

```
ilyamzy@ilyamzy:~$ docker -v
Docker version 28.0.1, build 068a01e
```

```
ilyamzy@ilyamzy:~$ docker run --name getstart -d docker/getting-started
aa7d6e79a74eea39037c1495a30b5bc5197dad7f099ef9a4a614e4f855f85bc0
```

```
ilyamzy@ilyamzy:~$ docker inspect getstart
```

```
{
  "NetworkSettings": {
    "Bridge": "",
    "SandboxID": "9e3c841088be21bd2c32999b6795bef36047a6127914e66d4347cdf4ddd9ce93",
    "SandboxKey": "/var/run/docker/netns/9e3c841088be",
    "Ports": {
      "80/tcp": null
    },
    "NetworkInterface": "eth0"
  },
  "Mounts": [
    {
      "Type": "bind",
      "Source": "/var/run/docker/netns/9e3c841088be",
      "Destination": "/var/run/docker/netns/9e3c841088be",
      "Mode": "rw",
      "ReadOnly": false,
      "Propagation": "rprivate",
      "MountOptions": []
    }
  ],
  "Labels": {}
}
```

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 80:80 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 80:80 - map port 80 of the host to port 80 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 80:80 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, remove, etc.).

To access the dashboard, follow the instructions in the [Docker Desktop manual](#). If you open the dashboard now, you will see this tutorial running! The container name (`jolly_bouman` below) is a randomly created name. So, you'll most likely have a different name.

**Table of contents**

- The command you just ran
- The Docker Dashboard
- What is a container?
- What is a container image?

Открыть

GeometricLib

~/353502\_MALINOUSKI\_19/IGI/LR2

```
FROM python:3.10-alpine

WORKDIR /app

COPY geometric_lib /app/geometric_lib
COPY script.py /app/script.py

CMD ["python", "script.py"]
```

```
ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2$ docker build -t geometric-lib-script -f GeometricLib .
[+] Building 0.9s (9/9) FINISHED
=> [internal] load build definition from GeometricLib
=> [internal] load dockerfile: 1778
=> [internal] load metadata for docker.io/library/python:3.10-alpine
=> [internal] load dockerignore
=> [internal] transferring context: 28
=> [1/4] FROM docker.io/library/python:3.10-alpine@sha256:4c4097e46608e9b9e25a486e3b72d62bac8947a3caa2ef142c2e8c8029a12b21
=> [internal] load build context
=> [internal] transferring context: 1.59kB
=> CACHED [2/4] WORKDIR /app
=> CACHED [3/4] COPY geometric_lib /app/geometric_lib
=> [4/4] COPY script.py /app/script.py
=> exporting to image
=> exporting layers
=> writing image sha256:eecabelf62bdc1dd8b792174e21140924778ad9e4458d3461cfc04852535
=> pushing to docker.io/library/geometric-lib-script
=> pushing layers
```

```
ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2$ docker run -it geometric-lib-script
```

Какая фигура вас интересует?

- 1) Круг
- 2) Квадрат

1

Что вы хотите узнать?

- 1) Периметр
- 2) Площадь

1

Введите радиус круга:

123123

Периметр круга: 773604.6245758722!

Открыть ▼
+

Dockerfile
~/353502\_MALINOUSKI\_19/IGI/LR2/... pose-react-nodejs-postgres/server

GeometricLib	database.js	Dockerfile	GeomtricLib	Geoc
--------------	-------------	------------	-------------	------

```

FROM node:18
WORKDIR /server
COPY src/ /server/src
COPY prisma/ /server/prisma
COPY package.json /server
RUN npm install
RUN npx prisma generate

