3.3s
=> => resolve docker.io/library/python:alpine@sha256:1a0501213b470de000d8432b3caab9d8de5489e9443c2cc7ccaa6b0aa5c3148e 0.0s
=> => sha256:c54b53ca8371c4e91a0d13c637c2c89d114b91ffa18289e93387e3e80da979f8 6.04kB / 6.04kB 0.0s
=> => sha256:4abcf20661432fb2d719aaf90656f55c287f8ca915dc1c92ec14ff61e67fbaf8 3.41MB / 3.41MB 1.5s
=> => sha256:dca80dc46cecdd1a97787a1dd6f74263b9d2f7b0dd3e2e15c109f5e34848c932 622.15kB / 622.15kB 0.8s
=> => sha256:fe9e15b6315c34de5c802bdbd343e3ec69bdc4ab870783fc1b9552daaef25e77 11.77MB / 11.77MB 2.7s
=> => sha256:1a0501213b470de000d8432b3caab9d8de5489e9443c2cc7ccaa6b0aa5c3148e 1.65kB / 1.65kB 0.0s
=> => sha256:849ed6079c9f797ca9c1b7d6aea1c00aea3ac35110cbd0d6003f15950017ea8d 1.37kB / 1.37kB 0.0s
=> => sha256:a8fd6f3f484fdfccf33965ca0f8807e5078a619803cf638d82bc4a405e91de04 242B / 242B 1.1s
=> => sha256:4fc96b5c1ba465ba27fb55d4766ade8624de4082ac1530b3293ac735ab3ead50 2.70MB / 2.70MB 2.1s
=> => extracting sha256:4abcf20661432fb2d719aaf90656f55c287f8ca915dc1c92ec14ff61e67fbaf8 0.1s
=> => extracting sha256:dca80dc46cecdd1a97787a1dd6f74263b9d2f7b0dd3e2e15c109f5e34848c932 0.2s
=> => extracting sha256:fe9e15b6315c34de5c802bdbd343e3ec69bdc4ab870783fc1b9552daaef25e77 0.3s
=> => extracting sha256:a8fd6f3f484fdfccf33965ca0f8807e5078a619803cf638d82bc4a405e91de04 0.0s
=> => extracting sha256:4fc96b5c1ba465ba27fb55d4766ade8624de4082ac1530b3293ac735ab3ead50 0.2s
=> [docs internal] load build context 0.0s
=> => transferring context: 153B 0.0s
=> [docs base 2/4] WORKDIR /app 0.1s
=> [docs base 3/4] COPY requirements.txt . 0.0s
=> [docs base 4/4] RUN pip install -r requirements.txt 13.0s
=> [docs] exporting to image 0.2s
=> => exporting layers 0.2s
=> => writing image sha256:1c74a8a1e96efa16fda21aa48c3268c035a4316ca7e5c80c046adb4796b895b9 0.0s
=> => naming to docker.io/library/getting-started-docs 0.0s
[+] Running 2/0
Network getting-started_default Created 0.0s
Container getting-started-docs-1 Created 0.0s
Attaching to docs-1
docs-1 | INFO - Building documentation...
docs-1 | WARNING - Config value: 'dev_addr'. Warning: The use of the IP address '0.0.0.0' suggests a production environment or the use of a proxy to connect to the MkDocs server. However, the MkDocs' server is intended for local development purposes only. Please use a third party production-ready server instead.
docs-1 | INFO - Cleaning site directory
docs-1 | INFO - The following pages exist in the docs directory, but are not included in the "nav" configuration:
docs-1 | - index.md
docs-1 | INFO - Documentation built in 0.51 seconds

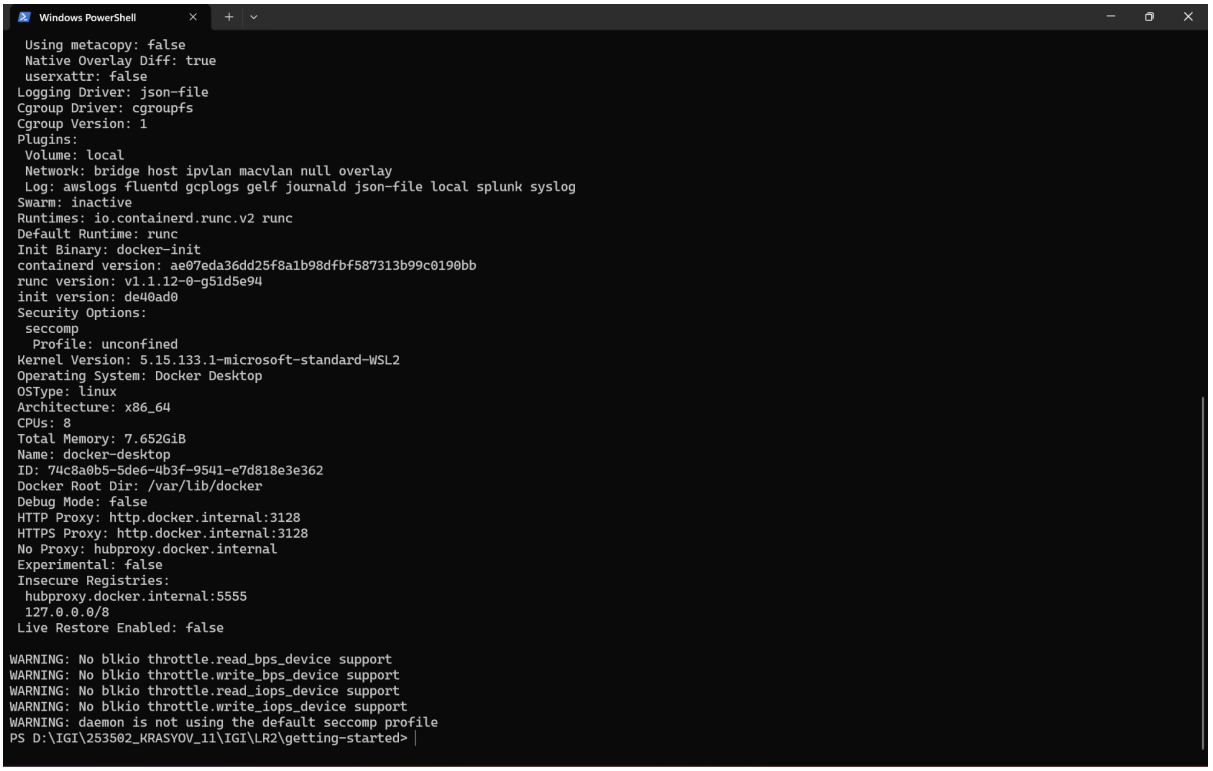

docker desktop
Search for images, containers, volumes, extens... Ctrl+K
Containers
Images
Volumes
Builds
Dev Environments BETA
Docker Scout
Extensions
Add Extensions
Containers
Give feedback
Container CPU usage
0.76% / 800% (8 CPUs available)
Container memory usage
29.97MB / 7.47GB
Show charts
Search
Only show running containers
Name
Image
Status
CPU (%)
Port(s)
Last started
Actions
suspicious_shirley
e43e11cdef31
docker/getting-started
Exited
0%
30:30
2 minutes ago
getting-started
Running (1/1)
0.74%
43 seconds ago
docs-1
b54c93fcc7a0
getting-started-docs
Running
0.74%
8000:8000
43 seconds ago



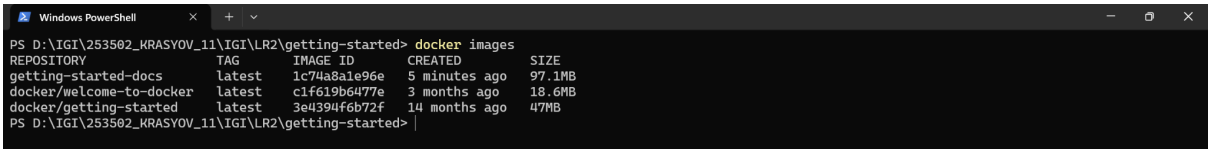
Проверим различные команды Docker:

Просмотр информации о пользователе:

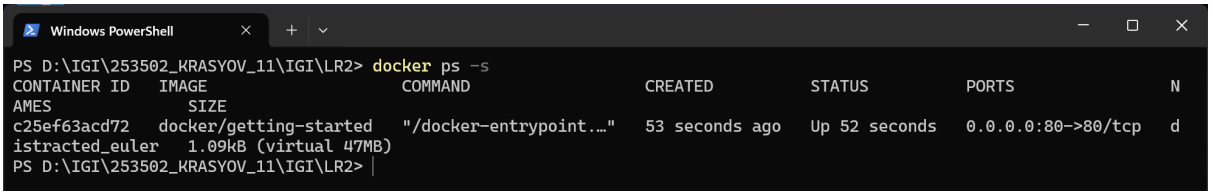
```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\getting-started> docker info
Client:
 Version:    25.0.3
 Context:    default
 Debug Mode: false
 Plugins:
  buildx: Docker Buildx (Docker Inc.)
    Version:  v0.12.1-desktop.4
    Path:     C:\Program Files\Docker\cli-plugins\docker-buildx.exe
  compose: Docker Compose (Docker Inc.)
    Version:  v2.24.6-desktop.1
    Path:     C:\Program Files\Docker\cli-plugins\docker-compose.exe
  debug: Get a shell into any image or container. (Docker Inc.)
    Version:  0.0.24
    Path:     C:\Program Files\Docker\cli-plugins\docker-debug.exe
  dev: Docker Dev Environments (Docker Inc.)
    Version:  v0.1.0
    Path:     C:\Program Files\Docker\cli-plugins\docker-dev.exe
  extension: Manages Docker extensions (Docker Inc.)
    Version:  v0.2.22
    Path:     C:\Program Files\Docker\cli-plugins\docker-extension.exe
  feedback: Provide feedback, right in your terminal! (Docker Inc.)
    Version:  v1.0.4
    Path:     C:\Program Files\Docker\cli-plugins\docker-feedback.exe
  init: Creates Docker-related starter files for your project (Docker Inc.)
    Version:  v1.0.1
    Path:     C:\Program Files\Docker\cli-plugins\docker-init.exe
  sbom: View the packaged-based Software Bill Of Materials (SBOM) for an image (Anchore Inc.)
    Version:  0.6.0
    Path:     C:\Program Files\Docker\cli-plugins\docker-sbom.exe
  scout: Docker Scout (Docker Inc.)
    Version:  v1.5.0
    Path:     C:\Program Files\Docker\cli-plugins\docker-scout.exe

Server:
 Containers: 3
  Running: 0
  Paused: 0
  Stopped: 3
 Images: 3
 Server Version: 25.0.3
 Storage Driver: overlay2
  Backing Filesystem: extfs
  Supports d_type: true
  Using metacopy: false
```



Просмотр всех образов:



Просмотр всех/запущенных контейнеров:



Просмотр логов контейнера:

```
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker logs c25ef63acd72
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2024/03/10 06:38:57 [notice] 1#1: using the "epoll" event method
2024/03/10 06:38:57 [notice] 1#1: nginx/1.23.3
2024/03/10 06:38:57 [notice] 1#1: built by gcc 12.2.1 20220924 (Alpine 12.2.1_git20220924-r4)
2024/03/10 06:38:57 [notice] 1#1: OS: Linux 5.15.133.1-microsoft-standard-WSL2
2024/03/10 06:38:57 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2024/03/10 06:38:57 [notice] 1#1: start worker processes
2024/03/10 06:38:57 [notice] 1#1: start worker process 30
2024/03/10 06:38:57 [notice] 1#1: start worker process 31
2024/03/10 06:38:57 [notice] 1#1: start worker process 32
2024/03/10 06:38:57 [notice] 1#1: start worker process 33
2024/03/10 06:38:57 [notice] 1#1: start worker process 34
2024/03/10 06:38:57 [notice] 1#1: start worker process 35
2024/03/10 06:38:57 [notice] 1#1: start worker process 36
2024/03/10 06:38:57 [notice] 1#1: start worker process 37
PS D:\IGI\253502_KRASYOV_11\IGI\LR2>
```

