

## ИГИ - ЛАБОРАТОРНАЯ РАБОТА 2

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

```
C:\Users\imsve>docker

Usage:  docker [OPTIONS] COMMAND

A self-sufficient runtime for containers

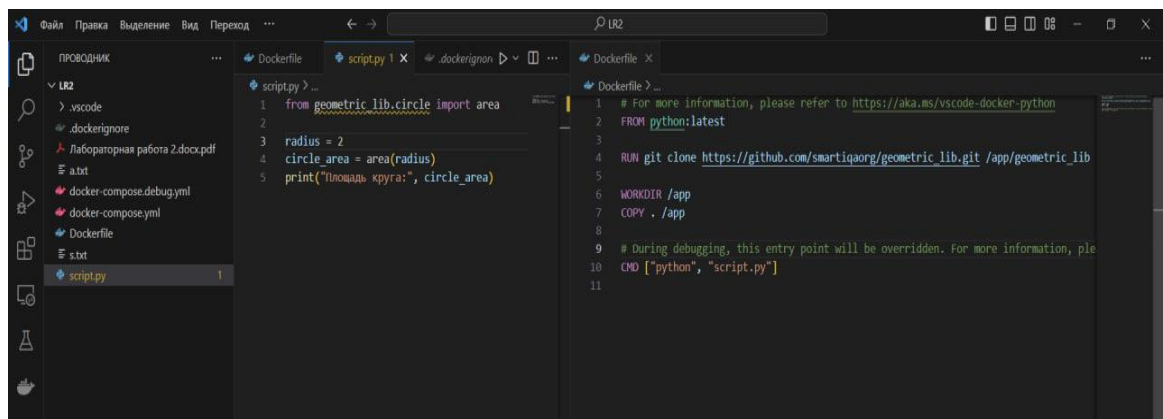
Common Commands:
run      Create and run a new container from an image
exec     Execute a command in a running container
ps       List containers
build    Build an image from a Dockerfile
pull     Download an image from a registry
push     Upload an image to a registry
images   List images
login    Log in to a registry
logout   Log out from a registry
search   Search Docker Hub for images
version  Show the Docker version information
info     Display system-wide information

Management Commands:
builder  Manage builds
buildx*  Docker Buildx (Docker Inc., v0.12.1-desktop.4)
checkpoint Manage checkpoints
compose* Docker Compose (Docker Inc., v2.24.5-desktop.1)
container Manage containers
```

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

```
C:\Users\imsve>docker run -d -p 80:80 docker/getting-started
53aa7010987116ef0cf2dcfbfd6214d09a6ff0b23151048c407a6e24dbeccafe
```

3. Создайте docker image, который запускает скрипт с использованием функций из [https://github.com/smartigaorg/geometric\\_lib](https://github.com/smartigaorg/geometric_lib). 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. Создать Dockerfile для реализации сборки собственных Docker образов. Использовать его для создания контейнера. Протестировать использование контейнера.



```
C:\253503_MINICH_17\igi\LR2>docker build -t my_script .
[+] Building 13.3s (9/9) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 419B
=> [internal] load metadata for docker.io/library/python:latest
=> [internal] load .dockerignore
=> => transferring context: 409B
=> [1/4] FROM docker.io/library/python:latest
=> [internal] load build context
=> => transferring context: 97.21kB
=> [2/4] RUN git clone https://github.com/smartigaorg/geometric_lib.git /app/geometric_lib
=> [3/4] WORKDIR /app
=> [4/4] COPY . /app
=> exporting to image
=> => exporting layers
=> => writing image sha256:e0d24f8d21f32b556c71ad9e16a1790e2d576f7f7dcea63d5de07790a0a160ac
=> => naming to docker.io/library/my_script

What's Next?
View a summary of image vulnerabilities and recommendations → docker scout quickview

C:\253503_MINICH_17\igi\LR2>docker run my_script
Площадь круга: 12.566370614359172
```