```

Открыть ▾



## Dockerfile

~/353502\_MALINOUSKI\_19/IGI/LR2... mpose-react-nodejs-postgres/client

on

Dockerfile

package-lock.js

docker-compose

Dockerfile

```
FROM node:18
ARG VITE_SERVER_URL=http://127.0.0.1:7999
ENV VITE_SERVER_URL=$VITE_SERVER_URL
WORKDIR /client
COPY public/ /client/public
COPY src/ /client/src
COPY index.html /client/
COPY package.json /client/
COPY vite.config.js /client/
RUN npm install
RUN npm run build
```

```
services:
  postgres:
    container_name: database
    ports:
      - "5431:5432"
    image: postgres
    environment:
      POSTGRES_USER: "${POSTGRES_USER}"
      POSTGRES_PASSWORD: "${POSTGRES_PASSWORD}"
      POSTGRES_DB: ${POSTGRES_DB}
    volumes:
      - ./docker_test_db:/var/lib/postgresql/data
    healthcheck:
      test: ["CMD-SHELL", "sh -c 'pg_isready -U ${POSTGRES_USER} -d ${POSTGRES_DB}'"]
      interval: 5s
      timeout: 60s
      retries: 5
      start_period: 80s
  server:
    container_name: server
    build:
      context: ./server
      dockerfile: Dockerfile
    ports:
      - "7999:8000"
    command: bash -c "npm run prisma migrate reset --force && npm start"
    environment:
      DATABASE_URL: "${DATABASE_URL}"
      PORT: "${SERVER_PORT}"
    depends_on:
      postgres:
        condition: service_healthy
  client:
    container_name: client
    build:
      context: ./client
      dockerfile: Dockerfile
    command: bash -c "npm run preview"
    ports:
      - "4172:4173"
    depends_on:
      - server
```

NETWORK ID	NAME	DRIVER	SCOPE
b1dfe61fa74f	bridge	bridge	local
f2a8bf956e47	docker-compose-react-nodejs-postgres_default	bridge	local
c674bbe2481a	fullstack-react-node-postgresql_fullstack-network	bridge	local
e81d17288974	host	host	local
401f8689ad90	none	null	local
121012aa3d3d	pern-stack_default	bridge	local
425f9b55c90a	starter-postgres-express-react_default	bridge	local

```

ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker network inspect bridge
[
  {
    "Name": "bridge",
    "Id": "b1dfe61fa74f7c1cb2ebd62384247b8e5c023a8dbc633adf0295c9fe9951d0c8",
    "Created": "2025-03-06T02:05:54.540958737+03:00",
    "Scope": "local",
    "Driver": "bridge",
    "EnableIPv6": false,
    "IPAM": {
      "Driver": "default",
      "Options": null,
      "Config": [
        {
          "Subnet": "172.18.0.0/16",
          "Gateway": "172.18.0.1"
        }
      ]
    },
    "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": {}
  }
]

```

```

ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker network inspect none
[
  {
    "Name": "none",
    "Id": "401f8689ad904ce5d68d47c1cf78e1c101edde48f0c5896a961bd5a8aec33a07",
    "Created": "2025-02-27T18:04:11.152425625+03:00",
    "Scope": "local",
    "Driver": "null",
    "EnableIPv6": false,
    "IPAM": {
      "Driver": "default",
      "Options": null,
      "Config": []
    },
    "Internal": false,
    "Attachable": false,
    "Ingress": false,
    "ConfigFrom": {
      "Network": ""
    },
    "ConfigOnly": false,
    "Containers": {},
    "Options": {},
    "Labels": {}
  }
]

```

```

ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker network inspect host
[
  {
    "Name": "host",
    "Id": "e81d17288974f6651eda3fc77ebc93fe22444fcbc3ee3a05318f1ef4c4d16eee",
    "Created": "2025-02-27T18:04:11.165516271+03:00",
    "Scope": "local",
    "Driver": "host",
    "EnableIPv6": false,
    "IPAM": {
      "Driver": "default",
      "Options": null,
      "Config": []
    },
    "Internal": false,
    "Attachable": false,
    "Ingress": false,
    "ConfigFrom": {
      "Network": ""
    },
    "ConfigOnly": false,
    "Containers": {},
    "Options": {},
    "Labels": {}
  }
]

