

Лабораторная работа №1

Основы работы с Docker и PostgreSQL

Студент(ка): Зырянова Татьяна Евгеньевна

Группа: РИМ – 150950

Предмет: Разработка приложений

Репозиторий ДЗ: <https://github.com/ZyrT12/Application-development-Sber>

Репозиторий: <https://github.com/ZyrT12/java-project-99> - работа с Docker и PostgreSQL в сторонних проектах

Цель работы: освоить фундаментальные концепции и базовые операции Docker: создание образов, запуск контейнеров, управление ими, работа с сетями и томами. На практике закрепить навыки, запустив изолированную базу данных PostgreSQL и подключившись к ней извне.

Задачи:

1. Установить и проверить работу Docker.
2. Изучить базовые команды Docker.
3. Запустить контейнер с PostgreSQL в изолированном режиме.
4. Запустить контейнер с pgAdmin и подключить его к контейнеру с БД через сеть Docker.
5. Подключиться к БД из pgAdmin, создать схему и выполнить запросы.
6. Обеспечить сохранность данных БД с помощью томов Docker

Выполнение заданий лабораторной работы

Часть 0: Установка и проверка Docker

Установка Docker, просмотр версии и проверка (команда docker run hello-world)

```
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber# docker --version
Docker version 27.5.1, build 27.5.1-0ubuntu3~24.04.2
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber# docker run hello-world

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber# -
```

Часть 1: Базовые команды Docker. Работа с образами и контейнерами

1) Просмотр информации через команды docker image, docker ps, docker ps-a.

```
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber# docker images
REPOSITORY      TAG          IMAGE ID       CREATED        SIZE
hello-world    latest        1b44b5a3e06a   7 weeks ago   10.1kB
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber# docker ps
CONTAINER ID   IMAGE          COMMAND         CREATED        STATUS        PORTS     NAMES
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber# docker ps -a
CONTAINER ID   IMAGE          COMMAND         CREATED        STATUS        PORTS     NAMES
f0541fa74262   hello-world    "/hello"       11 minutes ago  Exited (0)  11 minutes ago  elegant_blackwell
4c6eb2b841e4   hello-world    "/hello"       12 hours ago   Exited (0)  12 hours ago  confident_antonelli
5703d22e18f1   hello-world    "/hello"       16 hours ago   Exited (0)  16 hours ago  keen_elion
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber#
```

2) Запуск первого контейнера

```
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber#
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber# docker run -d --name my-nginx -p 8080:80 nginx:alpine
Unable to find image 'nginx:alpine' locally
alpine: Pulling from library/nginx
9824c27679d3: Pull complete
6bc572a349ec: Pull complete
403e3f251637: Pull complete
9adfbbae99cb7: Pull complete
7a8aa46741e18: Pull complete
c9ebef2ff2d2c: Pull complete
a992fbc61ecc: Pull complete
cb1ff4086f82: Pull complete
Digest: sha256:42a516af16b852e33b7682d5ef8acbd5d13fe08fecadc7ed98605ba5e3b26ab8
Status: Downloaded newer image for nginx:alpine
eed4eb24be6cf1bb9f6121496e4a98696ed67e7c84b0941964d9d0ecdba486844
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              PORTS               NAMES
eed4eb24be6c        nginx:alpine      "/docker-entrypoint..."   13 seconds ago    Up 12 seconds   0.0.0.0:8080->80/tcp, [::]:8080->80/tcp   my-nginx
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber#
```

3) Проверка работы контейнера (страница в браузере
<http://localhost:8080>)

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

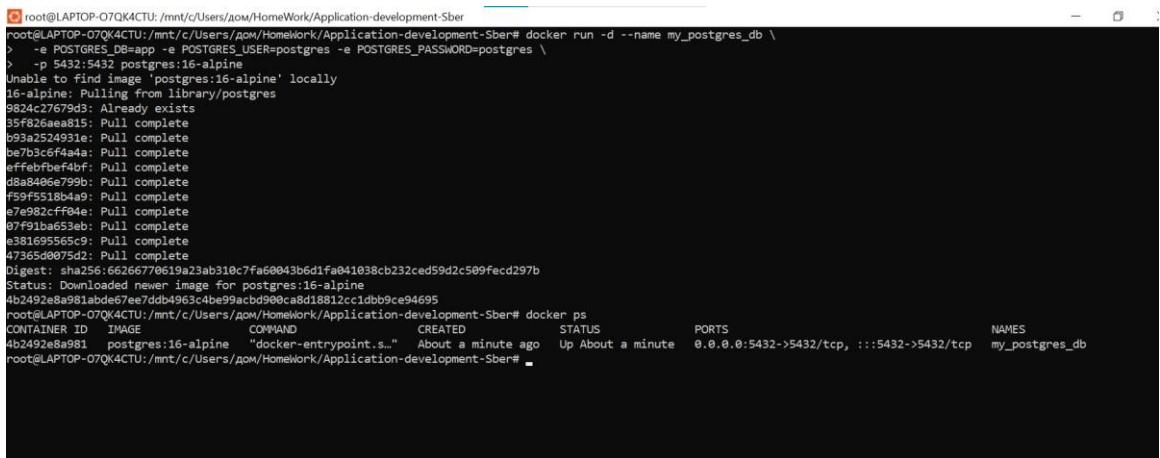
Thank you for using nginx.