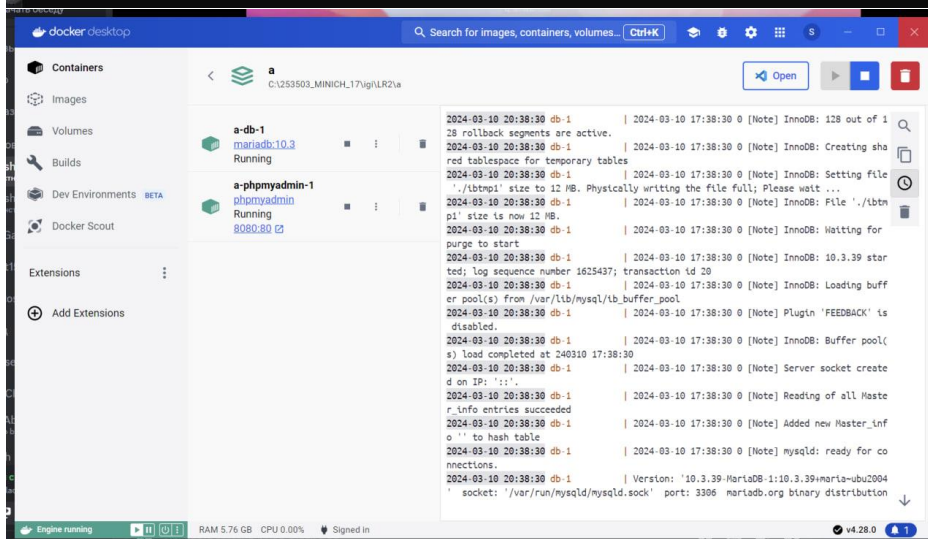
- Скачать любой доступный проект с GitHub с произвольным стеком технологий или использовать свой, ранее разработанный. Создать для него необходимый контейнер, используя Docker Compose для управления многоконтейнерными приложениями. Запустить проект в контейнере. (Примеры Images: [https://hub.docker.com/\\_/phpmyadmin](https://hub.docker.com/_/phpmyadmin), [https://hub.docker.com/\\_/mysql](https://hub.docker.com/_/mysql), [https://hub.docker.com/\\_/postgres](https://hub.docker.com/_/postgres)):

```
docker-compose.yml X
docker-compose.yml
1  version: '3.1'
2
3  services:
4    db:
5      image: mariadb:10.3
6      restart: always
7      networks:
8        - MyNetwork
9      volumes:
10       - db_data:/var/lib/mysql
11      environment:
12        MYSQL_ROOT_PASSWORD: 12345
13
14
15    phpmyadmin:
16      image: phpmyadmin
17      restart: unless-stopped
18      ports:
19        - "8080:80"
20      networks:
21        - MyNetwork
22
23  networks:
24    MyNetwork:
25
26  volumes:
27    db_data:
```

```
C:\253503_MINICH_17\igi\LR2\a>docker compose up -d
validating C:\253503_MINICH_17\igi\LR2\a\docker-compose.yml: networks must be a mapping

[+] Running 2/4H_17\igi\LR2\a>docker compose up -d
- Network a_MyNetwork      Created
- Volume "a_db_data"       Created
- Container a-phpmyadmin-1 Started
- Container a-db-1         Started

C:\253503_MINICH_17\igi\LR2\a>
```



The image shows a web browser displaying the phpMyAdmin interface at `localhost:8080/index.php?route=/&route=%2F`. The interface is in Russian and shows the 'Основные настройки' (Main settings) page. The 'Сервер баз данных' (Database server) section shows the following configuration:

- Сервер: db via TCP/IP
- Тип сервера: MariaDB
- Соединение сервера: SSL не используется
- Версия сервера: 10.3.39-MariaDB-1.10.3.39+maria-ubu2004 - mariadb.org binary distribution
- Версия протокола: 10
- Пользователь: root@172.18.0.2
- Кодировка сервера: cp1252 West European (latin1)

The 'Веб-сервер' (Web server) section shows:

- Apache/2.4.57 (Debian)
- Версия клиента базы данных: libmysql - mysqld 8.2.16
- PHP расширение: mysqli, curl, mbstring, sodium
- Версия PHP: 8.2.16

The 'phpMyAdmin' section shows:

- Информация о версии: 5.2.1 (актуально)
- Документация
- Официальная страница phpMyAdmin
- Пожертвовать
- Получить помощь
- Список изменений