```

```

ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker network create -
-driver bridge my_bridge

```

```

ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker network ls

```

NETWORK ID	NAME	DRIVER	SCOPE
b1dfe61fa74f	bridge	bridge	local
f2a8bf956e47	docker-compose-react-nodejs-postgres_default	bridge	local
c674bbe2481a	fullstack-react-node-postgresql_fullstack-network	bridge	local
e81d17288974	host	host	local
91d5c913571f	my_bridge	bridge	local
401f8689ad90	none	null	local
121012aa3d3d	pern-stack_default	bridge	local
425f9b55c90a	starter-postgres-express-react_default	bridge	local

```

ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker run -d --name my
_container --network my_bridge alpine sleep 1000
Unable to find image 'alpine:latest' locally
latest: Pulling from library/alpine
f18232174bc9: Already exists
Digest: sha256:a8560b36e8b8210634f77d9f7f9efd7ffa463e380b75e2e74aff4511df3ef88c
Status: Downloaded newer image for alpine:latest
efc8f836baa730e2e108232931204037e634dedf7eacced24b6f3b7e747c2a74
ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker inspect my_conta
iner

```

```

"NetworkSettings": {
  "Bridge": "",
  "SandboxID": "e923bba5eefb9bbfec0220b0b810a6a8aad81c508d19abb262699f82db0826cc",
  "HairpinMode": false,
  "LinkLocalIPv6Address": "",
  "LinkLocalIPv6PrefixLen": 0,
  "Ports": {},
  "SandboxKey": "/var/run/docker/netns/e923bba5eefb",
  "SecondaryIPAddresses": null,
  "SecondaryIPv6Addresses": null,
  "EndpointID": "",
  "Gateway": "",
  "GlobalIPv6Address": "",
  "GlobalIPv6PrefixLen": 0,
  "IPAddress": "",
  "IPPrefixLen": 0,
  "IPv6Gateway": "",
  "MacAddress": "",
  "Networks": {
    "my_bridge": {
      "IPAMConfig": null,
      "Links": null,
      "Aliases": [
        "efc8f836baa7"
      ],
      "NetworkID": "91d5c913571fcd4bc25d3049c30abf4851bd035105aa5547574d9ea031a77581",
      "EndpointID": "d6aee7b27d20af8604fc651ac8fb63d341695246a82d754ccfb4aa3cca41a3dd",
      "Gateway": "172.23.0.1",
      "IPAddress": "172.23.0.2",
      "IPPrefixLen": 16,
      "IPv6Gateway": "",
      "GlobalIPv6Address": "",
      "GlobalIPv6PrefixLen": 0,
      "MacAddress": "02:42:ac:17:00:02",
      "DriverOpts": null
    }
  }
}

```

```

ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker network disconnec
ct my_bridge my_container
ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS          NAMES
efc8f836baa7   alpine    "sleep 1000"            About a minute ago    Up About a minute           my_container

```

```

"NetworkSettings": {
  "Bridge": "",
  "SandboxID": "e923bba5eefb9bbfec0220b0b810a6a8aad81c508d19abb262699f82db0826cc",
  "HairpinMode": false,
  "LinkLocalIPv6Address": "",
  "LinkLocalIPv6PrefixLen": 0,
  "Ports": {},
  "SandboxKey": "/var/run/docker/netns/e923bba5eefb",
  "SecondaryIPAddresses": null,
  "SecondaryIPv6Addresses": null,
  "EndpointID": "",
  "Gateway": "",
  "GlobalIPv6Address": "",
  "GlobalIPv6PrefixLen": 0,
  "IPAddress": "",
  "IPPrefixLen": 0,
  "IPv6Gateway": "",
  "MacAddress": "",
  "Networks": {}
}

```

```

ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker network create -
-driver bridge my_bridge2
6581887a2455af79fc8b86238d8b844e8e2c766a9471fb47e238ccd866e828cc
ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker network ls

