

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

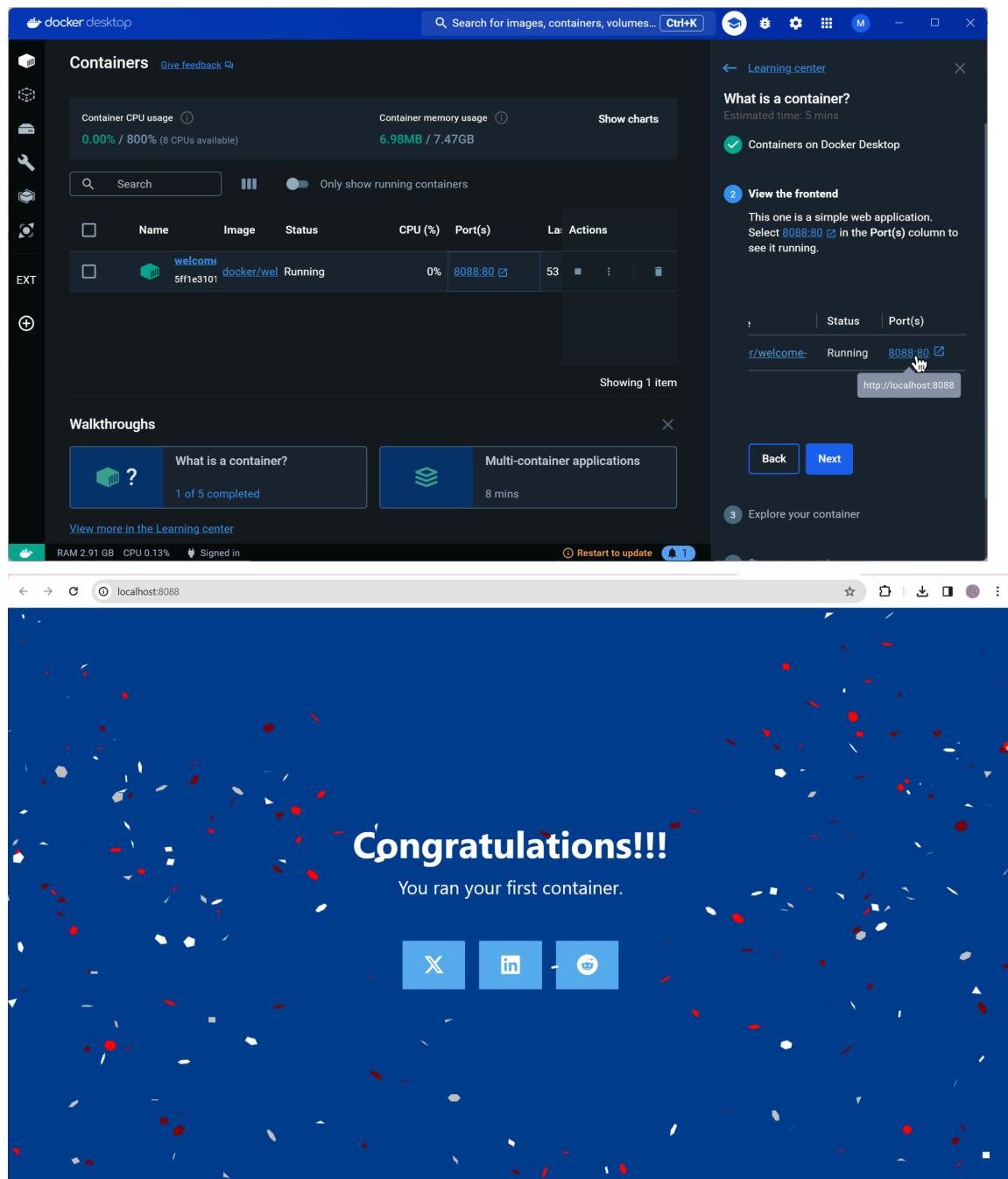
- Установите Docker
- Выполните базовую настройку

Установим Docker и ознакомимся с документацией

The screenshot shows a web browser displaying the Docker documentation at docs.docker.com/guides/walkthroughs/what-is-a-container/. The page title is "What is a container?". The left sidebar contains a navigation menu with sections like "Docker overview", "Get started", "Quick hands-on guides" (which is currently selected), "Run a container", "Run Docker Hub images", etc. The main content area describes what a container is: "A container is an isolated environment for your code. This means that a container has no knowledge of your operating system, or your files. It runs on the environment provided to you by Docker Desktop. Containers have everything that your code needs in order to run, down to a base operating system. You can use Docker Desktop to manage and explore your containers." Below this, there's a section titled "Step 1: Set up the walkthrough" with instructions for running a container using Docker Desktop. The right sidebar includes links for "Edit this page", "Request changes", "Contents", and a "Give feedback" button.

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:

```
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> git clone https://github.com/docker/getting-started.git
Cloning into 'getting-started'...
remote: Enumerating objects: 967, done.
remote: Counting objects: 100% (10/10), done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 967 (delta 3), reused 0 (delta 0), pack-reused 957
Receiving objects: 100% (967/967), 5.27 MiB | 2.23 MiB/s, done.
Resolving deltas: 100% (514/514), done.
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> |
```

```

Windows PowerShell
PS D:\IGI\253502_KRASOV_11\IGI\LR2> cd getting-started
PS D:\IGI\253502_KRASOV_11\IGI\LR2\getting-started> ls

Каталог: D:\IGI\253502_KRASOV_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_KRASOV_11\IGI\LR2\getting-started> code . Dockerfile
PS D:\IGI\253502_KRASOV_11\IGI\LR2\getting-started> |

```

```

Dockerfile - getting-started - Visual Studio Code
File Edit Selection View Go Run Terminal Help Dockerfile
Dockerfile
1 # Install the base requirements for the app.
2 # This stage is to support development.
3 FROM --platform=$BUILDPLATFORM python:alpine AS base
4 WORKDIR /app
5 COPY requirements.txt .
6 RUN pip install -r requirements.txt
7
8 FROM --platform=$BUILDPLATFORM node:18-alpine AS app-base
9 WORKDIR /app
10 COPY app/package.json app/yarn.lock .
11 COPY app/spec ./spec
12 COPY app/src ./src
13
14 # Run tests to validate app
15 FROM app-base AS test
16 RUN yarn install
17 RUN yarn test
18
19 # Clear out the node_modules and create the zip
20 FROM app-base AS app-zip-creator
21 COPY --from=test /app/package.json /app/yarn.lock .
22 COPY app/spec ./spec
23 COPY app/src ./src
24 RUN apk add zip && \
25     zip -r /app.zip /app
26
27 # Dev-ready container - actual files will be mounted in
28 FROM app-base AS app
29 COPY --from=app-zip-creator /app.zip /usr/share/nginx/html/assets/app.zip
30 CMD ["mkdocs", "serve", "-a", "0.0.0.0:8000"]
31
32 # Do the actual build of the mkdocs site
33 FROM --platform=$BUILDPLATFORM base AS build
34 COPY .
35 RUN mkdocs build
36
37 # Extract the static content from the build
38 # and use a nginx image to serve the content
39 FROM --platform=$TARGETPLATFORM nginx:alpine
40 COPY --from=app-zip-creator /app.zip /usr/share/nginx/html/assets/app.zip
41 COPY --from=build /app/site /usr/share/nginx/html