4) Удаление контейнера. Проверка, что контейнер удален – не найден после удаления.

```
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber#
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber# docker stop my-nginx && docker rm my-nginx
Error response from daemon: No such container: my-nginx
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber#
```

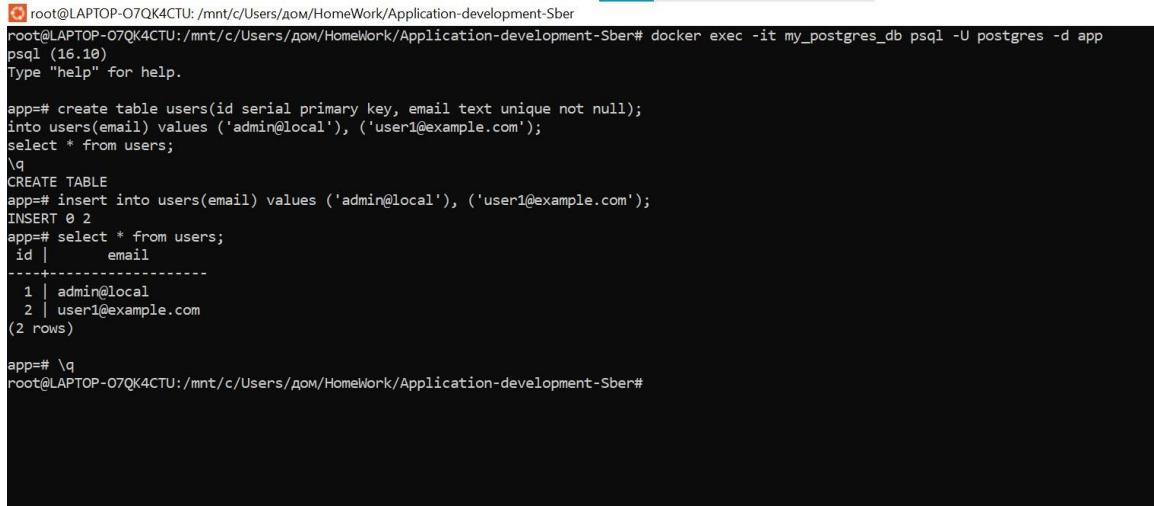
Часть 2: Запуск PostgreSQL в контейнере

1) Запуск контейнера с PostgreSQL. Проверка, что контейнер запущен и слушает порт.



```
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber
> docker run -d --name my_postgres_db \
> -e POSTGRES_DB=app -e POSTGRES_USER=postgres -e POSTGRES_PASSWORD=postgres \
> -p 5432:5432 postgres:16-alpine
Unable to find image 'postgres:16-alpine' locally
16-alpine: Pulling from library/postgres
9824c27679d3: Already exists
35f826aea815: Pull complete
b93a2524931e: Pull complete
be7b3c6f4da4: Pull complete
effebfbef4df: Pull complete
d8a8406e799b: Pull complete
f59f5518b4a9: Pull complete
e7e982cff04e: Pull complete
07f91b6a53ab: Pull complete
e381695565c9: Pull complete
47365d00075d2: Pull complete
Digest: sha256:66266778619a23ab310c7fa60043b6d1fa041038cb232ced59d2c509fecd297b
Status: Downloaded newer image for postgres:16-alpine
4b2492a8a981abde67ee7ddb4963c4be99acbd990:a8d18812cc1dbb9ce94695
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              PORTS               NAMES
4b2492a8a981       "postgres:16-alpine"   "docker-entrypoint.s..."   About a minute ago   Up About a minute   0.0.0.0:5432->5432/tcp, :::5432->5432/tcp   my_postgres_db
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber#
```