Below the browser window, a terminal window shows the following commands and output:

```
C:\253503_MINICH_17\igi\LR2\>a>docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS
e5a818677d61   phpmyadmin "/docker-entrypoint.s..." About a minute ago Up About a minute 0.0.0.0:8080->80/tcp
33d93809afab   mariadb:10.3 "docker-entrypoint.s..." 7 minutes ago Up 3 minutes    3306/tcp

C:\253503_MINICH_17\igi\LR2\>a>docker stop e5a818677d61 33d93809afab
e5a818677d61
33d93809afab

C:\253503_MINICH_17\igi\LR2\>a>docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS        NAMES
e5a818677d61   phpmyadmin "/docker-entrypoint.s..." 2 minutes ago Exited (0) 7 seconds ago a-phpmyadmin-1
33d93809afab   mariadb:10.3 "docker-entrypoint.s..." 8 minutes ago Exited (0) 6 seconds ago a-db-1

C:\253503_MINICH_17\igi\LR2\>a>
```

5. Настроить сети и тома для обеспечения связи между контейнерами и сохранения данных (исходные данные, логин, пароль и т.д.):

```

docker-compose.yml X
docker-compose.yml
1  version: '3.1'
2
3  services:
4    db:
5      image: mariadb:10.3
6      restart: always
7      networks:
8        - MyNetwork
9      volumes:
10       - db_data:/var/lib/mysql
11      environment:
12        MYSQL_ROOT_PASSWORD: 12345
13
14
15    phpmyadmin:
16      image: phpmyadmin
17      restart: unless-stopped
18      ports:
19        - "8080:80"
20      networks:
21        - MyNetwork
22
23  networks:
24    MyNetwork:
25
26  volumes:
27    db_data:

```

6. Разместите результат в созданный репозиторий в DockerHub:

```

C:\Users\imsve>docker login
Authenticating with existing credentials...
Login Succeeded

C:\Users\imsve>docker tag phpmyadmin svettta/lab1

C:\Users\imsve>docker images
REPOSITORY          TAG         IMAGE ID      CREATED       SIZE
my_script            latest      e0d24f8d21f3  20 hours ago  1.02GB
phpmyadmin           latest      a9695a48170e  3 weeks ago   562MB
svettta/lab1         latest      a9695a48170e  3 weeks ago   562MB
python               latest      a3aef63c6c10  4 weeks ago   1.02GB
mysql                latest      019814493c7a  7 weeks ago   632MB
mariadb              10.3       1172e50de434  10 months ago 369MB
docker/getting-started latest      3e4394f6b72f  14 months ago 47MB

C:\Users\imsve>docker push svettta/lab1
Using default tag: latest
The push refers to repository [docker.io/svettta/lab1]
9e9ce41333f7: Mounted from library/phpmyadmin
cfc46e474789: Mounted from library/phpmyadmin
c938f3f71810: Mounted from library/phpmyadmin
3487d64778e1: Mounted from library/phpmyadmin

```



```
C:\Users\imsve>docker tag mariadb:10.3 sveltta/lab1:_db

C:\Users\imsve>docker images
REPOSITORY          TAG             IMAGE ID        CREATED         SIZE
my_script            latest          e0d24f8d21f3   20 hours ago   1.02GB
phpmyadmin           latest          a9695a48170e   3 weeks ago    562MB
sveltta/lab1         latest          a9695a48170e   3 weeks ago    562MB
python               latest          a3aef63c6c10   4 weeks ago    1.02GB
mysql                latest          019814493c7a   7 weeks ago    632MB
mariadb              10.3           1172e50de434   10 months ago  369MB
sveltta/lab1         _db            1172e50de434   10 months ago  369MB
docker/getting-started latest          3e4394f6b72f   14 months ago  47MB

C:\Users\imsve>docker push sveltta/lab1:_db
The push refers to repository [docker.io/sveltta/lab1]
f18a6d652910: Mounted from library/mariadb
f7ef09ddb028: Mounted from library/mariadb
b45260616e68: Mounted from library/mariadb
063dafaf1afa3: Mounted from library/mariadb
7adde0ac1571: Mounted from library/mariadb
b03c86585035: Mounted from library/mariadb
```