```

```

docker-compose.yml - getting-started - Visual Studio Code
File Edit Selection View Go Run Terminal Help docker-compose.yml
docker-compose.yml
1 version: "3.7"
2
3 services:
4   docs:
5     build:
6       context: .
7       dockerfile: Dockerfile
8       target: dev
9     ports:
10       - 8000:8000
11     volumes:
12       - ./app

```

Соберём образ и запустим данный контейнер

```
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\getting-started> docker build -t docker/getting-started .
[+] Building 0.6s (27/27) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 1.26kB
=> [internal] load metadata for docker.io/library/nginx:alpine
=> [internal] load metadata for docker.io/library/python:alpine
=> [internal] load metadata for docker.io/library/node:18-alpine
=> [internal] load .dockerrcignore
=> => transferring context: 68B
=> [app-base 1/5] FROM docker.io/library/node:18-alpine@sha256:ca9f6cb0466f9638e59e0c249d335a07c867cd50c429b5c78
=> [internal] load build context
=> => transferring context: 7.81kB
=> [base 1/4] FROM docker.io/library/python:alpine@sha256:1a0501213b470de000d8432b3caab9d8e5489e9443c2cc7ccaa6b
=> [stage-6 1/3] FROM docker.io/library/nginx:alpine@sha256:6a2f8b28e45c4adea04ec207a251fd4a2df03ddc930f782af51e
=> CACHED [app-base 2/5] WORKDIR /app
=> CACHED [app-base 3/5] COPY app/package.json app/yarn.lock .
=> CACHED [app-base 4/5] COPY app/spec ./spec
=> CACHED [app-base 5/5] COPY app/src ./src
=> CACHED [test 1/2] RUN yarn install
=> CACHED [test 2/2] RUN yarn test
=> CACHED [app-zip-creator 1/4] COPY --from=test /app/package.json /app/yarn.lock .
=> CACHED [app-zip-creator 2/4] COPY app/spec ./spec
=> CACHED [app-zip-creator 3/4] COPY app/src ./src
=> CACHED [app-zip-creator 4/4] RUN apk add zip && zip -r /app.zip /app
=> CACHED [stage-6 2/3] COPY --from=app-zip-creator /app.zip /usr/share/nginx/html/assets/app.zip
=> CACHED [base 2/4] WORKDIR /app
=> CACHED [base 3/4] COPY requirements.txt .
=> CACHED [base 4/4] RUN pip install -r requirements.txt
=> CACHED [build 1/2] COPY . .
=> CACHED [build 2/2] RUN mkdocs build
```

```
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\getting-started> docker run -d -p 30:30 docker/getting-started
e43e11cdef3126075b1cc078f8c73a84cdbcad16793747a147086378c514acda
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\getting-started> |
```

The command you just ran

Congratulations! You have started the container for this tutorial! Let's first explain the command that you just ran. In case you forgot, here's the command:

```
docker run -d -p 30:30 docker/getting-started
```

You'll notice a few flags being used. Here's some more info on them:

- `-d` - run the container in detached mode (in the background)
- `-p 30:30` - map port 30 of the host to port 30 in the container
- `docker/getting-started` - the image to use

Pro tip

You can combine single character flags to shorten the full command. As an example, the command above could be written as:

```
docker run -dp 30:30 docker/getting-started
```

The Docker Dashboard

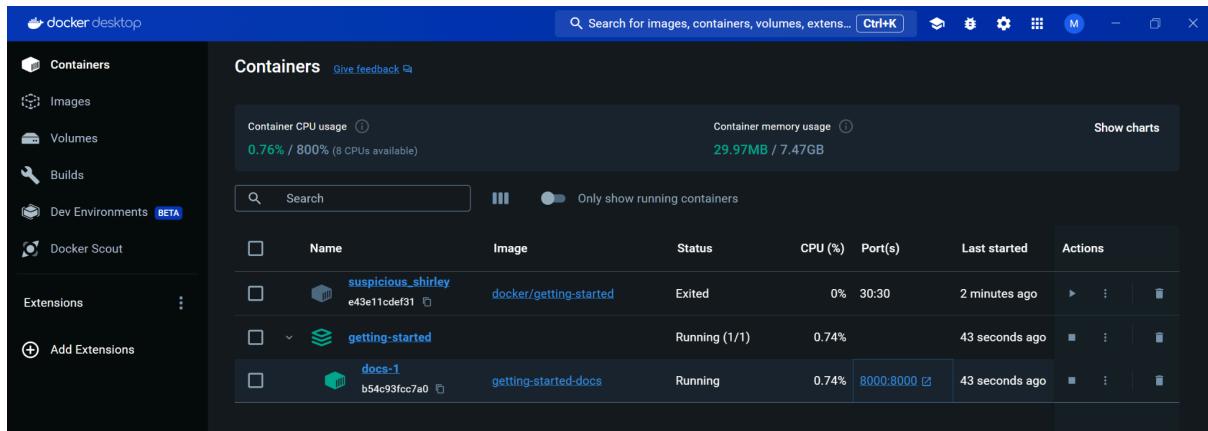
Before going any further, we want to highlight the Docker Dashboard, which gives you a quick view of the containers running on your machine. It provides you access to container logs, lets you get a shell inside the container, and allows you to easily manage container lifecycle (stop,