Остановка контейнера:

```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker stop c25ef63acd72
c25ef63acd72
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker ps -s
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES     SIZE
PS D:\IGI\253502_KRASYOV_11\IGI\LR2>
```

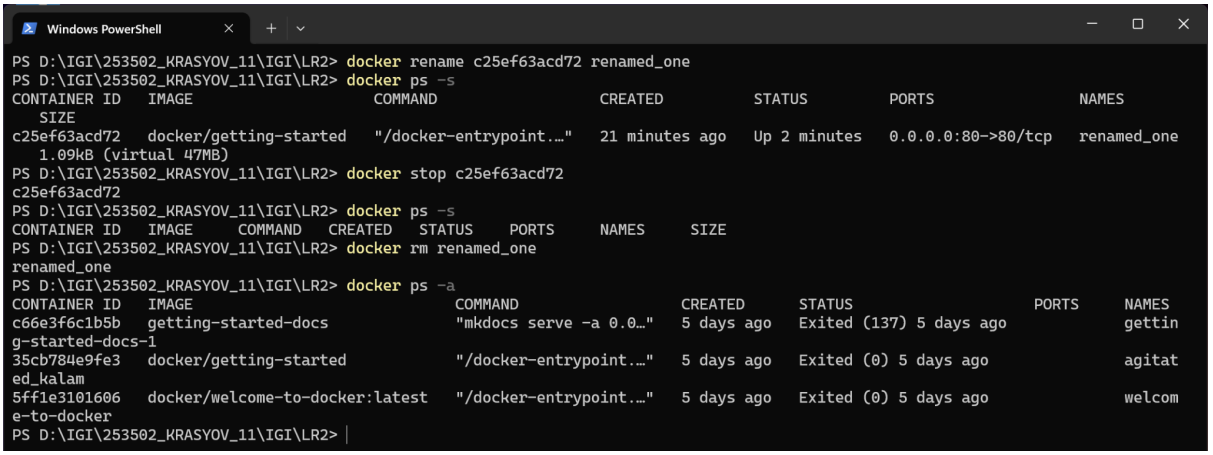
Запуск, приостановка и возобновление работы контейнера:

```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker start c25ef63acd72a4c3ded6852ead5d75d40a3fbc825004fd78589fbfbbc92bbea2
c25ef63acd72a4c3ded6852ead5d75d40a3fbc825004fd78589fbfbbc92bbea2
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker ps -s
CONTAINER ID   IMAGE                    COMMAND                  CREATED          STATUS         PORTS               NA
MES              SIZE
c25ef63acd72   docker/getting-started   "/docker-entrypoint.…"   16 minutes ago   Up 5 seconds   0.0.0.0:80->80/tcp   di
stracted_euler   1.09kB (virtual 47MB)
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker pause c25ef63acd72a4c3ded6852ead5d75d40a3fbc825004fd78589fbfbbc92bbea2
c25ef63acd72a4c3ded6852ead5d75d40a3fbc825004fd78589fbfbbc92bbea2
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker ps -s
CONTAINER ID   IMAGE                    COMMAND                  CREATED          STATUS                     PORTS
        NAMES              SIZE
c25ef63acd72   docker/getting-started   "/docker-entrypoint.…"   16 minutes ago   Up 32 seconds (Paused)   0.0.0.0:80->8
0/tcp   distracted_euler   1.09kB (virtual 47MB)
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker unpause c25ef63acd72a4c3ded6852ead5d75d40a3fbc825004fd78589fbfbbc92bbea2
c25ef63acd72a4c3ded6852ead5d75d40a3fbc825004fd78589fbfbbc92bbea2
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker ps -s
CONTAINER ID   IMAGE                    COMMAND                  CREATED          STATUS          PORTS               N
AMES              SIZE
c25ef63acd72   docker/getting-started   "/docker-entrypoint.…"   16 minutes ago   Up 45 seconds   0.0.0.0:80->80/tcp   d
istracted_euler   1.09kB (virtual 47MB)
PS D:\IGI\253502_KRASYOV_11\IGI\LR2>
```

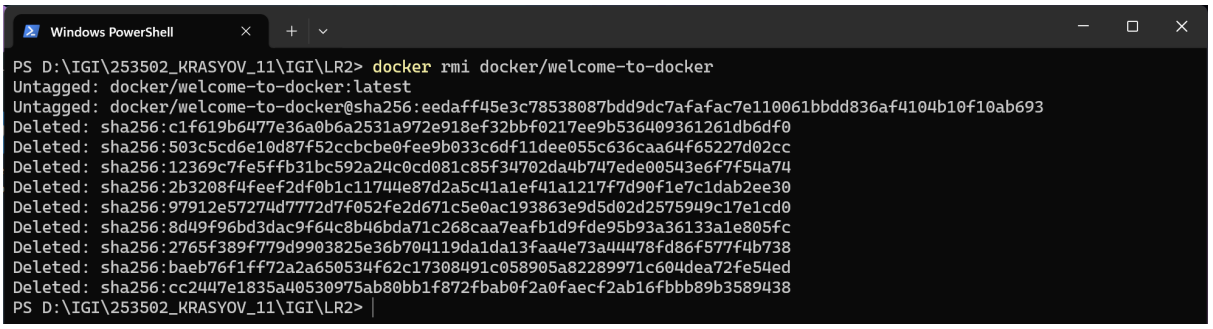
Перезапуск контейнера:

```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker ps -s
CONTAINER ID   IMAGE                    COMMAND                  CREATED          STATUS         PORTS               NA
MES              SIZE
c25ef63acd72   docker/getting-started   "/docker-entrypoint.…"   18 minutes ago   Up 2 minutes   0.0.0.0:80->80/tcp   di
stracted_euler   1.09kB (virtual 47MB)
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker restart c25ef63acd72
c25ef63acd72
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker ps -s
CONTAINER ID   IMAGE                    COMMAND                  CREATED          STATUS         PORTS               NA
MES              SIZE
c25ef63acd72   docker/getting-started   "/docker-entrypoint.…"   18 minutes ago   Up 5 seconds   0.0.0.0:80->80/tcp   di
stracted_euler   1.09kB (virtual 47MB)
PS D:\IGI\253502_KRASYOV_11\IGI\LR2>
```

Переименование и удаление контейнера:



Удаление образа:



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

a. 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

b. Создать Dockerfile для реализации сборки собственных Docker образов

c. Использовать его для создания контейнера. Протестировать использование контейнера

Клонируем GitHub репозиторий

Создаём Dockerfile

```
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\geometric_lib> code . Dockerfile
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\geometric_lib>
```

Dockerfile U

```
FROM python
ENV env_param=7
WORKDIR /app
COPY . .
VOLUME /app/data
ENTRYPOINT [ "python", "./main.py"]
```

Создаём главный скрипт-файл:

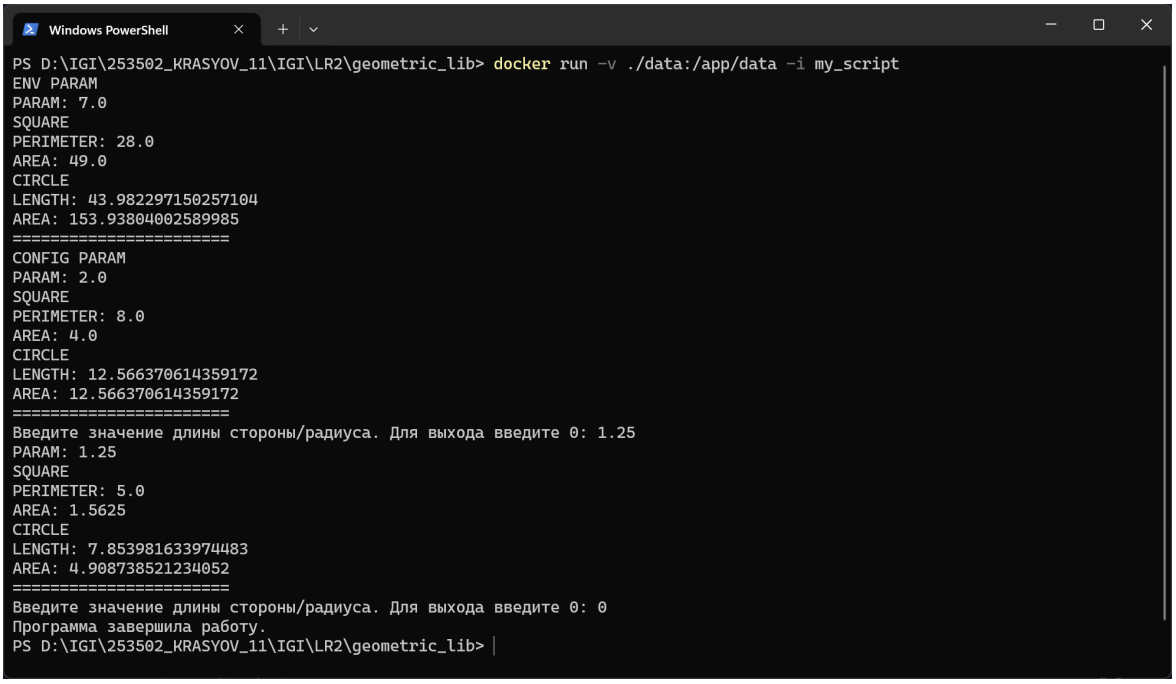
main.py U

```
import os
from square import *
from circle import *
from data import config

value = 1


def printRes(param: float) -> None:
    if (param <= 0):
        return
    print(f"PARAM: {param}")
    print("SQUARE")
    print(f"PERIMETER: {perimeter(param)}")
    print(f"AREA: {area(param)}")
    print("CIRCLE")
    print(f"LENGTH: {perimeterC(param)}")
    print(f"AREA: {areaC(param)}")
    print("======================")


print("ENV PARAM")
printRes(float(os.getenv('env_param')))
print("CONFIG PARAM")
printRes(float(config.param))

while value > 0:
    value = float(input(
        'Введите значение длины стороны/радиуса. Для выхода введите 0: '))
    printRes(value)

print("Программа завершила работу.")
```

Создаём образ:

```
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\geometric_lib> docker build -t my_script .
[+] Building 63.5s (9/9) FINISHED                                                                                  docker:default
 => [internal] load build definition from Dockerfile                                                                         0.0s
 => => transferring dockerfile: 162B                                                                                         0.0s
 => [internal] load metadata for docker.io/library/python:latest                                                             2.3s
 => [auth] library/python:pull token for registry-1.docker.io                                                                0.0s
 => [internal] load .dockerignore                                                                                            0.0s
 => => transferring context: 2B                                                                                              0.0s
 => [1/3] FROM docker.io/library/python:latest@sha256:e83d1f4d0c735c7a54fc9dae3cca8c58473e3b3de08fcb7ba3d342ee75            59.3s
 => => resolve docker.io/library/python:latest@sha256:e83d1f4d0c735c7a54fc9dae3cca8c58473e3b3de08fcb7ba3d342ee75c            0.0s
 => => sha256:a3aef63c6c1029222ff635a72a0fcaddba1fd7a26abdf2491ace8eb3c6544471e 7.11kB / 7.11kB                              0.0s
 => => sha256:e83d1f4d0c735c7a54fc9dae3cca8c58473e3b3de08fcb7ba3d342ee75cfc09d 2.14kB / 2.14kB                               0.0s
 => => sha256:35eff340c0acd837b7962f77ee4b8869385dd6fe7d3928375a08f0a3bdd18beb 2.01kB / 2.01kB                               0.0s
 => => sha256:7bb465c2914923b08ae03b7fc67b92a1ef9b09c4c1eb9d6711b22ee6bbb46a00 49.55MB / 49.55MB                            29.2s
 => => sha256:2b9b41aaa3c52ab268b47da303015b94ced04a1eb02e58860e58b283404974f4 24.05MB / 24.05MB                             4.8s
 => => sha256:49b40be4436eff6fe463f6977159dc727df37cabe65ade75c75c1caa3cb0a234 64.14MB / 64.14MB                            32.6s
 => => sha256:c558fac597f8ecbb7a66712e14912ce1d83b23a92ca8b6ff14eef209ab01aff2 211.12MB / 211.12MB                          53.5s
 => => sha256:11402150a57e537c64dc69a28bba37f13acdedd50d8788894398a7b774786e7d 6.39MB / 6.39MB                              32.1s
 => => extracting sha256:7bb465c2914923b08ae03b7fc67b92a1ef9b09c4c1eb9d6711b22ee6bbb46a00                                    1.8s
 => => extracting sha256:2b9b41aaa3c52ab268b47da303015b94ced04a1eb02e58860e58b283404974f4                                    0.4s
 => => sha256:e1aa7eb23da86d5da03e19f3873497080528674520d90f32ac5eafd31fb8d702d 22.71MB / 22.71MB                            39.1s
 => => extracting sha256:49b40be4436eff6fe463f6977159dc727df37cabe65ade75c75c1caa3cb0a234                                    2.0s
 => => sha256:297428e82567ff4bbd3fc4b5fe375e5aab1af71aaa00d93afe50583cb71d3ae3 244B / 244B                                  32.9s
 => => sha256:28e810024ad5211525fc84f6532311b6fce2848e8e99e765355b0d1430c224b4 2.70MB / 2.70MB                              34.0s
 => => extracting sha256:c558fac597f8ecbb7a66712e14912ce1d83b23a92ca8b6ff14eef209ab01aff2                                    4.6s
```