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

```
C:\Users\imsve>docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
323c65cbbaf8       MyNetwork           bridge              local
89e293162c48       a_MyNetwork         bridge              local
fc667091c10e       a_default           bridge              local
f9e012fe1780       bridge              bridge              local
92562512fa44       host                host                local
90720ddec06d       none                null                local
```

```
C:\Users\imsve>docker inspect bridge
[
  {
    "Name": "bridge",
    "Id": "f9e012fe178088792d51421f648c56cb76fc80739162571a7fb41ee4889552f1",
    "Created": "2024-03-11T09:26:52.219588591Z",
    "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,
    "Portables": false
  }
]
```

```
C:\Users\imsve>docker inspect host
[
  {
    "Name": "host",
    "Id": "92562512fa4428674d41c6d8af53b7d6f9d12a484c460f02a54be783cbdc8d5b",
    "Created": "2024-03-07T13:21:19.461969565Z",
    "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": {}
  }
]
```

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:

```
C:\Users\imsve>docker network create -d bridge another_one
262cd97c22c09b2c88719d11626348e434fa5113c272c82572735b7d8d034956
```

```
C:\Users\imsve>docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
323c65cbbaf8       MyNetwork           bridge              local
89e293162c48       a_MyNetwork         bridge              local
fc667091c10e       a_default           bridge              local
262cd97c22c0       another_one         bridge              local
f9e012fe1780       bridge              bridge              local
92562512fa44       host                host                local
90720ddec06d       none                null                local
```

```
C:\Users\imsve>docker run -it -d --name try --network another_one python
7f1bb2f7fd74a3f98369b1c5547641812fe7449bbebd3c57fa24b861b6cb5ff4
```

```
C:\Users\imsve>docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
323c65cbbaf8       MyNetwork           bridge              local
89e293162c48       a_MyNetwork         bridge              local
fc667091c10e       a_default           bridge              local
262cd97c22c0       another_one         bridge              local
f9e012fe1780       bridge              bridge              local
92562512fa44       host                host                local
90720ddec06d       none                null                local
```

Командная строка

```
C:\Users\imsve>docker inspect another_one
[
  {
    "Name": "another_one",
    "Id": "262cd97c22c09b2c88719d11626348e434fa5113c272c82572735b7d8d034956",
    "Created": "2024-03-11T09:40:39.800030118Z",
    "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": {
      "7f1bb2f7fd74a3f98369b1c5547641812fe7449bbebd3c57fa24b861b6cb5fff4": {
        "Name": "try",
        "EndpointID": "d0e3603550b2768a9332154fe9aeb45d4a656449964b2e9f50a0a9dfb995ec9d",
        "MacAddress": "02:42:ac:15:00:02",
        "IPv4Address": "172.21.0.2/16",
        "IPv6Address": ""
      }
    },
    "Options": {}
  }
]
```

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:

```
C:\Users\imsve>docker network create -d bridge bridge2
649b31a90a127b268ad6a31793faf8b74ee3dddabfe0d9006d91ef01ce7f958e

C:\Users\imsve>docker network ls
NETWORK ID        NAME                DRIVER              SCOPE
323c65cbbaf8     MyNetwork          bridge             local
89e293162c48     a_MyNetwork        bridge             local
fc667091c10e     a_default          bridge             local
262cd97c22c0     another_one        bridge             local
f9e012fe1780     bridge             bridge             local
649b31a90a12     bridge2            bridge             local
92562512fa44     host               host               local
90720ddec06d     none               null               local
```



```
C:\Users\imsve>docker inspect bridge2
[
  {
    "Name": "bridge2",
    "Id": "649b31a90a127b268ad6a31793faf8b74ee3dddabfe0d9006d91ef01ce7f958e",
    "Created": "2024-03-11T09:48:25.817768017Z",
    "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": {},
    "Options": {},
    "Labels": {}
  }
]
```

```
C:\Users\imsve>docker run -d --name try1 --network bridge2 python
f53d9c70617e9ed2eef52cba3293e9f6f4afeab045cf3b2c3ef9ffbc155219c3
```

```
C:\Users\imsve>docker run -d --name try2 --network bridge2 python
fdec6664e3c1b2dfb9bbc77964928074c69616469b5e086e55ffbf4f6fd7ca2d
```

```
C:\Users\imsve>docker run -d --name try3 --network bridge2 python
96440cc2a3482b27ad20b23f82d1a127d5f16240fdb755062dc646fac712d925
```