Используем compose чтобы собрать все контейнеры приложения вместе:

```

Windows PowerShell
PS D:\IGI\253502_KRASOV_1\IGI\LR2\getting-started> docker compose up
[+] Building 0.0s (0/0) docker:default
[+] Building 21.4s (10/10) FINISHED
=> [docs internal] load build definition from Dockerfile
=> => transferring dockerfile: 1.26kB
=> [docs internal] load metadata for docker.io/library/python:alpine
=> [docs auth] Library/python:pull token for registry-1.docker.io
=> [docs internal] load .dockernignore
=> => transferring context: 68B
=> [docs base 1/4] FROM docker.io/library/python:alpine@sha256:1a0501213b470de000d8432b3caab9d8de5489e9443c2cc7ccaa6b0aa5c3148e
=> => resolve docker.io/library/python:alpine@sha256:1a0501213b470de000d8432b3caab9d8de5489e9443c2cc7ccaa6b0aa5c3148e
=> => sha256:c54b53ca8371c4e91a0d13c637c289d114b91fffa18289e93387e3e80da979f8 6.04kB / 6.04kB
=> => sha256:4abc20661432fb2d719aaaf90656f55c287f8ca915dc1c92e14ff61e67fbaf8 3.41MB / 3.41MB
=> => sha256:dca80dc46cecdlla97787a1dd6f74263b9d2f7bdd3e2e15c109f5e34848c932 622.15kB / 622.15kB
=> => sha256:fe9e15b6315c34de5c802dbd343e3ec69bdc4ab870783fc1b9552daae2f25e77 11.77MB / 11.77MB
=> => sha256:1a0501213b470de000d8432b3caab9d8de5489e9443c2cc7ccaa6b0aa5c3148e 1.65kB / 1.65kB
=> => sha256:84f6ed6079c9f797ca9c1b7d6ea1c00aea3ac35110cbdd06003f15950017ea8d 1.37kB / 1.37kB
=> => sha256:a8fd6f3f484fdfccf33965ca0f8807e5078a619803cf638d82bc4a405e91de04 242B / 242B
=> => sha256:4fc96b5c1ba465ba27fb55d4766ade8624de4082ac1530b3293ac735ab3ead50 2.70MB / 2.70MB
=> => extracting sha256:4abc20661432fb2d719aaaf90656f55c287f8ca915dc1c92e14ff61e67fbaf8
=> => extracting sha256:dca80dc46cecdlla97787a1dd6f74263b9d2f7bdd3e2e15c109f5e34848c932
=> => extracting sha256:fe9e15b6315c34de5c802dbd343e3ec69bdc4ab870783fc1b9552daae2f25e77
=> => extracting sha256:a8fd6f3f484fdfccf33965ca0f8807e5078a619803cf638d82bc4a405e91de04
=> => extracting sha256:4fc96b5c1ba465ba27fb55d4766ade8624de4082ac1530b3293ac735ab3ead50
=> [docs internal] load build context
=> => transferring context: 153B
=> [docs base 2/4] WORKDIR /app
=> [docs base 3/4] COPY requirements.txt .
=> [docs base 4/4] RUN pip install -r requirements.txt
=> => exporting layers
=> => writing image sha256:1c748a1e96efaf16fda21aa48c3268c035a4316ca7e5c80c046adb4796b895b9
=> => naming to docker.io/library/getting-started-docs
[+] Running 2/0
  ✓ Network getting-started_default  Created
  ✓ Container getting-started-docs-1  Created
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

```



The screenshot shows a web browser window with the URL `localhost:8000/tutorial/`. The page is titled "Getting Started" and features a sidebar with links like "Getting Started", "Our Application", "Updating our App", etc. The main content area has a heading "The command you just ran" and a code snippet: `docker run -d -p 80:80 docker/getting-started`. A "Pro tip" box suggests combining flags: `docker run -dp 80:80 docker/getting-started`. On the right, there's a "Table of contents" sidebar with links to "The command you just ran", "The Docker Dashboard", "What is a container?", and "What is a container image?".

Проверим различные команды Docker:

Просмотр информации о пользователе:

```
PS D:\IGI\253502_KRASOV_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
```

```

Windows PowerShell

Using metacopy: false
Native Overlay Diff: true
userxattr: false
Logging Driver: json-file
Cgroup Driver: cgroupfs
Cgroup Version: 1
Plugins:
  Volume: local
  Network: bridge host ipvlan macvlan null overlay
  Log: awslogs fluentd gcplogs gelf journald json-file local splunk syslog
Swarm: inactive
Runtimes: io.containerd.runc.v2 runc
Default Runtime: runc
Init Binary: docker-init
containerd version: ae07eda36dd25f8a1b98dfbf587313b99c0190bb
runc version: v1.1.12-0-g51d5e94
init version: de40ad0
Security Options:
  seccomp
    Profile: unconfined
Kernel Version: 5.15.133.1-microsoft-standard-WSL2
Operating System: Docker Desktop
OSType: linux
Architecture: x86_64
CPUs: 8
Total Memory: 7.652GiB
Name: docker-desktop
ID: 74c8a0b5-5de6-4b3f-9541-e7d818e3e362
Docker Root Dir: /var/lib/docker
Debug Mode: false
HTTP Proxy: http.docker.internal:3128
HTTPS Proxy: https.docker.internal:3128
No Proxy: hubproxy.docker.internal
Experimental: false
Insecure Registries:
  Hubproxy.docker.internal:5555
  127.0.0.0/8
Live Restore Enabled: false

WARNING: No blkio throttle.read_bps_device support
WARNING: No blkio throttle.write_bps_device support
WARNING: No blkio throttle.read_iops_device support
WARNING: No blkio throttle.write_iops_device support
WARNING: daemon is not using the default seccomp profile
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\getting-started> |

```

Просмотр всех образов:

```

Windows PowerShell

PS D:\IGI\253502_KRASYOV_11\IGI\LR2\getting-started> docker images
REPOSITORY          TAG      IMAGE ID      CREATED        SIZE
getting-started-docs   latest   1c74aab9e96e  5 minutes ago  97.1MB
docker/welcome-to-docker  latest   c1f619b6477e  3 months ago  18.6MB
docker/getting-started  latest   3e439af6b72f  14 months ago  47MB
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\getting-started> |

```

Просмотр всех/запущенных контейнеров:

```

Windows PowerShell

PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker ps -a
CONTAINER ID   IMAGE           COMMAND                  CREATED        STATUS          PORTS
TS             NAMES
c25ef63acd72  docker/getting-started  "/docker-entrypoint..."  11 minutes ago  Up 11 minutes  0.0
.0.0:80->80/tcp
c66e3f6c1b5b  getting-started-docs   "mkdocs serve -a 0.0..."  5 days ago   Exited (137) 5 days ago
      getting-started-docs-1
35cb784e9fe3  docker/getting-started  "/docker-entrypoint..."  5 days ago   Exited (0) 5 days ago
      agitated_kalam
5ff1e3101606  docker/welcome-to-docker:latest  "/docker-entrypoint..."  5 days ago   Exited (0) 5 days ago
      welcome-to-docker
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> |

```

```

Windows PowerShell

PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker ps -s
CONTAINER ID   IMAGE           COMMAND                  CREATED        STATUS          PORTS
AMES           SIZE
c25ef63acd72  docker/getting-started  "/docker-entrypoint..."  53 seconds ago  Up 52 seconds  0.0.0.0:80->80/tcp  d
istracted_euler  1.09kB (virtual 47MB)
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> |

```

Просмотр логов контейнера:

```

PS D:\IGI\253502_KRASOV_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_KRASOV_11\IGI\LR2>

```

Остановка контейнера:

```

Windows PowerShell
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker stop c25ef63acd72
c25ef63acd72
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker ps -s
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES SIZE
PS D:\IGI\253502_KRASOV_11\IGI\LR2>