Запускаем контейнер и проверяем его работу:



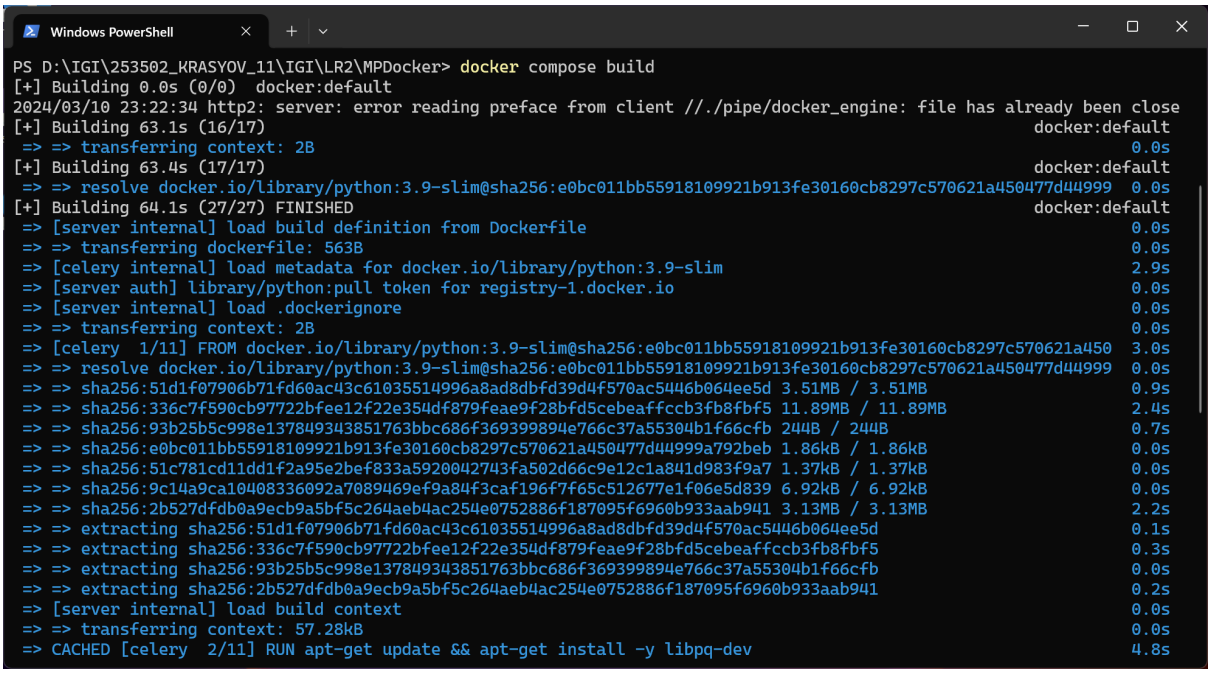
Задание 4. Скачать любой доступный проект с GitHub с произвольным стеком технологий (пример – см. индивидуальное задание) или использовать свой, ранее разработанный. Создать для него необходимый контейнер, используя Docker Compose для управления много контейнерными приложениями. Запустить проект в контейнере **Задание 5.** Настроить сети и тома для обеспечения связи между контейнерами и сохранения данных (исходные данные, логин, пароль и т.д.)

Клонируем проект с GitHub репозитория:

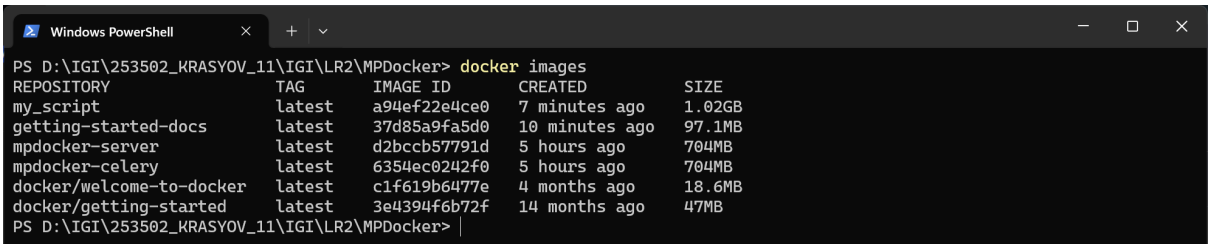
Docker-compose file:

```yaml
services:
  nginx:
    image: nginx:1.25-alpine
    restart: always
    volumes:
      - ./nginx/default.conf:/etc/nginx/conf.d/default.conf
      - ./nginx/mefodiy.net.crt:/etc/nginx/ssl/mefodiy.net.crt
      - ./nginx/mefodiy.net.key:/etc/nginx/ssl/mefodiy.net.key
      - ./nginx/ca_bundle.crt:/etc/nginx/ssl/ca_bundle.crt
      - media:/var/www/html/media
      - ./frontend/build/:/var/www/html
    depends_on:
      - server
    ports:
      - 80:80
  server:
    restart: unless-stopped
    network_mode: host
    build:
      context: ./backends/
      dockerfile: Dockerfile
    environment:
      OPENAI_API_KEY: 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
      YOOKASSA_API: https://api.yookassa.ru/v3/
      SQL_ENGINE: django.db.backends.postgresql
      SQL_DATABASE: postgres
      SQL_HOST: localhost
      SQL_USER: postgres
      SQL_PASSWORD: postgres
      SQL_PORT: 5432

    entrypoint: ./api.sh
    volumes:
      - media:/home/app/media
  celery:
    restart: always
    network_mode: host
    build:
      context: ./backends/
      dockerfile: Dockerfile
    entrypoint: ./celery.sh
    volumes:
      - media:/home/app/media
      - /var/run/docker.sock:/var/run/docker.sock
    environment:
      OPENAI_API_KEY: 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
      YOOKASSA_API: https://api.yookassa.ru/v3/
      SQL_ENGINE: django.db.backends.postgresql
      SQL_DATABASE: postgres
      SQL_HOST: localhost
      SQL_USER: postgres
      SQL_PASSWORD: postgres
      SQL_PORT: 5432
    depends_on:
      - server
  postgres:
    image: postgres
    environment:
      POSTGRES_DB: postgres
      POSTGRES_USER: postgres
      POSTGRES_PASSWORD: postgres
    ports:
      - "5432:5432"
  redis:
    image: "redis:latest"
    restart: always
    network_mode: host
    ports:
      - "6379:6379"
volumes:
  media: {}
```

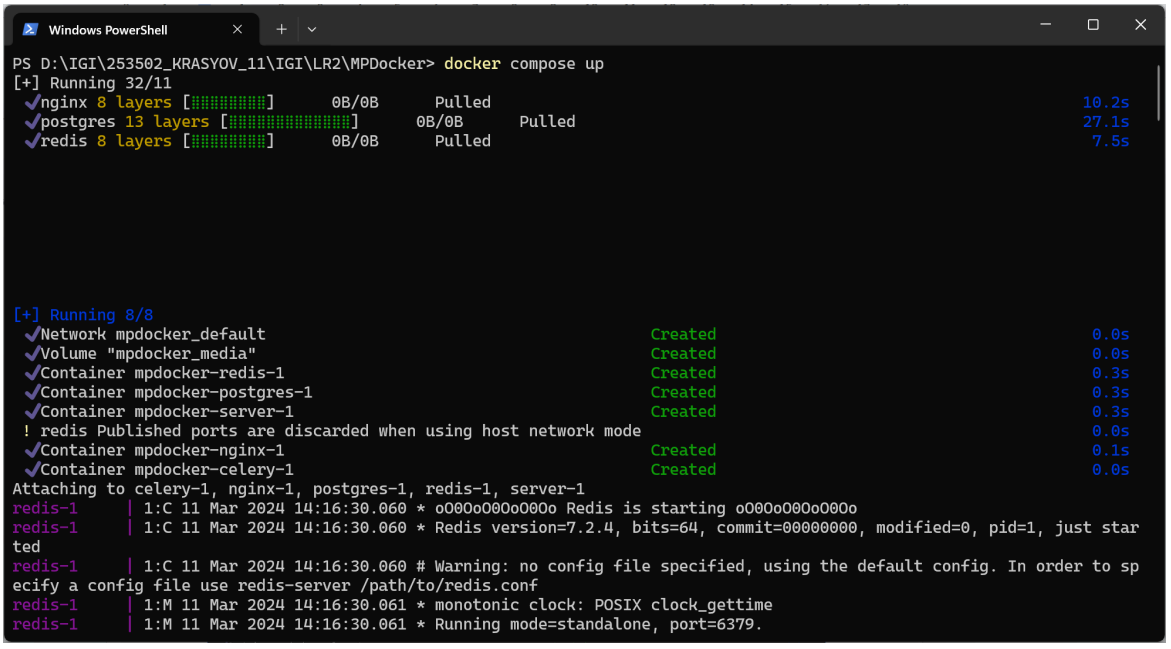
Выполним сборку образов и запустим приложение



```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\MPDocker> docker compose build
[+] Building 0.0s (0/0)  docker:default
2024/03/10 23:22:34 http2: server: error reading preface from client //./pipe/docker_engine: file has already been close
[+] Building 63.1s (16/17)                                                                              docker:default
 => => transferring context: 2B                                                                                  0.0s
[+] Building 63.4s (17/17)                                                                              docker:default
 => => resolve docker.io/library/python:3.9-slim@sha256:e0bc011bb55918109921b913fe30160cb8297c570621a450477d44999  0.0s
[+] Building 64.1s (27/27) FINISHED                                                                     docker:default
 => [server internal] load build definition from Dockerfile                                                      0.0s
 => => transferring dockerfile: 563B                                                                             0.0s
 => [celery internal] load metadata for docker.io/library/python:3.9-slim                                        2.9s
 => [server auth] library/python:pull token for registry-1.docker.io                                             0.0s
 => [server internal] load .dockerignore                                                                         0.0s
 => => transferring context: 2B                                                                                  0.0s
 => [celery  1/11] FROM docker.io/library/python:3.9-slim@sha256:e0bc011bb55918109921b913fe30160cb8297c570621a450  3.0s
 => => resolve docker.io/library/python:3.9-slim@sha256:e0bc011bb55918109921b913fe30160cb8297c570621a450477d44999  0.0s
 => => sha256:51d1f07906b71fd60ac43c61035514996a8ad8dbfd39d4f570ac5446b064ee5d 3.51MB / 3.51MB                   0.9s
 => => sha256:336c7f590cb97722bfee12f22e354df879feae9f28bfd5cebeaffccb3fb8fbf5 11.89MB / 11.89MB                 2.4s
 => => sha256:93b25b5c998e137849343851763bbc686f369399894e766c37a55304b1f66cfb 244B / 244B                       0.7s
 => => sha256:e0bc011bb55918109921b913fe30160cb8297c570621a450477d44999a792beb 1.86kB / 1.86kB                   0.0s
 => => sha256:51c781cd11dd1f2a95e2bef833a5920042743fa502d66c9e12c1a841d983f9a7 1.37kB / 1.37kB                   0.0s
 => => sha256:9c14a9ca10408336092a7089469ef9a84f3caf196f7f65c512677e1f06e5d839 6.92kB / 6.92kB                   0.0s
 => => sha256:2b527dfdb0a9ecb9a5bf5c264aeb4ac254e0752886f187095f6960b933aab941 3.13MB / 3.13MB                   2.2s
 => => extracting sha256:51d1f07906b71fd60ac43c61035514996a8ad8dbfd39d4f570ac5446b064ee5d                         0.1s
 => => extracting sha256:336c7f590cb97722bfee12f22e354df879feae9f28bfd5cebeaffccb3fb8fbf5                         0.3s
 => => extracting sha256:93b25b5c998e137849343851763bbc686f369399894e766c37a55304b1f66cfb                        0.0s
 => => extracting sha256:2b527dfdb0a9ecb9a5bf5c264aeb4ac254e0752886f187095f6960b933aab941                        0.2s
 => [server internal] load build context                                                                         0.0s
 => => transferring context: 57.28kB                                                                             0.0s
 => CACHED [celery  2/11] RUN apt-get update && apt-get install -y libpq-dev                                     4.8s
```