```

NETWORK ID	NAME	DRIVER	SCOPE
b1dfe61fa74f	bridge	bridge	local
f2a8bf956e47	docker-compose-react-nodejs-postgres_default	bridge	local
c674bbe2481a	fullstack-react-node-postgresql_fullstack-network	bridge	local
e81d17288974	host	host	local
91d5c913571f	my_bridge	bridge	local
6581887a2455	my_bridge2	bridge	local
401f8689ad90	none	null	local
121012aa3d3d	pern-stack_default	bridge	local
425f9b55c90a	starter-postgres-express-react_default	bridge	local

```

ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker run -d --name co
ntainer1 --network my_bridge2 alpine sleep 1000
docker run -d --name container2 --network my_bridge2 alpine sleep 1000
docker run -d --name container3 --network my_bridge2 alpine sleep 1000
67268fd2f9f9a134f926a3c93e88f36c01eaf05605d5b7c0225fcf17d9c102db
4402bb6f688f8bc5c4918fad02fd7fb6f81abd70d0d94c8543b0829d9851930f
034aafe58ab2c931a9aca33b43a61b3d34cfb1163972c46487c0087817f9e436

```

```

"Networks": {
  "my_bridge2": {
    "IPAMConfig": null,
    "Links": null,
    "Aliases": [
      "67268fd2f9f9"
    ],
    "NetworkID": "6581887a2455af79fc8b86238d8b844e8e2c766a9471fb47e238ccd866e828cc",
    "EndpointID": "d63a3c9050a8f8f0c5592577d62ad003ab13e9323cb9300f8b1da73b1b56f303",
    "Gateway": "172.24.0.1",
    "IPAddress": "172.24.0.2",
    "IPPrefixLen": 16,
    "IPv6Gateway": "",
    "GlobalIPv6Address": "",
    "GlobalIPv6PrefixLen": 0,
    "MacAddress": "02:42:ac:18:00:02",
    "DriverOpts": null
  }
}

```



```

    "Networks": {
      "my_bridge2": {
        "IPAMConfig": null,
        "Links": null,
        "Aliases": [
          "4402bb6f688f"
        ],
        "NetworkID": "6581887a2455af79fc8b86238d8b844e8e2c766a9471fb47e238ccd866e828cc",
        "EndpointID": "5946d8720c0b7465a929fe017e91d31fe575f6342ffe515e0b821a0469080e99",
        "Gateway": "172.24.0.1",
        "IPAddress": "172.24.0.3",
        "IPPrefixLen": 16,
        "IPv6Gateway": "",
        "GlobalIPv6Address": "",
        "GlobalIPv6PrefixLen": 0,
        "MacAddress": "02:42:ac:18:00:03",
        "DriverOpts": null
      }
    }
  }
}
]

```

```

    "Networks": {
      "my_bridge2": {
        "IPAMConfig": null,
        "Links": null,
        "Aliases": [
          "034aafe58ab2"
        ],
        "NetworkID": "6581887a2455af79fc8b86238d8b844e8e2c766a9471fb47e238ccd866e828cc",
        "EndpointID": "a4b8c3f445bb30ebb4be7601659b9af11975c14b6028edc2bf7f324fa158db47",
        "Gateway": "172.24.0.1",
        "IPAddress": "172.24.0.4",
        "IPPrefixLen": 16,
        "IPv6Gateway": "",
        "GlobalIPv6Address": "",
        "GlobalIPv6PrefixLen": 0,
        "MacAddress": "02:42:ac:18:00:04",
        "DriverOpts": null
      }
    }
  }
}
]

```

```

ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker exec -it contain
er1 sh