```

Запуск, приостановка и возобновление работы контейнера:

```

Windows PowerShell
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker start c25ef63acd72a4c3ded6852ead5d75d40a3fb825004fd78589fbffbc92bbea2
c25ef63acd72a4c3ded6852ead5d75d40a3fb825004fd78589fbffbc92bbea2
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker ps -s
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES SIZE
MES
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_KRASOV_11\IGI\LR2> docker pause c25ef63acd72a4c3ded6852ead5d75d40a3fb825004fd78589fbffbc92bbea2
c25ef63acd72a4c3ded6852ead5d75d40a3fb825004fd78589fbffbc92bbea2
PS D:\IGI\253502_KRASOV_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->80/tcp
0/0/tcp distracted_euler 1.09kB (virtual 47MB)
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker unpause c25ef63acd72a4c3ded6852ead5d75d40a3fb825004fd78589fbffbc92bbea2
c25ef63acd72a4c3ded6852ead5d75d40a3fb825004fd78589fbffbc92bbea2
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker ps -s
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES SIZE
AMES
c25ef63acd72 docker/getting-started "/docker-entrypoint..." 16 minutes ago Up 45 seconds 0.0.0.0:80->80/tcp di
istracted_euler 1.09kB (virtual 47MB)
PS D:\IGI\253502_KRASOV_11\IGI\LR2>

```

Перезапуск контейнера:

```

Windows PowerShell
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker ps -s
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES SIZE
MES
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_KRASOV_11\IGI\LR2> docker restart c25ef63acd72
c25ef63acd72
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker ps -s
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES 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_KRASOV_11\IGI\LR2>

```

Переименование и удаление контейнера:

```

PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker rename c25ef63acd72 renamed_one
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker ps -s
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
c25ef63acd72 docker/getting-started "/docker-entrypoint..." 21 minutes ago Up 2 minutes 0.0.0.0:80->80/tcp renamed_one
1.09kB (virtual 47MB)
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker stop c25ef63acd72
c25ef63acd72
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker ps -s
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker rm renamed_one
renamed_one
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
c66e3f6c1b5b getting-started-docs "mkdocs serve -a 0.0..." 5 days ago Exited (137) 5 days ago
g-started-docs-1
35cb784e9fe3 docker/getting-started "/docker-entrypoint..." 5 days ago Exited (0) 5 days ago
ed_kalam
5ff1e3101606 docker/welcome-to-docker:latest "/docker-entrypoint..." 5 days ago Exited (0) 5 days ago
e-to-docker
PS D:\IGI\253502_KRASOV_11\IGI\LR2>

```

Удаление образа:

```

PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker rmi docker/welcome-to-docker
Untagged: docker/welcome-to-docker:latest
Deleted: sha256:c1f619b6477e36a0b6a2531a972e918ef32bbf0217ee9b536409361261db6df0
Deleted: sha256:503c5cd6e10d87f52ccbcbe0fee9b033c6df11dee0055c636caa64f65227d02cc
Deleted: sha256:12369c7fe5ffb31bc592a24c0cd081c85f34702da4b747ede00543e6f7f54a74
Deleted: sha256:2b3208f4feef2df0b1c1744e87d2a5c41a1ef41a1217f7d90f1e7c1dab2ee30
Deleted: sha256:97912e57274d7772d7f052fe2d671c5e0ac193863e9d5d02d2575949c17e1cd0
Deleted: sha256:8d49f96bd3dac9f64c8b46bda71c268caa7eafbf1d9fde95b93a36133aae805fc
Deleted: sha256:2765f389f779d9903825e36b704119da1da13faa4e73a44u78fd86f577f4b738
Deleted: sha256:baeb76f1ff72a2a650534f62c17308491c058905a82289971c604dea72fe54ed
Deleted: sha256:cc2447e1835a40530975ab80bb1f872fbab0f2a0faecf2ab16fb89b3589438
PS D:\IGI\253502_KRASOV_11\IGI\LR2>

```

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

Клонируем GitHub репозиторий

```

PS D:\IGI\253502_KRASOV_11\IGI\LR2> git clone https://github.com/smartiqaorg/geometric_lib
Cloning into 'geometric_lib'...
remote: Enumerating objects: 35, done.
remote: Total 35 (delta 0), reused 0 (delta 0), pack-reused 35
Receiving objects: 100% (35/35), 4.60 KiB | 4.60 MiB/s, done.
Resolving deltas: 100% (6/6), done.
PS D:\IGI\253502_KRASOV_11\IGI\LR2>

```

Создаём Dockerfile

```
PS D:\IGI\253502_KRASOV_11\IGI\LR2\geometric_lib> code . Dockerfile
PS D:\IGI\253502_KRASOV_11\IGI\LR2\geometric_lib> |
```

 Dockerfile U

```
Dockerfile
1 FROM python
2 ENV env_param=7
3 WORKDIR /app
4 COPY .
5 VOLUME /app/data
6 ENTRYPOINT [ "python", "./main.py" ]
```

Создаём главный скрипт-файл:

 main.py U

```
main.py
main.py
1 import os
2 from square import *
3 from circle import *
4 from data import config
5
6 value = 1
7
8
9 def printRes(param: float) -> None:
10     if (param <= 0):
11         return
12     print("PARAM: {param}")
13     print("SQUARE")
14     print(f"PERIMETER: {perimeter(param)}")
15     print(f"AREA: {area(param)}")
16     print("CIRCLE")
17     print(f"LENGTH: {perimeterC(param)}")
18     print(f"AREA: {areaC(param)}")
19     print("=====")
20
21
22 print("ENV PARAM")
23 printRes(float(os.getenv('env_param')))
24 print("CONFIG PARAM")
25 printRes(float(config.param))
26
27 while value > 0:
28     value = float(input(
29         'Введите значение длины стороны/радиуса. Для выхода введите 0: '))
30     printRes(value)
31
32 print("Программа завершила работу.")
```