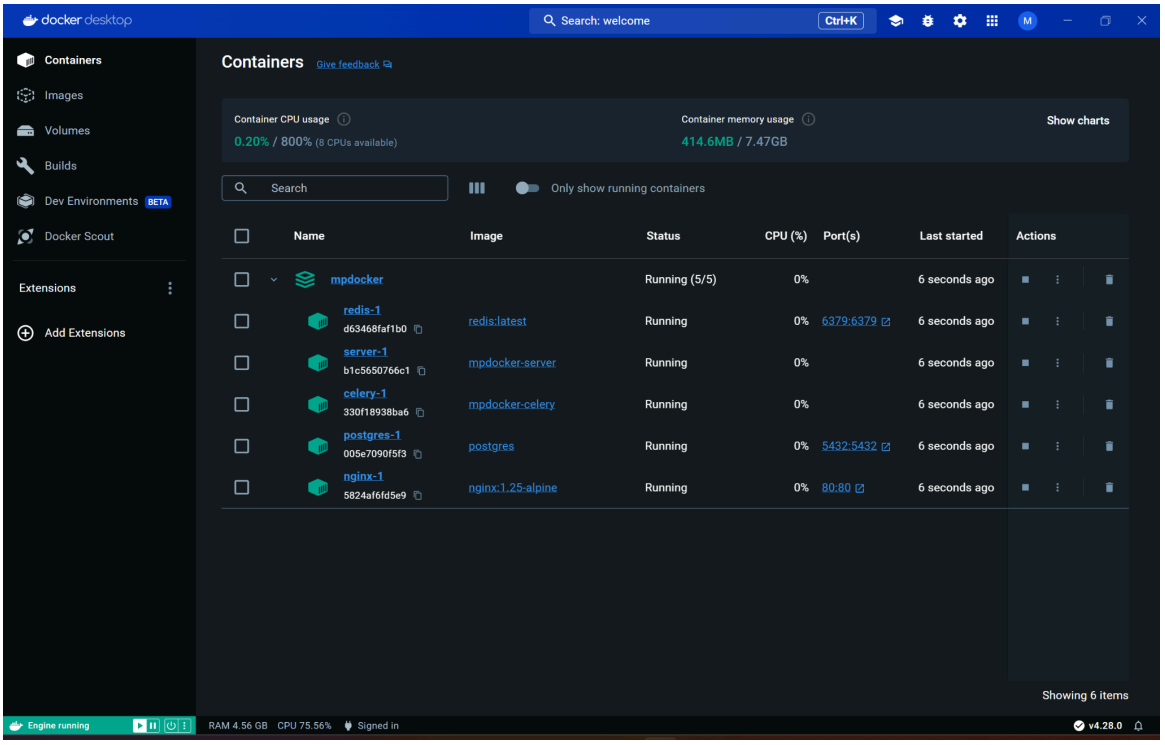


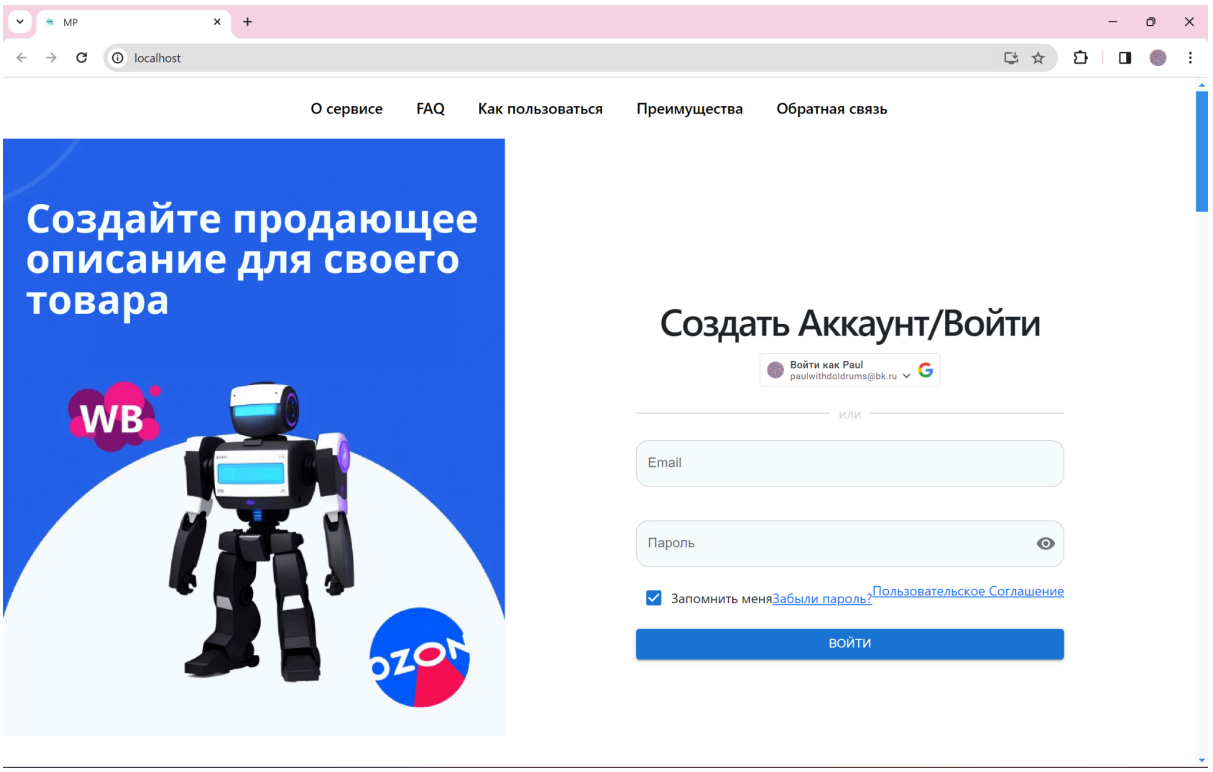
```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\MPDocker> docker images
REPOSITORY                TAG       IMAGE ID       CREATED          SIZE
my_script                 latest    a94ef22e4ce0   7 minutes ago    1.02GB
getting-started-docs      latest    37d85a9fa5d0   10 minutes ago   97.1MB
mpdocker-server           latest    d2bccb57791d   5 hours ago      704MB
mpdocker-celery           latest    6354ec0242f0   5 hours ago      704MB
docker/welcome-to-docker  latest    c1f619b6477e   4 months ago     18.6MB
docker/getting-started    latest    3e4394f6b72f   14 months ago    47MB
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\MPDocker>
```