2) Подключение к БД из контейнера через psql.



```
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber# docker exec -it my_postgres_db psql -U postgres -d app
psql (16.10)
Type "help" for help.

app=# create table users(id serial primary key, email text unique not null);
into users(email) values ('admin@local'), ('user1@example.com');
select * from users;
\q
CREATE TABLE
app=# insert into users(email) values ('admin@local'), ('user1@example.com');
INSERT 0 2
app=# select * from users;
 id |      email
----+
  1 | admin@local
  2 | user1@example.com
(2 rows)

app=# \q
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber#
```

Часть 3: Подключение к БД через pgAdmin из второго контейнера

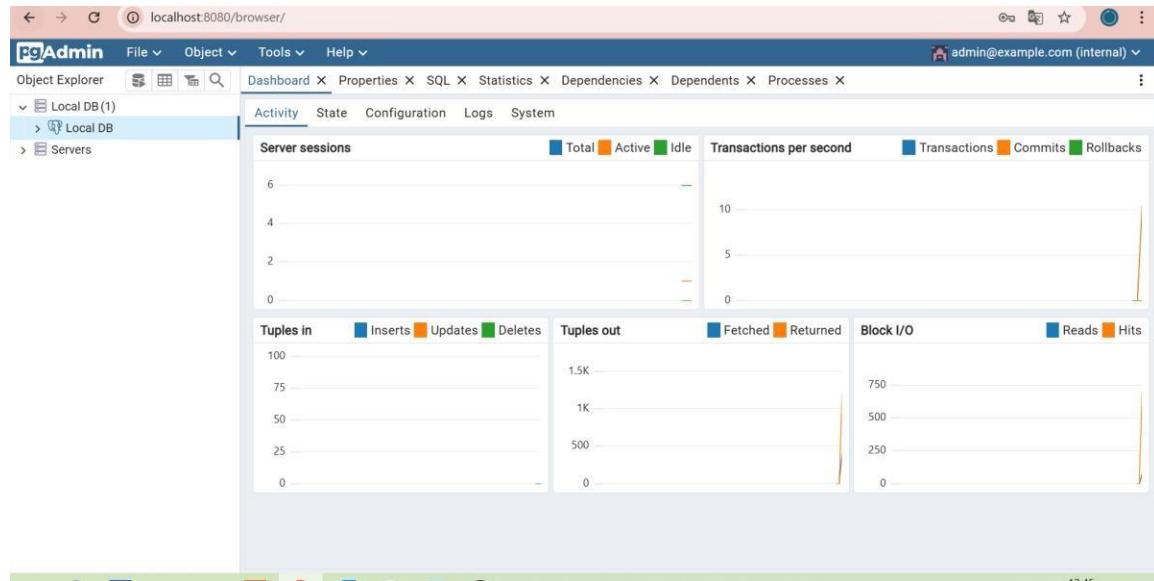
1) Создание сети Docker, подключение контейнера с PostgreSQL к сети.

```
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber# docker network connect my_network my_postgres_db
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber# docker network inspect my_network
[
    {
        "Name": "my_network",
        "Id": "60861e4e9511ffff17bfa16cf747f7cc316e0203803b09f57516c9675a35043",
        "Created": "2025-09-30T12:26:44.724965087+05:00",
        "Scope": "local",
        "Driver": "bridge",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": {},
            "Config": [
                {
                    "Subnet": "172.18.0.0/16",
                    "Gateway": "172.18.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {
            "4b2492e8a981abde67ee7ddb4963c4be99acbd900ca8d18812cc1dbb9ce94695": {
                "Name": "my_postgres_db",
                "EndpointID": "edc1841ad8812ed2e81c54eb0b68731d5d82d8c0c35da05fa7b742b406d4f7fe",
                "MacAddress": "02:42:ac:12:00:02",
                "IPv4Address": "172.18.0.2/16",
                "IPv6Address": ""
            }
        },
        "Options": {},
        "Labels": {}
    }
]
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber#
```