Создаём образ:

```
PS D:\IGI\253502_KRASOV_11\IGI\LR2\geometric_lib> docker build -t my_script .
[+] Building 63.5s (9/9) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 162B
=> [internal] load metadata for docker.io/library/python:latest
=> [auth] library/python:pull token for registry-1.docker.io
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [1/3] FROM docker.io/library/python:latest@sha256:e83d1f4d0c735c7a54fc9daea3ccaa8c58473e3b3de08fc7ba3d342ee75c
=> => resolve docker.io/library/python:latest@sha256:e83d1f4d0c735c7a54fc9daea3ccaa8c58473e3b3de08fc7ba3d342ee75c
=> => sha256:a3aeef63c6c1029222ff635a72a0fcaddba1fd7a26abdf2491ace8eb3c654471e 7.11kB / 7.11kB 0.0s
=> => sha256:e83d1f4d0c735c7a54fc9daea3ccaa8c58473e3b3de08fc7ba3d342ee75c09d 2.14kB / 2.14kB 0.0s
=> => sha256:35efff340c8acd837b7962f77e0eb8869385dd6fe7d3928375a08f0a3bdd18beb 2.01kB / 2.01kB 0.0s
=> => sha256:7bb465c2914923b08ae03b7fc67b92a1ef9b09c4c1eb9d6711b22ee6bbb46a00 49.55MB / 49.55MB 29.2s
=> => sha256:2b9b41aaa3c52ab268b47da303015b94ced04a1eb02e58860e58b283404974f4 24.05MB / 24.05MB 4.8s
=> => sha256:49b40be4436eff6fe463f6977159dc727df37cabef65ade75c1caa3cb0a234 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:e1aa7eb23da86d5a03e19f387349708052867452d96f32ac5eaf3d1fb8d702d 22.71MB / 22.71MB 39.1s
=> => extracting sha256:49b40be4436eff6fe463f6977159dc727df37cabef65ade75c75c1caa3cb0a234 2.0s
=> => sha256:297428e82567ff4bbd3fc4b5fe375e5aab1af71aaa00d93afe50583cb71d3ae3 2448 / 2448 32.9s
=> => sha256:28e810024ad5211525fc84f6532311b6fce2848e8e99e765355b0d1430c224b4 2.70MB / 2.70MB 34.0s
=> => extracting sha256:c558fac597f8ecbb7a66712e14912ce1d83b23a92ca8b6ff14eef209ab01aff2 4.6s
```

Запускаем контейнер и проверяем его работу:

```
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\geometric_lib> docker run -v ./data:/app/data -i my_script
ENV PARAM
PARAM: 7.0
SQUARE
PERIMETER: 28.0
AREA: 49.0
CIRCLE
LENGTH: 43.982297150257104
AREA: 153.93804002589985
=====
CONFIG PARAM
PARAM: 2.0
SQUARE
PERIMETER: 8.0
AREA: 4.0
CIRCLE
LENGTH: 12.566370614359172
AREA: 12.566370614359172
=====
Введите значение длины стороны/радиуса. Для выхода введите 0: 1.25
PARAM: 1.25
SQUARE
PERIMETER: 5.0
AREA: 1.5625
CIRCLE
LENGTH: 7.853981633974483
AREA: 4.908738521234052
=====
Введите значение длины стороны/радиуса. Для выхода введите 0: 0
Программа завершила работу.
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\geometric_lib>
```

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 многою 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:

```
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> git clone https://github.com/JustGimli/MPDocker.git
Cloning into 'MPDocker'...
remote: Enumerating objects: 1560, done.
remote: Counting objects: 100% (438/438), done.
remote: Compressing objects: 100% (309/309), done.
remote: Total 1560 (delta 156), reused 367 (delta 116), pack-reused 1122Receiving objects: 99% (1545/1560), 9.66 MiB | Receiving objects: 100% (1560/1560), 10.15 MiB | 2.11 MiB/s, done.

Resolving deltas: 100% (755/755), done.
PS D:\IGI\253502_KRASYOV_11\IGI\LR2>
```

Docker-compose file:

```
1 services:
2   nginx:
3     image: nginx:1.25-alpine
4     restart: always
5     volumes:
6       - ./nginx/default.conf:/etc/nginx/conf.d/default.conf
7       - ./nginx/mefodiy.net.crt:/etc/nginx/ssl/mefodiy.net.crt
8       - ./nginx/mefodiy.net.key:/etc/nginx/ssl/mefodiy.net.key
9       - ./nginx/ca_bundle.crt:/etc/nginx/ssl/ca_bundle.crt
10      - media:/var/www/html/media
11      - ./frontend/build:/var/www/html
12     depends_on:
13       - server
14     ports:
15       - 80:80
16   server:
17     restart: unless-stopped
18     network_mode: host
19     build:
20       context: ./backends/
21       dockerfile: Dockerfile
22     environment:
23       OPENAI_API_KEY: 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
24       YOOKASSA_API: https://api.yookassa.ru/v3/
25       SQL_ENGINE: django.db.backends.postgresql
26       SQL_DATABASE: postgres
27       SQL_HOST: localhost
28       SQL_USER: postgres
29       SQL_PASSWORD: postgres
30       SQL_PORT: 5432
31
32     entrypoint: ./api.sh
33     volumes:
34       - media:/home/app/media
35   celery:
36     restart: always
37     network_mode: host
38     build:
39       context: ./backends/
40       dockerfile: Dockerfile
41     entrypoint: ./celery.sh
42
43     volumes:
44       - media:/home/app/media
45       - /var/run/docker.sock:/var/run/docker.sock
46     environment:
47       OPENAI_API_KEY: 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
48       YOOKASSA_API: https://api.yookassa.ru/v3/
49       SQL_ENGINE: django.db.backends.postgresql
50       SQL_DATABASE: postgres
51       SQL_HOST: localhost
52       SQL_USER: postgres
53       SQL_PASSWORD: postgres
54       SQL_PORT: 5432
55     depends_on:
56       - server
57   postgres:
58     image: postgres
59     environment:
60       POSTGRES_DB: postgres
61       POSTGRES_USER: postgres
62       POSTGRES_PASSWORD: postgres
63     ports:
64       - "5432:5432"
65   redis:
66     image: "redis:latest"
67     restart: always
68     network_mode: host
69     ports:
70       - "6379:6379"
71     volumes:
72       media: {}
```