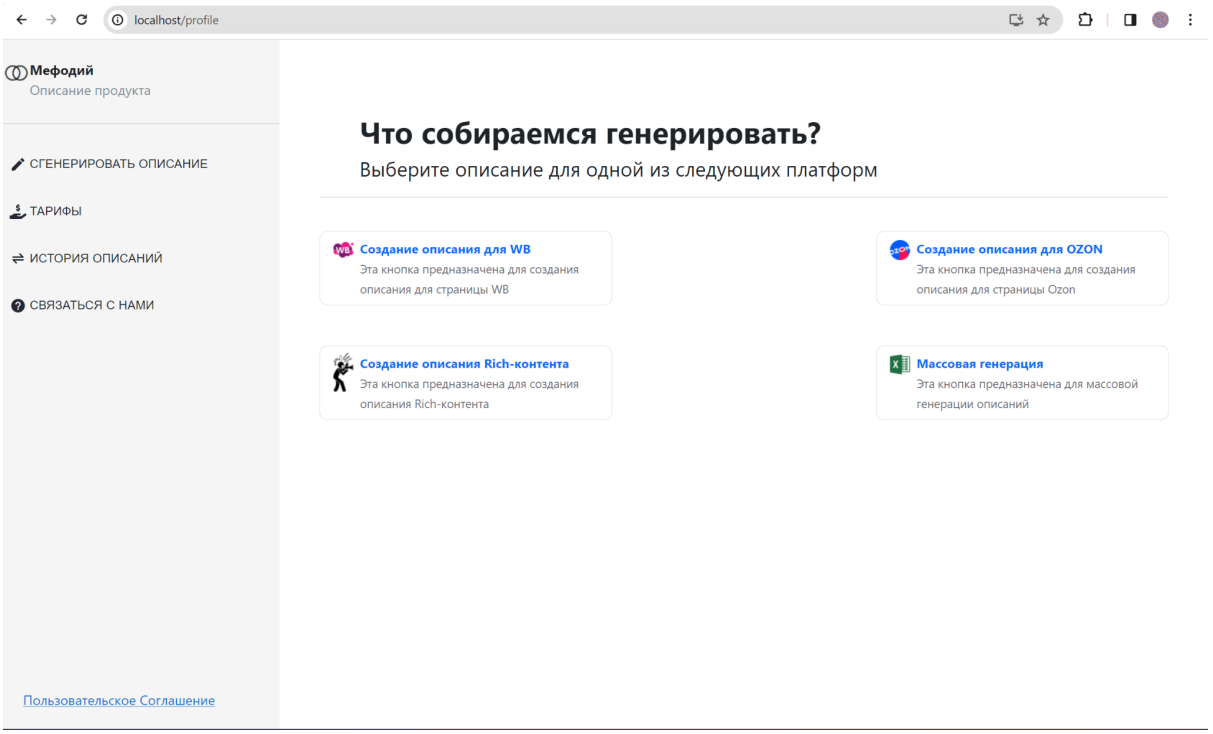


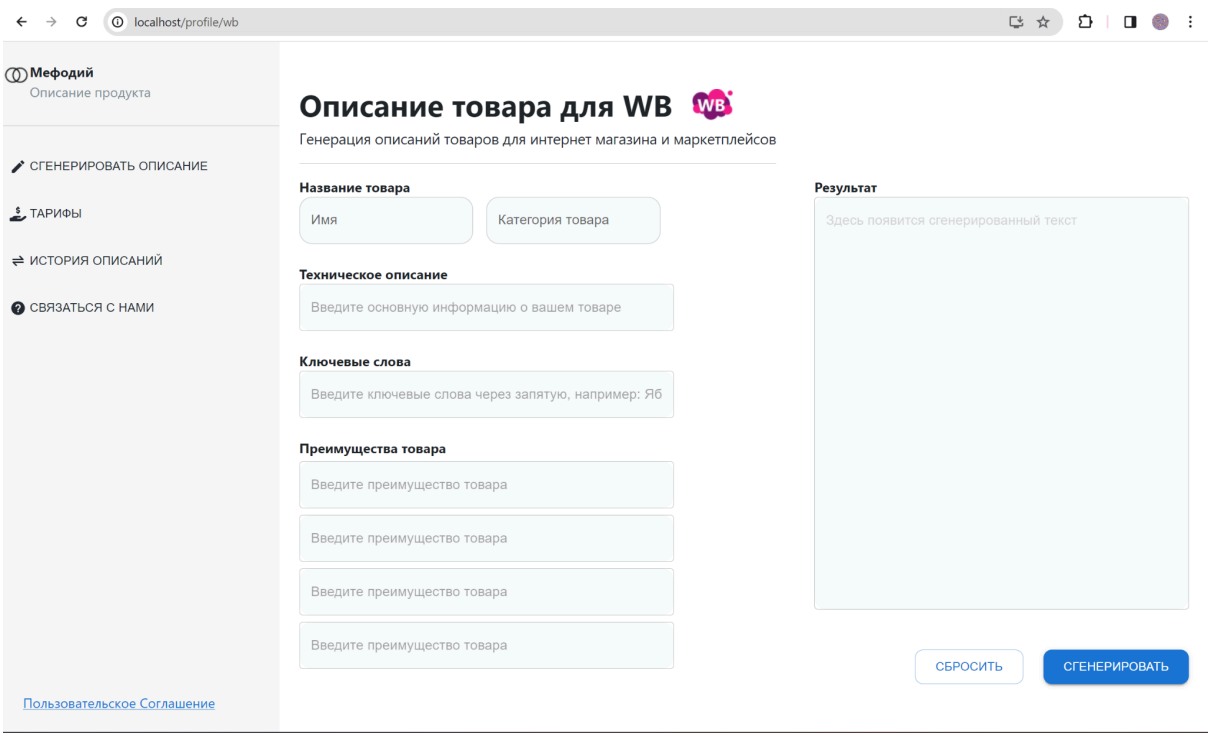
```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\MPDocker> docker compose up
[+] Running 32/11
 ✔nginx 8 layers [##########]      0B/0B      Pulled                                                      10.2s
 ✔postgres 13 layers [##############]      0B/0B      Pulled                                             27.1s
 ✔redis 8 layers [##########]      0B/0B      Pulled                                                      7.5s

[+] Running 8/8
 ✔Network mpdocker_default                                  Created                                      0.0s
 ✔Volume "mpdocker_media"                                   Created                                      0.0s
 ✔Container mpdocker-redis-1                                Created                                      0.3s
 ✔Container mpdocker-postgres-1                             Created                                      0.3s
 ✔Container mpdocker-server-1                               Created                                      0.3s
 ! redis Published ports are discarded when using host network mode                                      0.0s
 ✔Container mpdocker-nginx-1                                Created                                      0.1s
 ✔Container mpdocker-celery-1                               Created                                      0.0s
Attaching to celery-1, nginx-1, postgres-1, redis-1, server-1
redis-1     | 1:C 11 Mar 2024 14:16:30.060 * oO0OoO0OoO0Oo Redis is starting oO0OoO0OoO0Oo
redis-1     | 1:C 11 Mar 2024 14:16:30.060 * Redis version=7.2.4, bits=64, commit=00000000, modified=0, pid=1, just star
ted
redis-1     | 1:C 11 Mar 2024 14:16:30.060 # Warning: no config file specified, using the default config. In order to sp
ecify a config file use redis-server /path/to/redis.conf
redis-1     | 1:M 11 Mar 2024 14:16:30.061 * monotonic clock: POSIX clock_gettime
redis-1     | 1:M 11 Mar 2024 14:16:30.061 * Running mode=standalone, port=6379.
```

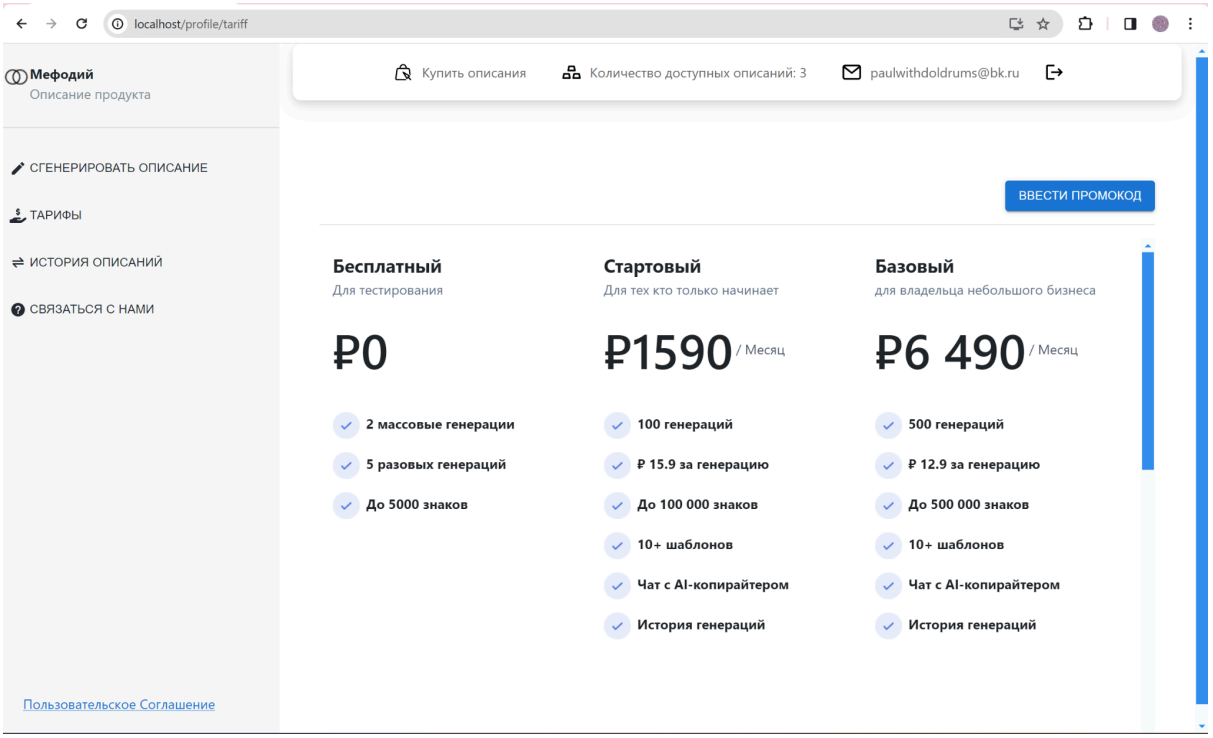
Проверим его работоспособность





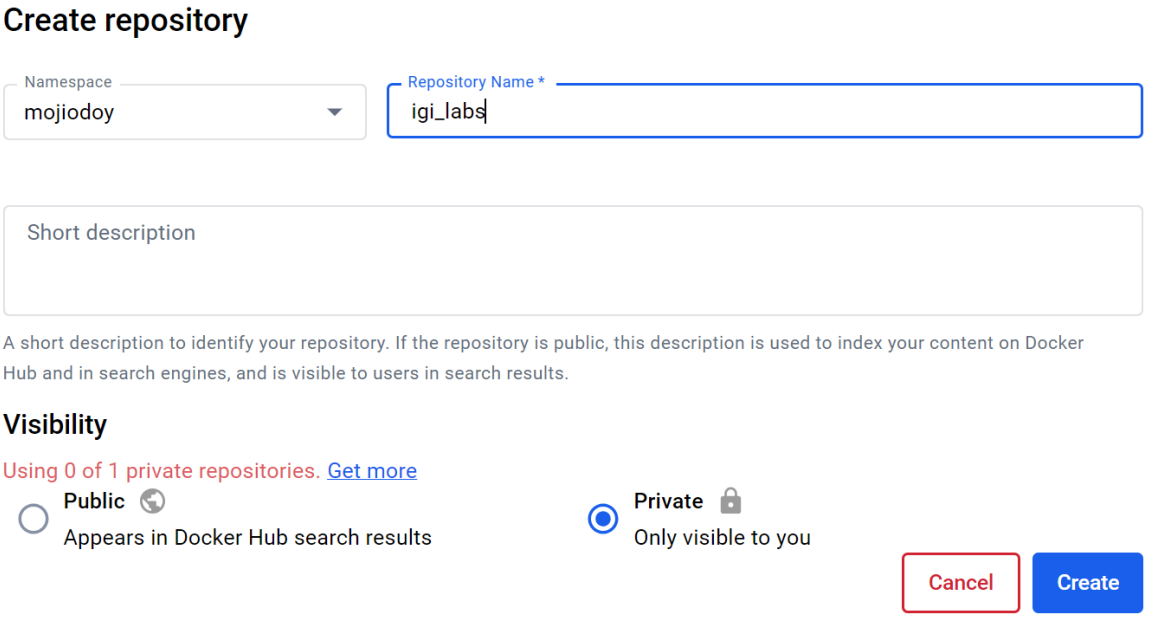

localhost/profile
Мефодий
Описание продукта
СГЕНЕРИРОВАТЬ ОПИСАНИЕ
ТАРИФЫ
ИСТОРИЯ ОПИСАНИЙ
СВЯЗАТЬСЯ С НАМИ
Что собираемся генерировать?
Выберите описание для одной из следующих платформ
Создание описания для WB
Эта кнопка предназначена для создания описания для страницы WB
Создание описания для OZON
Эта кнопка предназначена для создания описания для страницы Ozon
Создание описания Rich-контента
Эта кнопка предназначена для создания описания Rich-контента
Массовая генерация
Эта кнопка предназначена для массовой генерации описаний
Пользовательское Соглашение


localhost/profile/wb
Мефодий
Описание продукта
СГЕНЕРИРОВАТЬ ОПИСАНИЕ
ТАРИФЫ
ИСТОРИЯ ОПИСАНИЙ
СВЯЗАТЬСЯ С НАМИ
Описание товара для WB
Генерация описаний товаров для интернет магазина и маркетплейсов
Название товара
Имя
Категория товара
Техническое описание
Введите основную информацию о вашем товаре
Ключевые слова
Введите ключевые слова через запятую, например: Яб
Преимущества товара
Введите преимущество товара
Введите преимущество товара
Введите преимущество товара
Введите преимущество товара
Результат
Здесь появится сгенерированный текст
СБРОСИТЬ
СГЕНЕРИРОВАТЬ
Пользовательское Соглашение