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

```
C:\Users\imsve>docker network create -d overlay ov  
ekm2vwe5q6dg5sidf92eynxah
```

```
C:\Users\imsve>docker network ls
```

| NETWORK ID    | NAME            | DRIVER  | SCOPE |
|---------------|-----------------|---------|-------|
| 323c65cbba8f8 | MyNetwork       | bridge  | local |
| 89e293162c48  | a_MyNetwork     | bridge  | local |
| fc667091c10e  | a_default       | bridge  | local |
| 262cd97c22c0  | another_one     | bridge  | local |
| f9e012fe1780  | bridge          | bridge  | local |
| 649b31a90a12  | bridge2         | bridge  | local |
| f97815606d1f  | docker_gwbridge | bridge  | local |
| 92562512fa44  | host            | host    | local |
| 3hydppri41f3  | ingress         | overlay | swarm |
| 90720ddec06d  | none            | null    | local |
| ekm2vwe5q6dg  | ov              | overlay | swarm |

```
C:\Users\imsve>docker network inspect ov
```

```
[  
  {  
    "Name": "ov",  
    "Id": "ekm2vwe5q6dg5sidf92eynxah",  
    "Created": "2024-03-11T10:04:45.470547181Z",  
    "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"  
    }  
  }  
]
```

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

```
C:\Users\imsve>docker network create -d overlay ov2
t2t5i7l7ew5ekkmr5gpxwx8fs
```

```
C:\Users\imsve>docker network ls
```

| NETWORK ID   | NAME            | DRIVER  | SCOPE |
|--------------|-----------------|---------|-------|
| 323c65cbbaf8 | MyNetwork       | bridge  | local |
| 89e293162c48 | a_MyNetwork     | bridge  | local |
| fc667091c10e | a_default       | bridge  | local |
| 262cd97c22c0 | another_one     | bridge  | local |
| f9e012fe1780 | bridge          | bridge  | local |
| 649b31a90a12 | bridge2         | bridge  | local |
| f97815606d1f | docker_gwbridge | bridge  | local |
| 92562512fa44 | host            | host    | local |
| 3hydppri4lf3 | ingress         | overlay | swarm |
| 90720ddec06d | none            | null    | local |
| ekm2vwe5q6dg | ov              | overlay | swarm |
| t2t5i7l7ew5e | ov2             | overlay | swarm |

```
C:\Users\imsve>docker network inspect ov2
```

```
[
  {
    "Name": "ov2",
    "Id": "t2t5i7l7ew5ekkmr5gpxwx8fs",
    "Created": "2024-03-11T10:07:41.291576971Z",
    "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
  }
]
```

```
C:\Users\imsve>docker network rm ov2  
ov2
```

```
C:\Users\imsve>docker network ls
```

| NETWORK ID   | NAME            | DRIVER  | SCOPE |
|--------------|-----------------|---------|-------|
| 323c65cbbaf8 | MyNetwork       | bridge  | local |
| 89e293162c48 | a_MyNetwork     | bridge  | local |
| fc667091c10e | a_default       | bridge  | local |
| 262cd97c22c0 | another_one     | bridge  | local |
| f9e012fe1780 | bridge          | bridge  | local |
| 649b31a90a12 | bridge2         | bridge  | local |
| f97815606d1f | docker_gwbridge | bridge  | local |
| 92562512fa44 | host            | host    | local |
| 3hydppri41f3 | ingress         | overlay | swarm |
| 90720ddec06d | none            | null    | local |
| ekm2vwe5q6dg | ov              | overlay | swarm |

Попробовать создать сеть host, сохранить результат в отчет:

```
C:\Users\imsve>docker network create -d host h  
Error response from daemon: only one instance of "host" network is allowed
```