Выполним сборку образов и запустим приложение

```
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)
=> => transferring context: 2B
[+] Building 63.4s (17/17)
=> => resolve docker.io/library/python:3.9-slim@sha256:e0bc011bb55918109921b913fe30160cb8297c570621a450477d44999
[+] Building 64.1s (27/27) FINISHED
=> [server internal] load build definition from Dockerfile
=> transferring dockerfile: 563B
=> [celery internal] load metadata for docker.io/library/python:3.9-slim
=> [server auth] library/python:pull token for registry-1.docker.io
=> [server internal] load .dockerignore
=> transferring context: 2B
=> [celery 1/11] FROM docker.io/library/python:3.9-slim@sha256:e0bc011bb55918109921b913fe30160cb8297c570621a450477d44999
=> => resolve docker.io/library/python:3.9-slim@sha256:e0bc011bb55918109921b913fe30160cb8297c570621a450477d44999
=> => sha256:51d1f07906b71fd60ac43c61035514996a8ad8dbfd39d4f570ac546b064ee5d 3.51MB / 3.51MB
=> => sha256:336c7f590cb97722bfee12f22e354df879feae9f28bfd5cebeaffccb3fb8fb5 11.89MB / 11.89MB
=> => sha256:93b25b5c998e137849343851763bbc686f369399894e766c37a55304b1f66cfb 244B / 244B
=> => sha256:e0bc011bb55918109921b913fe30160cb8297c570621a450477d44999 1.86kB / 1.86kB
=> => sha256:51c781cd11dd1f2a95e2bef833a5920042743fa502d6cc9e12c1a841d983f9a7 1.37kB / 1.37kB
=> => sha256:9c14a9ca10408336092a7089469ef9a84f3caf196f7f65c512677e1f06e5d839 6.92kB / 6.92kB
=> => sha256:2b527dfdb0a9ecb9a5bf5c264aeb4ac254e0752886f187095f6960b933aab941 3.13MB / 3.13MB
=> => extracting sha256:51d1f07906b71fd60ac43c61035514996a8ad8dbfd39d4f570ac5446b064ee5d
=> => extracting sha256:336c7f590cb97722bfee12f22e354df879feae9f28bfd5cebeaffccb3fb8fb5
=> => extracting sha256:93b25b5c998e137849343851763bbc686f369399894e766c37a55304b1f66cfb
=> => extracting sha256:2b527dfdb0a9ecb9a5bf5c264aeb4ac254e0752886f187095f6960b933aab941
=> [server internal] load build context
=> => transferring context: 57.28kB
=> CACHED [celery 2/11] RUN apt-get update && apt-get install -y libpq-dev
4.8s
```

```
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-celelry    latest   6354ec0242f0  5 hours ago   704MB
docker/welcome-to-docker latest   c1f619b6477e  4 months ago  18.6MB
docke/getting-started latest   3e4394f6b72f  14 months ago  47MB
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\MPDocker> |
```

```
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
  ✓ Container mpdocker-nginx-1                            Created          0.0s
  ✓ Container mpdocker-celelry-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 * o000o000o000 Redis is starting o000o000o000o
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

Docker Desktop interface showing the 'Containers' tab. The sidebar includes 'Containers', 'Images', 'Volumes', 'Builds', 'Dev Environments (BETA)', and 'Docker Scout'. The main area displays container statistics: Container CPU usage (0.20% / 800%) and Container memory usage (414.6MB / 7.47GB). A table lists six running containers:

Name	Image	Status	CPU (%)	Port(s)	Last started	Actions
mpdocker		Running (5/5)	0%		6 seconds ago	[...]
redis-1	redis:latest	Running	0%	6379:6379	6 seconds ago	[...]
server-1	btc5650766c1	mpdocker-server	Running	0%	6 seconds ago	[...]
celery-1	330f18938ba6	mpdocker-celery	Running	0%	6 seconds ago	[...]
postgres-1	005e7090f5f3	postgres	Running	5432:5432	6 seconds ago	[...]
nginx-1	5824af6fd5e9	nginx:1.25-alpine	Running	80:80	6 seconds ago	[...]

Showing 6 items

The browser window shows two pages side-by-side.

Left Page (localhost):

Header: MP

Content: Создайте продающее описание для своего товара

Image: A white robot standing next to the WB logo and the Ozon logo.

Right Page (localhost):

Header: localhost

Section: Создать Аккаунт/Войти

Sign-in options: Войти как Paul paulwithdrums@bk.ru or Google

Or

Email input field

Password input field with visibility toggle

Запомнить меня [Забыли пароль?](#) [Пользовательское Соглашение](#)

ВОЙТИ button

localhost/profile

Мефодий
Описание продукта

СГЕНЕРИРОВАТЬ ОПИСАНИЕ

ТАРИФЫ

ИСТОРИЯ ОПИСАНИЙ

СВЯЗЬСЯТЬ 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

СВЯЗЬСЯТЬ 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](#)

The screenshot shows a web interface for a service. On the left, there's a sidebar with user information ('Методий'), a 'ГЕНЕРИРОВАТЬ ОПИСАНИЕ' button, a 'ТАРИФЫ' section, a 'ИСТОРИЯ ОПИСАНИЙ' link, and a 'СВЯЗЬСЯСЬ С НАМИ' link. The main area displays three pricing plans:

- Бесплатный** (For testing) - **₽0**: Includes 2 massive generations, 5 repeat generations, and up to 5000 characters.
- Стартовый** (For those just starting) - **₽1590 / Месяц**: Includes 100 generations, 15.9 rubles per generation, up to 100,000 characters, 10+ templates, AI chat, and generation history.
- Базовый** (For small business owners) - **₽6 490 / Месяц**: Includes 500 generations, 12.9 rubles per generation, up to 500,000 characters, 10+ templates, AI chat, and generation history.

A 'ВВЕСТИ ПРОМОКОД' (Enter coupon code) button is located at the top right. A 'Пользовательское Соглашение' (User Agreement) link is at the bottom left.

Задание 6. Разместите результат в созданный репозиторий в DockerHub

Создадим репозиторий

Create repository

Namespace: **mojiodoy** Repository Name *: **igi_labs**

Short description

A short description to identify your repository. If the repository is public, this description is used to index your content on Docker Hub and in search engines, and is visible to users in search results.

Visibility

Using 0 of 1 private repositories. [Get more](#)

Public

Appears in Docker Hub search results

Private

Only visible to you

[Cancel](#)

[Create](#)

Сохраним образы в репозитории:

```
Windows PowerShell
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\MPDocker> docker images
REPOSITORY          TAG      IMAGE ID   CREATED        SIZE
mpdocker-server    latest   c02c8783f535  45 minutes ago  707MB
mpdocker-celery    latest   26450435271d  45 minutes ago  707MB
my_script           latest   3ac9a991cfe6  About an hour ago  1.02GB
docker/getting-started latest   f86b47f26124  About an hour ago  48.8MB
getting-started-docs latest   88d4a6f01f94  About an hour ago  97.1MB
postgres            latest   eb634efa7e4  2 weeks ago   431MB
nginx               1.25-alpine 6913ed9ec8d0  3 weeks ago   42.6MB
redis               latest   d1397258b209  2 months ago   138MB
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\MPDocker> docker tag mpdocker-server:latest mojiodoy/igi-labs:server
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\MPDocker> docker push mojiodoy/igi-labs:server
The push refers to repository [docker.io/mojiodoy/igi-labs]
5f70bf18a086: Pushed
9d7ee2ae47f6: Pushed
299cabbab0d4: Pushed
10fafd81684f: Pushed
989376f667d9: Pushed
e443dcfc6bb16: Pushed
1c74f29dbf0a: Pushed
4c26f0e4877f: Pushed
516d5dcdfa86: Pushed
5a4af3b9b4c6: Pushed
e720351aac4f: Pushed
7ce4521e7cb2: Pushed
619b503a99a8: Pushed
defau9bf7f54: Pushed
ba473bfd54e: Pushed
ceb365432eec: Pushed
server: digest: sha256:05c5ed6b933aeec20e910fa8e896c20d97e1e64e6a599ac07bd524ac169dbb0c size: 3676
```