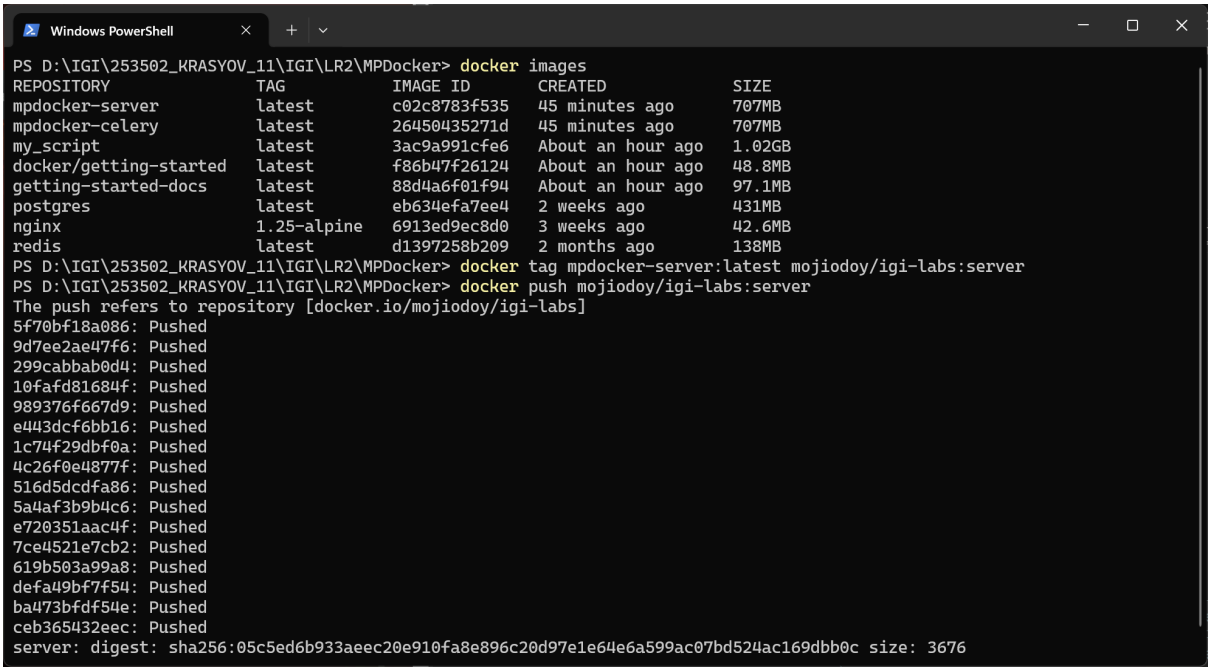


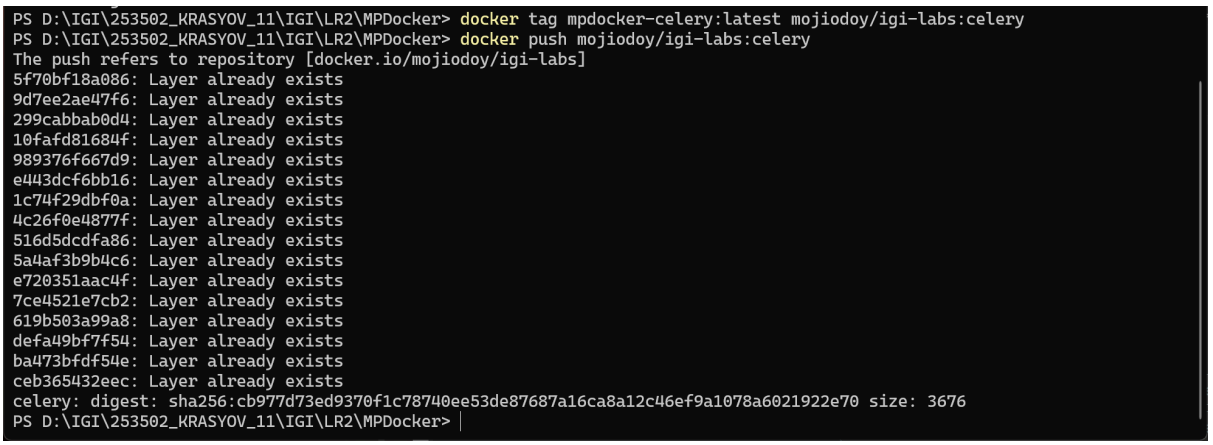
Задание 6. Разместите результат в созданный репозиторий в DockerHub

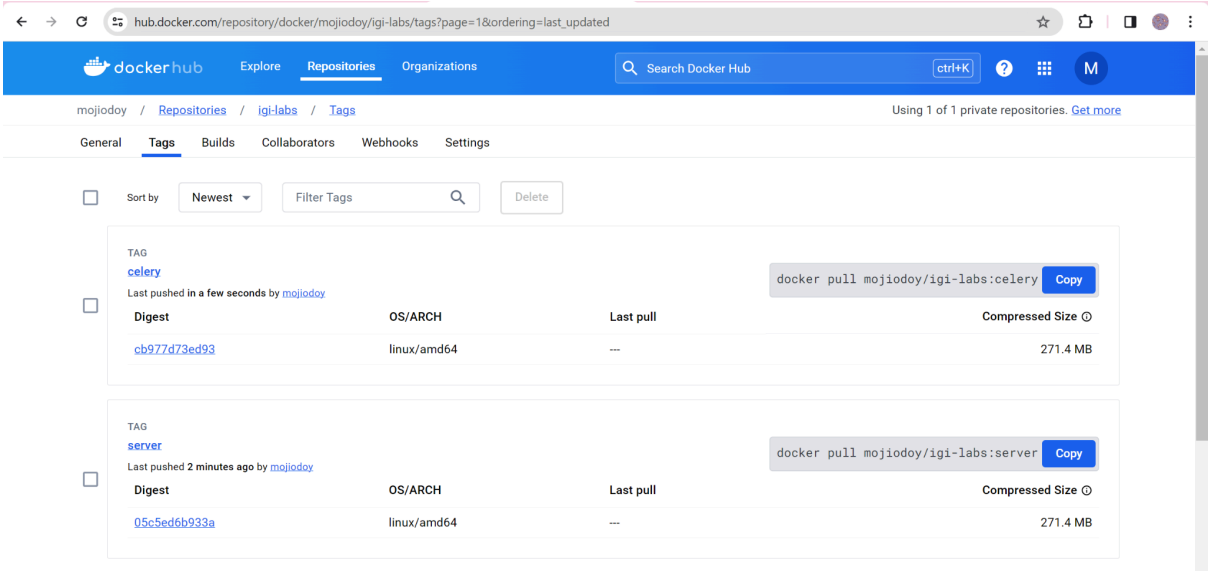
Создадим репозиторий



Сохраним образы в репозитории:



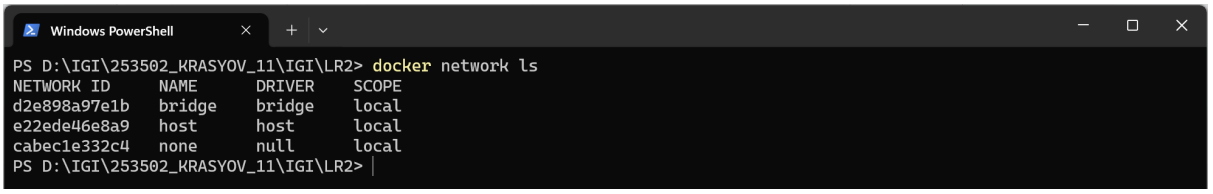




Задание 7. Выполните следующие действия с целью изучить особенности сетевого взаимодействия:

- 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 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.

Получим сведения о всех сетях:



Теперь получим более подробную информацию о каждом типе сетей:
Bridge:

```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network ls
NETWORK ID     NAME      DRIVER    SCOPE
d2e898a97e1b   bridge    bridge    local
e22ede46e8a9   host      host      local
cabec1e332c4   none      null      local
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network inspect bridge
[
    {
        "Name": "bridge",
        "Id": "d2e898a97e1ba4b182cfc5a0b1082f2b03265e1efd95f4b739a81cbb2962a813",
        "Created": "2024-03-11T16:10:34.197789981Z",
        "Scope": "local",
        "Driver": "bridge",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": null,
            "Config": [
                {
                    "Subnet": "172.17.0.0/16",
                    "Gateway": "172.17.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {},
        "Options": {
            "com.docker.network.bridge.default_bridge": "true",
            "com.docker.network.bridge.enable_icc": "true",
            "com.docker.network.bridge.enable_ip_masquerade": "true",
            "com.docker.network.bridge.host_binding_ipv4": "0.0.0.0",
            "com.docker.network.bridge.name": "docker0",
            "com.docker.network.driver.mtu": "1500"
        },
        "Labels": {}
    }
]
PS D:\IGI\253502_KRASYOV_11\IGI\LR2>
```

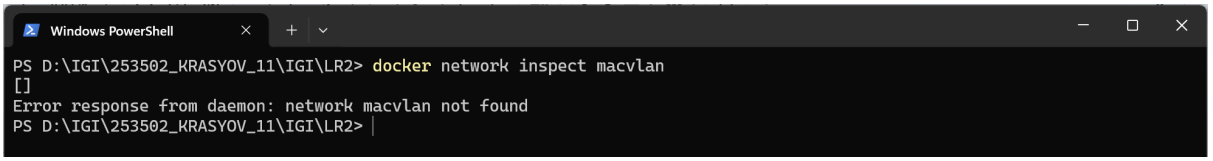
Host:

```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network inspect host
[
    {
        "Name": "host",
        "Id": "e22ede46e8a95b8e66ec29a13059f8fe6effa9ea528d4b2294cdf27568513710",
        "Created": "2024-03-11T16:00:44.244368722Z",
        "Scope": "local",
        "Driver": "host",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": null,
            "Config": null
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {},
        "Options": {},
        "Labels": {}
    }
]
PS D:\IGI\253502_KRASYOV_11\IGI\LR2>
```

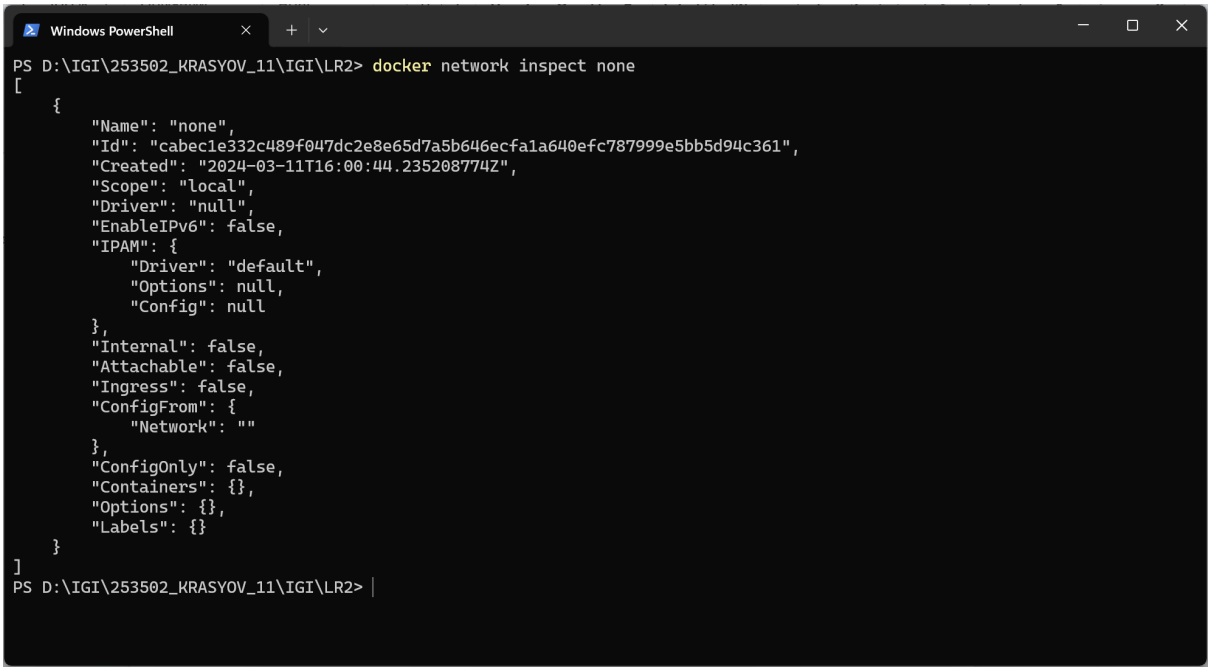
Overlay:

```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network inspect overlay
[]
Error response from daemon: network overlay not found
PS D:\IGI\253502_KRASYOV_11\IGI\LR2>
```

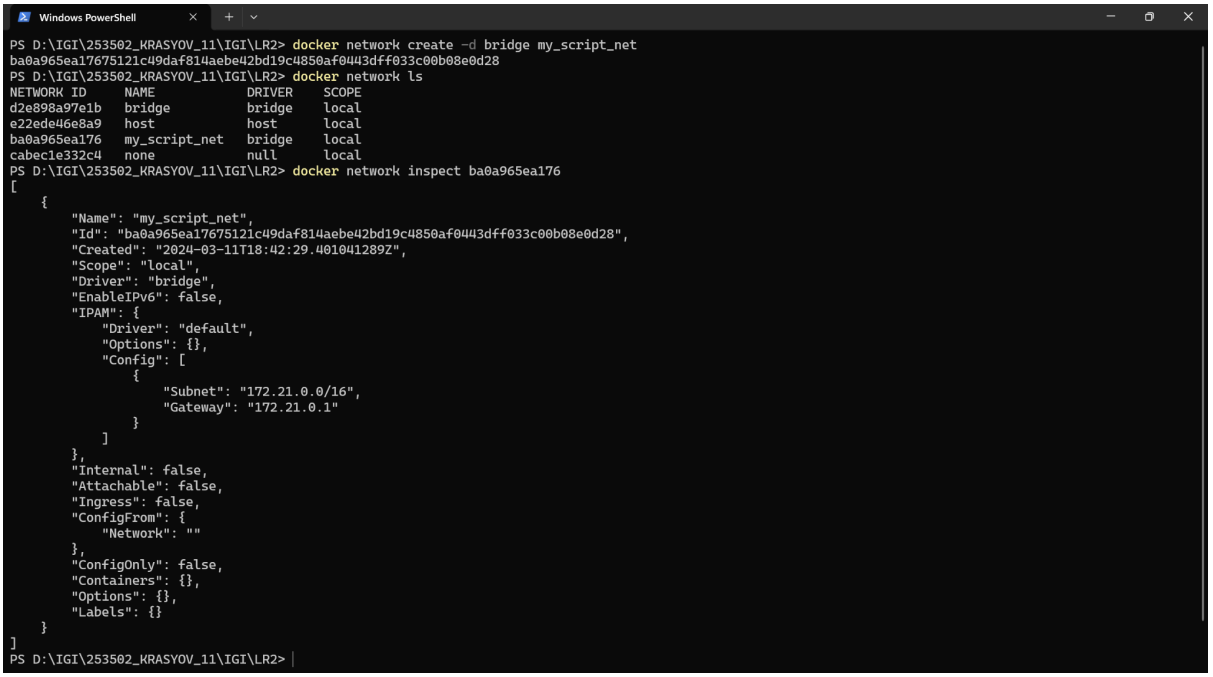
Macvlan:



None:



Создадим новую сеть типа bridge:



Запустим контейнер в этой сети:

```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker images
REPOSITORY                TAG           IMAGE ID       CREATED             SIZE
mpdocker-celery           latest        26450435271d   About an hour ago   707MB
mojiodoy/igi-labs         celery        26450435271d   About an hour ago   707MB
mojiodoy/igi-labs         server        c02c8783f535   About an hour ago   707MB
mpdocker-server           latest        c02c8783f535   About an hour ago   707MB
my_script                 latest        3ac9a991cfe6   2 hours ago         1.02GB
docker/getting-started    latest        f86b47f26124   2 hours ago         48.8MB
getting-started-docs      latest        88d4a6f01f94   2 hours ago         97.1MB
postgres                  latest        eb634efa7ee4   2 weeks ago         431MB
nginx                     1.25-alpine   6913ed9ec8d0   3 weeks ago         42.6MB
httpd                     latest        2776f4da9d55   7 weeks ago         167MB
redis                     latest        d1397258b209   2 months ago        138MB
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker run -it -d -p 5000:5000 --network=my_script_net httpd
130a60bebea443cc4f477a30ef05c2ebb2752fff6241f56de6a2aa81a8a5ff41
PS D:\IGI\253502_KRASYOV_11\IGI\LR2>
```

Просмотрим информацию о контейнере:

```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker ps
CONTAINER ID   IMAGE     COMMAND              CREATED             STATUS             PORTS                                       NAMES
130a60bebea4   httpd     "httpd-foreground"   About a minute ago   Up About a minute   80/tcp, 0.0.0.0:5000->5000/tcp   trusting_nash
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker inspect 130a60bebea4
```

```
Windows PowerShell
        "ExecIDs": null,
        "HostConfig": {
            "Binds": null,
            "ContainerIDFile": "",
            "LogConfig": {
                "Type": "json-file",
                "Config": {}
            },
            "NetworkMode": "my_script_net",
            "PortBindings": {
                "5000/tcp": [
                    {
                        "HostIp": "",
                        "HostPort": "5000"
                    }
                ]
            },
            "RestartPolicy": {
                "Name": "no",
                "MaximumRetryCount": 0
            },
            "AutoRemove": false,
            "VolumeDriver": "",
            "VolumesFrom": null,
            "ConsoleSize": [
                30,
                120
            ],
```

Просмотрим информацию о нашей сети:

```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network inspect my_script_net
[
    {
        "Name": "my_script_net",
        "Id": "ba0a965ea17675121c49daf814aebe42bd19c4850af0443dff033c00b08e0d28",
        "Created": "2024-03-11T18:42:29.401041289Z",
        "Scope": "local",
        "Driver": "bridge",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": {},
            "Config": [
                {
                    "Subnet": "172.21.0.0/16",
                    "Gateway": "172.21.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {
            "130a60bebea443cc4f477a30ef05c2ebb2752fff6241f56de6a2aa81a8a5ff41": {
                "Name": "trusting_nash",
                "EndpointID": "740c1e285d8f20ba44394f6dd3917bcb20d5993af6e49f3e23483fd331f94c55",
                "MacAddress": "02:42:ac:15:00:02",
                "IPv4Address": "172.21.0.2/16",
                "IPv6Address": ""
            }
        },
        "Options": {},
        "Labels": {}
    }
]
PS D:\IGI\253502_KRASYOV_11\IGI\LR2>
```

Отключим контейнер от сети и просмотрим информацию о ней:

```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network disconnect my_script_net 130a60bebea443cc4f477a30ef05c2ebb2752fff6241f56de6a2aa81a8a5ff41
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network inspect my_script_net
[
    {
        "Name": "my_script_net",
        "Id": "ba0a965ea17675121c49daf814aebe42bd19c4850af0443dff033c00b08e0d28",
        "Created": "2024-03-11T18:42:29.401041289Z",
        "Scope": "local",
        "Driver": "bridge",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": {},
            "Config": [
                {
                    "Subnet": "172.21.0.0/16",
                    "Gateway": "172.21.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {},
        "Options": {},
        "Labels": {}
    }
]
PS D:\IGI\253502_KRASYOV_11\IGI\LR2>
```

Создадим новую bridge сеть и подключим в неё наши образы:

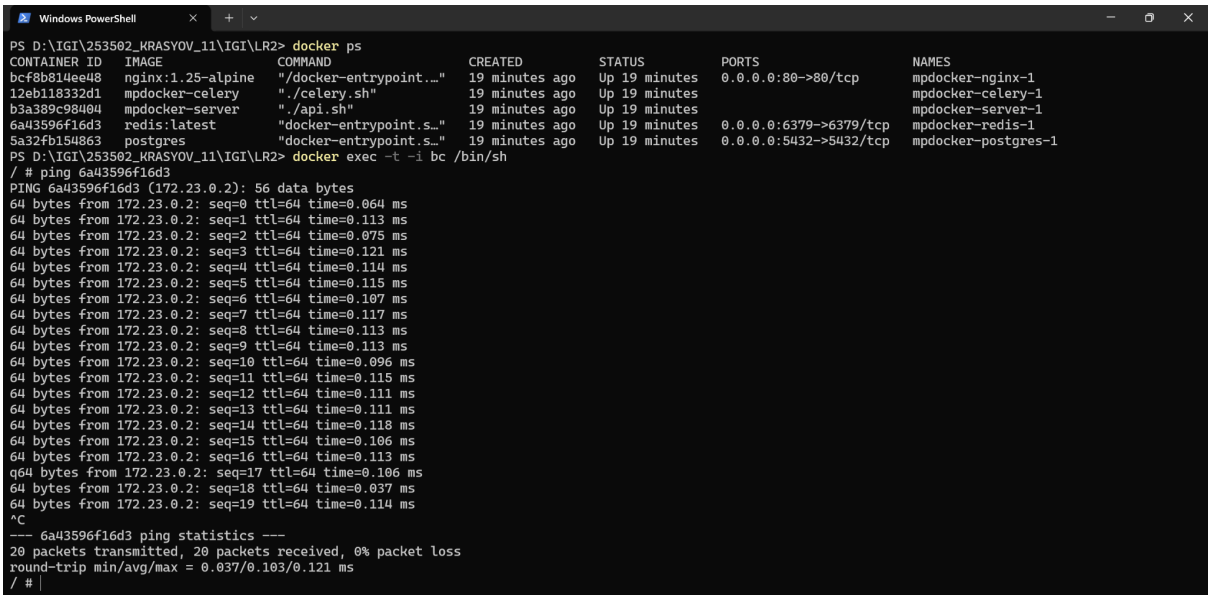
```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network create -d bridge new_big_net
7a693e790cd174a5f95beadd614f5155111254320eb18c4433e4ea890efae86f
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network ls
NETWORK ID     NAME          DRIVER    SCOPE
d2e898a97e1b   bridge        bridge    local
e22ede46e8a9   host          host      local
ba0a965ea176   my_script_net bridge    local
7a693e790cd1   new_big_net   bridge    local
cabec1e332c4   none          null      local
PS D:\IGI\253502_KRASYOV_11\IGI\LR2>
```

```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker ps
CONTAINER ID   IMAGE              COMMAND                  CREATED          STATUS          PORTS                    NAMES
bcf8b814ee48   nginx:1.25-alpine  "/docker-entrypoint.…"   16 seconds ago   Up 14 seconds   0.0.0.0:80->80/tcp       mpdocker-nginx-1
12eb118332d1   mpdocker-celery    "./celery.sh"            16 seconds ago   Up 15 seconds                            mpdocker-celery-1
b3a389c98404   mpdocker-server    "./api.sh"               16 seconds ago   Up 15 seconds                            mpdocker-server-1
6a43596f16d3   redis:latest       "docker-entrypoint.s…"   16 seconds ago   Up 14 seconds   0.0.0.0:6379->6379/tcp   mpdocker-redis-1
5a32fb154863   postgres           "docker-entrypoint.s…"   16 seconds ago   Up 14 seconds   0.0.0.0:5432->5432/tcp   mpdocker-postgres-1
```

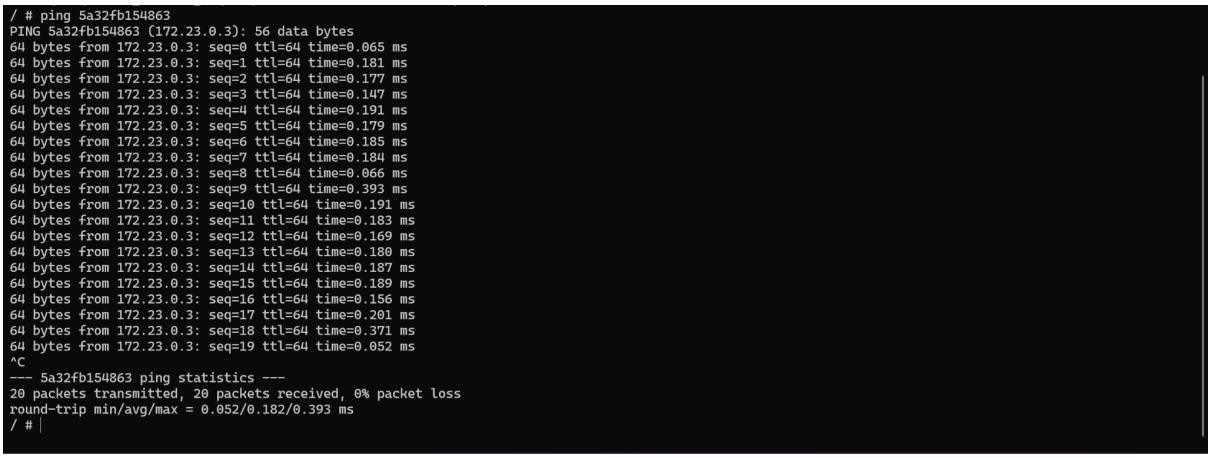
```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network connect new_big_net bcf
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network connect new_big_net 6a4
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network connect new_big_net 5a32
```

```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network inspect new_big_net
[
    {
        "Name": "new_big_net",
        "Id": "7a693e790cd174a5f95beadd614f5155111254320eb18c4433e4ea890efae86f",
        "Created": "2024-03-11T19:02:41.208375252Z",
        "Scope": "local",
        "Driver": "bridge",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": {},
            "Config": [
                {
                    "Subnet": "172.22.0.0/16",
                    "Gateway": "172.22.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {
            "5a32fb1548634e3b09799d394ef895a874037a8da527733ff6583ec8694a67c9": {
                "Name": "mpdocker-postgres-1",
                "EndpointID": "17bedeb40e6422bdb98ab4385f15a3228e5a5bacaaf7e19e0259543c87bf7389",
                "MacAddress": "02:42:ac:16:00:04",
                "IPv4Address": "172.22.0.4/16",
                "IPv6Address": ""
            },
            "6a43596f16d33f5ba1cfa512ad04d785d48ae607927c8782a068721fb7ef0fbe": {
                "Name": "mpdocker-redis-1",
                "EndpointID": "e29e0e66cad37525927afe6bf8d3603ac0036ae2c4e82fe15f988c7cc6353277",
                "MacAddress": "02:42:ac:16:00:03",
                "IPv4Address": "172.22.0.3/16",
                "IPv6Address": ""
            },
            "bcf8b814ee488ce1db1586318fe220f74b03d05715abe9912ba0ca4d51b39101": {
                "Name": "mpdocker-nginx-1",
                "EndpointID": "0af9cfb6a2589f0b07187cd539c303611716e4782805fefbe631dae8c8402e0e",
                "MacAddress": "02:42:ac:16:00:02",
```