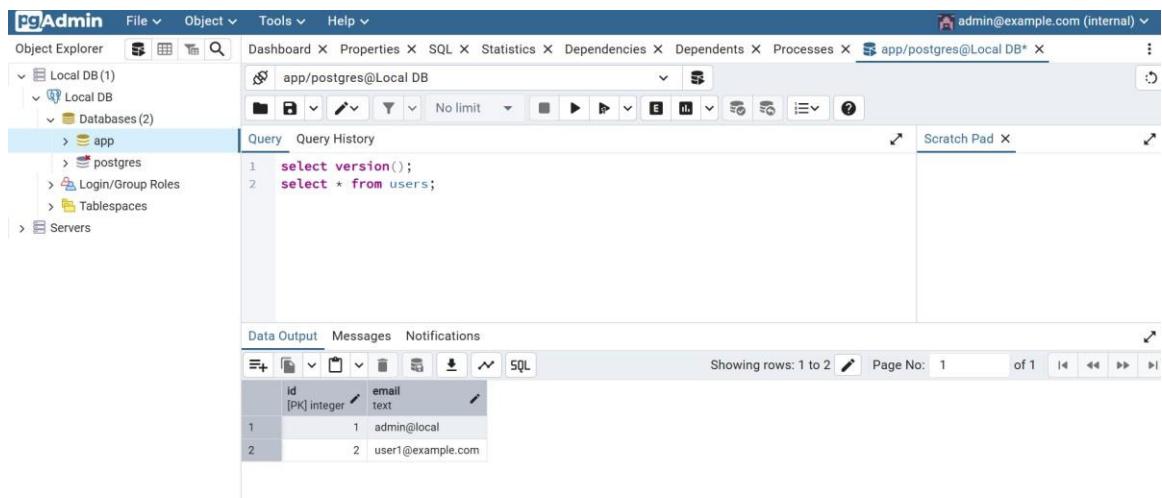
3) Запуск pgAdmin в той же сети.

```
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber# docker rm -f my_pgadmin 2>/dev/null || true
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber# docker run -d --name my_pgadmin --network my_network \
-e PGADMIN_DEFAULT_EMAIL=admin@example.com \
-e PGADMIN_DEFAULT_PASSWORD=admin \
-p 8080:80 dpage/pgadmin4:8
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber# docker run -d --name my_pgadmin --network my_network \
> -e PGADMIN_DEFAULT_EMAIL=admin@example.com \
> -e PGADMIN_DEFAULT_PASSWORD=admin \
> -p 8080:80 dpage/pgadmin4:8
Unable to find image 'dpage/pgadmin4:8' locally
8: Pulling from dpage/pgadmin4
38a8310d387e: Pull complete
087843ea2956: Pull complete
6db836a75a2d: Pull complete
5d0e4706d10: Pull complete
31ebcef82521: Pull complete
91551c39a7c3: Pull complete
210d55276a54: Pull complete
0f3a11d54a10: Pull complete
ddc3056dbb91: Pull complete
545d1f431ff52: Pull complete
48449e1741e8: Pull complete
1213e95defd0: Pull complete
55c17a7b26f0: Pull complete
31af12c6548e: Pull complete
93a2e5af292e: Pull complete
609a98bd4f87: Pull complete
Digest: sha256:8a68677a97b8c8d1427dc915672a26d2c4a04376916a68256f53d669d6171be7
Status: Downloaded newer image for dpage/pgadmin4:8
5f6f17ff52ff73c31ab07d69a8464c1b9d84147e119602bdcce1e0fe9e0dfc06
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/HomeWork/Application-development-Sber#
```

4.1) Проверка работы контейнера (страница в браузере
<http://localhost:8080>



4.1) Выполнение запроса через Query Tool в pgAdmin



Часть 4: Сохранение данных с помощью Томов (Volumes)

1-2) Остановка и удаление текущего контейнера. Создание тома для хранения данных БД.