```
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\MPDocker> docker tag mpdocker-celery:latest mojiodoy/igi-labs:celery
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\MPDocker> docker push mojiodoy/igi-labs:celery
The push refers to repository [docker.io/mojiodoy/igi-labs]
5f70bf18a086: Layer already exists
9d7ee2ae47f6: Layer already exists
299cabbab0d4: Layer already exists
10fafd81684f: Layer already exists
989376f667d9: Layer already exists
e443dcfc6bb16: Layer already exists
1c74f29dbf0a: Layer already exists
4c26f0e4877f: Layer already exists
516d5dcdfa86: Layer already exists
5a4af3b9b4c6: Layer already exists
e720351aac4f: Layer already exists
7ce4521e7cb2: Layer already exists
619b503a99a8: Layer already exists
defau9bf7f54: Layer already exists
ba473bfd54e: Layer already exists
ceb365432eec: Layer already exists
celery: digest: sha256:cb977d73ed9370f1c78740ee53de87687a16ca8a12c46ef9a1078a6021922e70 size: 3676
PS D:\IGI\253502_KRASYOV_11\IGI\LR2\MPDocker> |
```

The screenshot shows the Docker Hub interface for the repository `mojiodoy/igi-labs`. The `Tags` tab is selected. There are two entries:

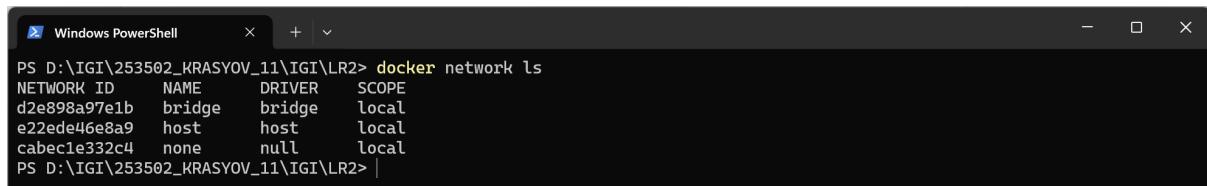
- celery**: Last pushed in a few seconds by `mojiodoy`.
 - Digest: `cb977d73ed93`
 - OS/ARCH: `linux/amd64`
 - Last pull: `---`
 - Compressed Size: `271.4 MB`
- server**: Last pushed 2 minutes ago by `mojiodoy`.
 - Digest: `05c5ed6b933a`
 - OS/ARCH: `linux/amd64`
 - Last pull: `---`
 - Compressed Size: `271.4 MB`

For each entry, there is a `Copy` button next to the Docker pull command.

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:



```
PS D:\IGI\253502_KRASOV_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_KRASOV_11\IGI\LR2> |
```

Теперь получим более подробную информацию о каждом типе сетей:

Bridge:

```
Windows PowerShell x + 
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker network ls
NETWORK ID      NAME      DRIVER      SCOPE
d2e898a97e1b    bridge    bridge      local
e2zede46e8a9    host      host      local
cabec1e332c4    none      null      local
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker network inspect bridge
[ {
  "Name": "bridge",
  "Id": "d2e898a97e1ba4b182cf5a0b1082f2b03265e1ef95f4b739a81cbb2962a813",
  "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_KRASOV_11\IGI\LR2> |
```

Host:

```
Windows PowerShell x + 
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker network inspect host
[ {
  "Name": "host",
  "Id": "e2zede46e8a95b8e66ec29a13059f8fe6effa9ea528d4b2294cdf27568513710",
  "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_KRASOV_11\IGI\LR2> |
```

Overlay:

```
Windows PowerShell x + 
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker network inspect overlay
[]
Error response from daemon: network overlay not found
PS D:\IGI\253502_KRASOV_11\IGI\LR2> |
```

Macvlan:

```
Windows PowerShell + - ×
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network inspect macvlan
[]
Error response from daemon: network macvlan not found
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> |
```

None:

```
Windows PowerShell + - ×
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network inspect none
[
  {
    "Name": "none",
    "Id": "cabec1e332c489f047dc2e8e65d7a5b646ecfa1a640efc787999e5bb5d94c361",
    "Created": "2024-03-11T16:00:44.235208774Z",
    "Scope": "local",
    "Driver": "null",
    "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> |
```

Создадим новую сеть типа bridge:

```
Windows PowerShell + - ×
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network create -d bridge my_script_net
ba@965ea17675121c49da#814aebe42bd19c4850af0443dff033c00b08e0d28
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network ls
NETWORK ID      NAME      DRIVER      SCOPE
d28989a97e1b    bridge    bridge      local
e2zed46e8a9    host      host      local
ba@965ea176    my_script_net    bridge      local
cabec1e332c4    none      null      local
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network inspect ba@965ea176
[
  {
    "Name": "my_script_net",
    "Id": "ba@965ea17675121c49da#814aebe42bd19c4850af0443dff033c00b08e0d28",
    "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> |
```

Запустим контейнер в этой сети:

```

Windows PowerShell
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker images
REPOSITORY          TAG      IMAGE ID      CREATED             SIZE
mpdocker-ce celery    latest      26450435271d  About an hour ago  707MB
mojiiodoy/igi-labs celery    latest      26450435271d  About an hour ago  707MB
mojiiodoy/igi-labs server    latest      c02c8783f535  About an hour ago  707MB
mpdocker-server     latest      c02c8783f535  About an hour ago  707MB
my_script           latest      3ac9a991cf6   2 hours ago       1.02GB
docker/getting-started latest      f86b47f26124  2 hours ago       48.8MB
getting-started-docs latest      88d4aef01f94  2 hours ago       97.1MB
postgres            latest      eb634efa7ee4  2 weeks ago        431MB
nginx               1.25-alpine latest      6913ed9ec8d0  3 weeks ago        42.6MB
httpd               latest      2776f4da9d55  7 weeks ago        167MB
redis               latest      d1397258b209  2 months ago       138MB
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker run -it -d -p 5000:5000 --network=my_script_net httpd
130a60bebea443cc4f477a30ef05c2ebb2752ffff6241f56de6a2aa81a8a5ff41
PS D:\IGI\253502_KRASOV_11\IGI\LR2>

```

Просмотрим информацию о контейнере:

```

Windows PowerShell
PS D:\IGI\253502_KRASOV_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_KRASOV_11\IGI\LR2> docker inspect 130a60bebea4
{
  "Id": "130a60bebea443cc4f477a30ef05c2ebb2752ffff6241f56de6a2aa81a8a5ff41",
  "Created": "2024-03-11T18:42:29.401041Z",
  "Status": "running",
  "RunningSince": "2024-03-11T18:42:29.401041Z",
  "Image": "httpd:1.25-alpine",
  "ImageID": "6913ed9ec8d0",
  "Labels": {},
  "Ports": [
    {
      "HostPort": "5000"
    }
  ],
  "HostConfig": {
    "Binds": null,
    "ContainerIDFile": null,
    "LogConfig": {
      "Type": "json-file",
      "Config": {}
    },
    "NetworkMode": "my_script_net",
    "PortBindings": {
      "5000/tcp": [
        {
          "HostIp": "",
          "HostPort": "5000"
        }
      ]
    },
    "RestartPolicy": {
      "Name": "no",
      "MaximumRetryCount": 0
    },
    "AutoRemove": false,
    "VolumeDriver": null,
    "VolumesFrom": null,
    "ConsoleSize": [
      30,
      120
    ],
    "Env": []
  }
}

```

Просмотрим информацию о нашей сети:

```

Windows PowerShell
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker network inspect my_script_net
[
  {
    "Name": "my_script_net",
    "Id": "ba0a965ea17675121c49daf814aebe42bd19c4850af0443dff033c00b08e0d28",
    "Created": "2024-03-11T18:42:29.401041Z",
    "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": {
      "130a60bebea443cc4f477a30ef05c2ebb2752ffff6241f56de6a2aa81a8a5ff41": {
        "Name": "trusting_nash",
        "EndpointID": "746c1e285d8f20ba44394f6dd3917bcb20d5993af6e49f3e23483fd331f94c55",
        "MacAddress": "02:42:ac:15:00:02",
        "IPv4Address": "172.21.0.2/16",
        "IPv6Address": ""
      }
    },
    "Options": {},
    "Labels": {}
  }
]
PS D:\IGI\253502_KRASOV_11\IGI\LR2>

```

Отключим контейнер от сети и просмотрим информацию о ней:

```

Windows PowerShell
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker network disconnect my_script_net 130a60bebea443cc4f477a30ef05c2ebb2752fff6241f56de6a2aa81a8a5ff41
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker network inspect my_script_net
[{"Name": "my_script_net", "Id": "ba0a965ea17675121c49daf814aebef42bd19c4850af0443dff033c00b08e0d28", "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_KRASOV_11\IGI\LR2>

```

Создадим новую bridge сеть и подключим в неё наши образы:

```

Windows PowerShell
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker network create -d bridge new_big_net
7a693e790cd174a5f95beadd614f515511254320eb18c4433e4ea890efae86f
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker network ls
NETWORK ID      NAME      DRIVER      SCOPE
d2e898a97e1b    bridge    bridge      local
e22ede4668a9   host      host       local
ba0a965ea176   my_script_net  bridge      local
7a693e790cd1   new_big_net   bridge      local
cabec1e332c4   none      null       local
PS D:\IGI\253502_KRASOV_11\IGI\LR2>

```

```

Windows PowerShell
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS              PORTS                 NAMES
bcf8814ee48        nginx:1.25-alpine  "/docker-entrypoint..."  16 seconds ago    Up 14 seconds     0.0.0.0:80->80/tcp   mpdocker-nginx-1
12eb118332d1       mpdocker-celepy   "./celery.sh"          16 seconds ago    Up 15 seconds     0.0.0.0:15000->15000/tcp  mpdocker-celepy-1
b3a389c98404       mpdocker-server   "./api.sh"            16 seconds ago    Up 15 seconds     0.0.0.0:6379->6379/tcp  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
PS D:\IGI\253502_KRASOV_11\IGI\LR2>

```

```

Windows PowerShell
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker network connect new_big_net bcfc8814ee48
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker network connect new_big_net 12eb118332d1
PS D:\IGI\253502_KRASOV_11\IGI\LR2>

```

```

Windows PowerShell
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker network inspect new_big_net
[{"Name": "new_big_net", "Id": "7a693e790cd174a5f95beadd614f515511254320eb18c4433e4ea890efae86f", "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": "17bedeb40e6422bdb98ab4385f15a3228e5a5bacaaaf7e19e0259543c87bf7389", "MacAddress": "02:42:ac:16:00:94", "IPv4Address": "172.22.0.4/16", "IPv6Address": ""}, "6a43596f16d33f5balcf512ad04d785d48ae607927c8782a068721fb7ef0fbe": {"Name": "mpdocker-redis-1", "EndpointID": "e29e066cad37525927afe6bf8d3603ac0036ae2c4e82fe15f988c7cc6353277", "MacAddress": "02:42:ac:16:00:03", "IPv4Address": "172.22.0.3/16", "IPv6Address": ""}, "bcf88b14ee488ce1db1586318fe220f74b03d05715abe9912ba0ca4d51b39101": {"Name": "mpdocker-nginx-1", "EndpointID": "6a9c9cfb6a2589f0b07187cd539c303611716e4782805fefbe631dae8c8402e0e", "MacAddress": "02:42:ac:16:00:02"}, "ba0a965ea176": {"Name": "my_script_net", "EndpointID": "17bedeb40e6422bdb98ab4385f15a3228e5a5bacaaaf7e19e0259543c87bf7389", "MacAddress": "02:42:ac:16:00:94", "IPv4Address": "172.21.0.1/16", "IPv6Address": ""}}], PS D:\IGI\253502_KRASOV_11\IGI\LR2>

```

Зайдём в оболочку одного контейнера и пропингуем 2 остальных:

```
Windows PowerShell D:\IGI\253502_KRASOV_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_KRASOV_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=0.097 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 D:\IGI\253502_KRASOV_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_KRASOV_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-32jkbtdkgqu26c6b1u9u2rjc 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_KRASOV_11\IGI\LR2> docker network create --d overlay my_overlay_net
bzulogm0cuszjp5xbm2w9yyi
PS D:\IGI\253502_KRASOV_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
bzulogm0cusz   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_KRASOV_11\IGI\LR2> |
```

```
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker network inspect my_overlay_net
[ {
    "Name": "my_overlay_net",
    "Id": "bzulogm0cuszjpd5x bm2w9yyi",
    "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_KRASOV_11\IGI\LR2> |
```

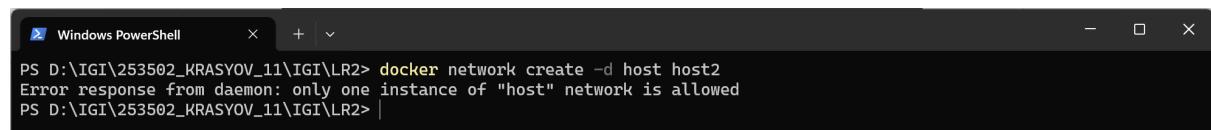
Создадим ещё одну overlay сеть и удалим её

```
Windows PowerShell
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker network create -d overlay overlay_to_delete
uk2xfowuwi7ufipa02i6l1v2u
PS D:\IGI\253502_KRASOV_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
bzulogm0cusz   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_KRASOV_11\IGI\LR2> |
```

```
Windows PowerShell
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker network inspect overlay_to_delete
[ {
    "Name": "overlay_to_delete",
    "Id": "uk2xfowuwi7ufipa02i6l1v2u",
    "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_KRASOV_11\IGI\LR2> |
```

```
Windows PowerShell
PS D:\IGI\253502_KRASOV_11\IGI\LR2> docker network rm overlay_to_delete
overlay_to_delete
PS D:\IGI\253502_KRASOV_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
bzulogm0cusz   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_KRASOV_11\IGI\LR2> |
```

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



```
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> docker network create -d host host2
Error response from daemon: only one instance of "host" network is allowed
PS D:\IGI\253502_KRASYOV_11\IGI\LR2> |
```