Зайдём в оболочку одного контейнера и пропингуем 2 остальных:

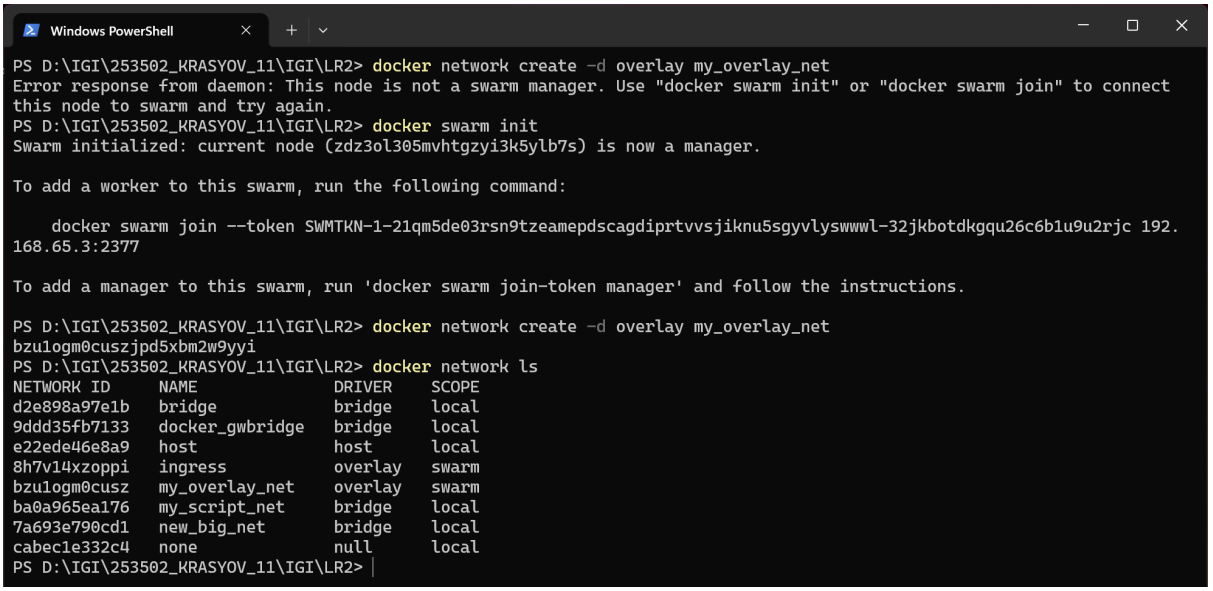


```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker ps
CONTAINER ID   IMAGE              COMMAND                  CREATED          STATUS          PORTS                    NAMES
bcf8b814ee48   nginx:1.25-alpine  "/docker-entrypoint.…"   19 minutes ago   Up 19 minutes   0.0.0.0:80->80/tcp       mpdocker-nginx-1
12eb118332d1   mpdocker-celery    "./celery.sh"            19 minutes ago   Up 19 minutes                            mpdocker-celery-1
b3a389c98404   mpdocker-server    "./api.sh"               19 minutes ago   Up 19 minutes                            mpdocker-server-1
6a43596f16d3   redis:latest       "docker-entrypoint.s…"   19 minutes ago   Up 19 minutes   0.0.0.0:6379->6379/tcp   mpdocker-redis-1
5a32fb154863   postgres           "docker-entrypoint.s…"   19 minutes ago   Up 19 minutes   0.0.0.0:5432->5432/tcp   mpdocker-postgres-1
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker exec -t -i bc /bin/sh
/ # ping 6a43596f16d3
PING 6a43596f16d3 (172.23.0.2): 56 data bytes
64 bytes from 172.23.0.2: seq=0 ttl=64 time=0.064 ms
64 bytes from 172.23.0.2: seq=1 ttl=64 time=0.113 ms
64 bytes from 172.23.0.2: seq=2 ttl=64 time=0.075 ms
64 bytes from 172.23.0.2: seq=3 ttl=64 time=0.121 ms
64 bytes from 172.23.0.2: seq=4 ttl=64 time=0.114 ms
64 bytes from 172.23.0.2: seq=5 ttl=64 time=0.115 ms
64 bytes from 172.23.0.2: seq=6 ttl=64 time=0.107 ms
64 bytes from 172.23.0.2: seq=7 ttl=64 time=0.117 ms
64 bytes from 172.23.0.2: seq=8 ttl=64 time=0.113 ms
64 bytes from 172.23.0.2: seq=9 ttl=64 time=0.113 ms
64 bytes from 172.23.0.2: seq=10 ttl=64 time=0.096 ms
64 bytes from 172.23.0.2: seq=11 ttl=64 time=0.115 ms
64 bytes from 172.23.0.2: seq=12 ttl=64 time=0.111 ms
64 bytes from 172.23.0.2: seq=13 ttl=64 time=0.111 ms
64 bytes from 172.23.0.2: seq=14 ttl=64 time=0.118 ms
64 bytes from 172.23.0.2: seq=15 ttl=64 time=0.106 ms
64 bytes from 172.23.0.2: seq=16 ttl=64 time=0.113 ms
q64 bytes from 172.23.0.2: seq=17 ttl=64 time=0.106 ms
64 bytes from 172.23.0.2: seq=18 ttl=64 time=0.037 ms
64 bytes from 172.23.0.2: seq=19 ttl=64 time=0.114 ms
^C
--- 6a43596f16d3 ping statistics ---
20 packets transmitted, 20 packets received, 0% packet loss
round-trip min/avg/max = 0.037/0.103/0.121 ms
/ #
```



```
/ # ping 5a32fb154863
PING 5a32fb154863 (172.23.0.3): 56 data bytes
64 bytes from 172.23.0.3: seq=0 ttl=64 time=0.065 ms
64 bytes from 172.23.0.3: seq=1 ttl=64 time=0.181 ms
64 bytes from 172.23.0.3: seq=2 ttl=64 time=0.177 ms
64 bytes from 172.23.0.3: seq=3 ttl=64 time=0.147 ms
64 bytes from 172.23.0.3: seq=4 ttl=64 time=0.191 ms
64 bytes from 172.23.0.3: seq=5 ttl=64 time=0.179 ms
64 bytes from 172.23.0.3: seq=6 ttl=64 time=0.185 ms
64 bytes from 172.23.0.3: seq=7 ttl=64 time=0.184 ms
64 bytes from 172.23.0.3: seq=8 ttl=64 time=0.066 ms
64 bytes from 172.23.0.3: seq=9 ttl=64 time=0.393 ms
64 bytes from 172.23.0.3: seq=10 ttl=64 time=0.191 ms
64 bytes from 172.23.0.3: seq=11 ttl=64 time=0.183 ms
64 bytes from 172.23.0.3: seq=12 ttl=64 time=0.169 ms
64 bytes from 172.23.0.3: seq=13 ttl=64 time=0.180 ms
64 bytes from 172.23.0.3: seq=14 ttl=64 time=0.187 ms
64 bytes from 172.23.0.3: seq=15 ttl=64 time=0.189 ms
64 bytes from 172.23.0.3: seq=16 ttl=64 time=0.156 ms
64 bytes from 172.23.0.3: seq=17 ttl=64 time=0.201 ms
64 bytes from 172.23.0.3: seq=18 ttl=64 time=0.371 ms
64 bytes from 172.23.0.3: seq=19 ttl=64 time=0.052 ms
^C
--- 5a32fb154863 ping statistics ---
20 packets transmitted, 20 packets received, 0% packet loss
round-trip min/avg/max = 0.052/0.182/0.393 ms
/ #
```

Создадим overlay сеть



```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network create -d overlay my_overlay_net
Error response from daemon: This node is not a swarm manager. Use "docker swarm init" or "docker swarm join" to connect
this node to swarm and try again.
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker swarm init
Swarm initialized: current node (zdz3ol305mvhtgzyi3k5ylb7s) is now a manager.

To add a worker to this swarm, run the following command:

    docker swarm join --token SWMTKN-1-21qm5de03rsn9tzeamepdscagdiprtvvsjiknu5sgyvlyswwwl-32jkbotdkgqu26c6b1u9u2rjc 192.
168.65.3:2377

To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructions.

PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network create -d overlay my_overlay_net
bzu1ogm0cuszjpd5xbm2w9yyi
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network ls
NETWORK ID     NAME              DRIVER    SCOPE
d2e898a97e1b   bridge            bridge    local
9ddd35fb7133   docker_gwbridge   bridge    local
e22ede46e8a9   host              host      local
8h7v14xzoppi   ingress           overlay   swarm
bzu1ogm0cusz   my_overlay_net    overlay   swarm
ba0a965ea176   my_script_net     bridge    local
7a693e790cd1   new_big_net       bridge    local
cabec1e332c4   none              null      local
PS D:\IGI\253502_KRASYOV_11\IGI\LR2>
```

```
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network inspect my_overlay_net
[
    {
        "Name": "my_overlay_net",
        "Id": "bzu1ogm0cuszjpd5xbm2w9yyi",
        "Created": "2024-03-11T19:32:07.753713401Z",
        "Scope": "swarm",
        "Driver": "overlay",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": null,
            "Config": [
                {
                    "Subnet": "10.0.1.0/24",
                    "Gateway": "10.0.1.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": null,
        "Options": {
            "com.docker.network.driver.overlay.vxlanid_list": "4097"
        },
        "Labels": null
    }
]
PS D:\IGI\253502_KRASYOV_11\IGI\LR2>
```

Создадим ещё одну overlay сеть и удалим её

```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network create -d overlay overlay_to_delete
uk2xfowuwi7uf1pa02i6l1v2u
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network ls
NETWORK ID     NAME                DRIVER    SCOPE
d2e898a97e1b   bridge              bridge    local
9ddd35fb7133   docker_gwbridge     bridge    local
e22ede46e8a9   host                host      local
8h7v14xzoppi   ingress             overlay   swarm
bzu1ogm0cusz   my_overlay_net      overlay   swarm
ba0a965ea176   my_script_net       bridge    local
7a693e790cd1   new_big_net         bridge    local
cabec1e332c4   none                null      local
uk2xfowuwi7u   overlay_to_delete   overlay   swarm
PS D:\IGI\253502_KRASYOV_11\IGI\LR2>
```

```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network inspect overlay_to_delete
[
    {
        "Name": "overlay_to_delete",
        "Id": "uk2xfowuwi7uf1pa02i6l1v2u",
        "Created": "2024-03-11T19:34:22.033055964Z",
        "Scope": "swarm",
        "Driver": "overlay",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": null,
            "Config": [
                {
                    "Subnet": "10.0.2.0/24",
                    "Gateway": "10.0.2.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": null,
        "Options": {
            "com.docker.network.driver.overlay.vxlanid_list": "4098"
        },
        "Labels": null
    }
]
PS D:\IGI\253502_KRASYOV_11\IGI\LR2>
```

```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network rm overlay_to_delete
overlay_to_delete
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network ls
NETWORK ID     NAME              DRIVER    SCOPE
d2e898a97e1b   bridge            bridge    local
9ddd35fb7133   docker_gwbridge   bridge    local
e22ede46e8a9   host              host      local
8h7v14xzoppi   ingress           overlay   swarm
bzu1ogm0cusz   my_overlay_net    overlay   swarm
ba0a965ea176   my_script_net     bridge    local
7a693e790cd1   new_big_net       bridge    local
cabec1e332c4   none              null      local
PS D:\IGI\253502_KRASYOV_11\IGI\LR2>
```

Попробуем создать ещё одну сеть типа host, однако получаем ошибку, т.к может существовать только один экземпляр сети с типом host