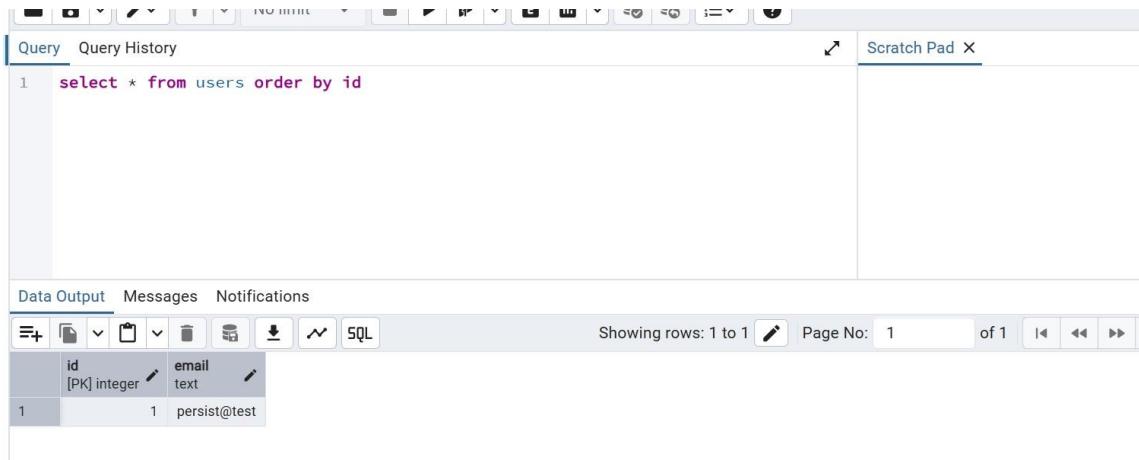
```
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/Homework/Application-development-Sber# docker stop my_postgres_db && docker rm my_postgres_db
my_postgres_db
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/Homework/Application-development-Sber# docker volume create postgres_data
postgres_data
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/Homework/Application-development-Sber# docker run -d --name my_postgres_db --network my_network \
> -e POSTGRES_DB=app -e POSTGRES_USER=postgres -e POSTGRES_PASSWORD=postgres \
> -p 5432:5432 -v postgres_data:/var/lib/postgresql/data postgres:16-alpine
5715a2f253f7956246ccfbef791f679294eb2c4441b1926f181b24b26e32d99
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/Homework/Application-development-Sber# docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
5715a2f253f7 postgres:16-alpine "docker-entrypoint.s..." 22 seconds ago Up 21 seconds 0.0.0.0:5432->5432/tcp, :::5432->5432/tcp my_postgres_db
99841f341527 dpage/pgadmin4:8 "/entrypoint.sh" 20 minutes ago Up 20 minutes 443/tcp, 0.0.0.0:8080->80/tcp, [::]:8080->80/tcp my_pgadmin
f0541fa74262 hello-world "/hello" 2 hours ago Exited (0) 2 hours ago elegant_blackwell
4c6eb2b841e4 hello-world "/hello" 14 hours ago Exited (0) 14 hours ago confident_antonei
li
5703d22e18f1 hello-world "/hello" 18 hours ago Exited (0) 18 hours ago keen_elion
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/Homework/Application-development-Sber#
```

3-4) Запуск нового контейнера с PostgreSQL, подключив том.
Проверка сохранности данных.

```
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/Homework/Application-development-Sber# docker exec -it my_postgres_db \
1> \l
1>   pos>   psql -U postgres -d app -c \
1>   > "create table if not exists users(id serial primary key, email text unique not null);"
CREATE TABLE
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/Homework/Application-development-Sber# docker exec -it my_postgres_db \
1>   psql -U postgres -d app -c \
1>   > "insert into users(email) values ('persist@test');"
INSERT 0 1
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/Homework/Application-development-Sber# docker restart my_postgres_db
my_postgres_db
root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/Homework/Application-development-Sber# docker exec -it my_postgres_db \
1>   psql -U postgres -d app -c "select * from users order by id;"
1>   id | email
1>   -----
1>   1 | persist@test
(1 row)

root@LAPTOP-07QK4CTU:/mnt/c/Users/дом/Homework/Application-development-Sber#
```

4) Проверка таблицы Users



The screenshot shows the pgAdmin interface. In the top-left pane, there is a 'Query History' tab with one entry: 'select * from users order by id'. Below it is a 'Scratch Pad' tab. The bottom half of the screen displays the results of the query in a table. The table has two columns: 'id' and 'email'. The 'id' column is labeled '[PK] integer' and the 'email' column is labeled 'text'. There is one row with values: '1' and 'persist@test'. The table has standard pgAdmin column headers and a toolbar with various icons.

	id [PK] integer	email text
1	1	persist@test

Часть 5: Перенос конфигурации контейнеров в docker-compose.yaml – файл на GitHub

Вывод

В ходе работы я разобралась, как запускать контейнеры, работать с PostgreSQL внутри Docker, подключаться к БД через pgAdmin, использовать тома для сохранения данных и объединять контейнеры в одну сеть. Также научилась оформлять конфигурацию через docker-compose.

Ответы на вопросы

1. Что такое Docker?

Платформа контейнеризации, позволяющая упаковывать приложение и все его зависимости в изолированные среды (контейнеры) для повторяемого запуска на любой машине.

2. Для чего нужны тома и сети Docker?

Тома обеспечивают постоянное хранение данных за пределами жизненного цикла контейнера; сети позволяют контейнерам безопасно находить и взаимодействовать друг с другом по именам.

3. Как подключиться к контейнеру и выполнить в нём команды?

Использовать docker exec -it <container> <команда>, например:
docker exec -it my_postgres_db psql -U postgres.

4. Для чего нужен pgAdmin?

Это графический веб-интерфейс для администрирования PostgreSQL: создание и изменение БД, таблиц, пользователей, выполнение SQL-запросов и просмотр результатов.