```

```

/ # ping 172.24.0.3
PING 172.24.0.3 (172.24.0.3): 56 data bytes
64 bytes from 172.24.0.3: seq=0 ttl=64 time=0.164 ms
64 bytes from 172.24.0.3: seq=1 ttl=64 time=0.140 ms
64 bytes from 172.24.0.3: seq=2 ttl=64 time=0.131 ms
64 bytes from 172.24.0.3: seq=3 ttl=64 time=0.216 ms
64 bytes from 172.24.0.3: seq=4 ttl=64 time=0.160 ms
64 bytes from 172.24.0.3: seq=5 ttl=64 time=0.141 ms
64 bytes from 172.24.0.3: seq=6 ttl=64 time=0.181 ms
64 bytes from 172.24.0.3: seq=7 ttl=64 time=0.215 ms

```

```

/ # ping 172.24.0.4
PING 172.24.0.4 (172.24.0.4): 56 data bytes
64 bytes from 172.24.0.4: seq=0 ttl=64 time=0.146 ms
64 bytes from 172.24.0.4: seq=1 ttl=64 time=0.131 ms
64 bytes from 172.24.0.4: seq=2 ttl=64 time=0.183 ms
64 bytes from 172.24.0.4: seq=3 ttl=64 time=0.122 ms
64 bytes from 172.24.0.4: seq=4 ttl=64 time=0.182 ms
64 bytes from 172.24.0.4: seq=5 ttl=64 time=0.212 ms

```

```
ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker swarm init
Swarm initialized: current node (y4c50sq8um0i30gjmpl28yna) is now a manager.
```

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

```
docker swarm join --token SWMTKN-1-12crg9w34k7gweql0o4pd3s8d31yu4cearcx5p68txfcf3j3cx-b25w9h9bk0jmtj4mrs
hvdwleo 192.168.1.103:2377
```

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

```
ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker network create -
-driver overlay my_overlay
n7zwhi0lgqc3r3zuls17rb0er
ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$
```

```
ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker network inspect
my_overlay
```

```
[
  {
    "Name": "my_overlay",
    "Id": "n7zwhi0lgqc3r3zuls17rb0er",
    "Created": "2025-03-06T01:26:45.715479997Z",
    "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
  }
]
```

```
ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$
```

```

ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker network create -
-driver overlay my_overlay2
p03ol3qmub71k11fyf8durlv
ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker network inspect
my_overlay2
[
  {
    "Name": "my_overlay2",
    "Id": "p03ol3qmub71k11fyf8durlv",
    "Created": "2025-03-06T01:29:44.39083418Z",
    "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
  }
]

```

```

ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker network ls
NETWORK ID        NAME                                     DRIVER  SCOPE
b1dfe61fa74f     bridge                                bridge  local
f2a8bf956e47     docker-compose-react-nodejs-postgres_default  bridge  local
edeadc67d19      docker_gwbridge                      bridge  local
c674bbe2481a     fullstack-react-node-postgresql_fullstack-network  bridge  local
e81d17288974     host                                 host    local
0fczrpjtnmub     ingress                              overlay  swarm
91d5c913571f     my_bridge                            bridge  local
6581887a2455     my_bridge2                          bridge  local
n7zwhi0lgqc3     my_overlay                          overlay  swarm
p03ol3qmub71     my_overlay2                         overlay  swarm
401f8689ad90     none                                null    local
121012aa3d3d     pern-stack_default                 bridge  local
425f9b55c90a     starter-postgres-express-react_default  bridge  local
ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$
ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker network rm my_ov
erlay2
my_overlay2
ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker network ls
NETWORK ID        NAME                                     DRIVER  SCOPE
b1dfe61fa74f     bridge                                bridge  local
f2a8bf956e47     docker-compose-react-nodejs-postgres_default  bridge  local
edeadc67d19      docker_gwbridge                      bridge  local
c674bbe2481a     fullstack-react-node-postgresql_fullstack-network  bridge  local
e81d17288974     host                                 host    local
0fczrpjtnmub     ingress                              overlay  swarm
91d5c913571f     my_bridge                            bridge  local
6581887a2455     my_bridge2                          bridge  local
n7zwhi0lgqc3     my_overlay                          overlay  swarm
401f8689ad90     none                                null    local
121012aa3d3d     pern-stack_default                 bridge  local
425f9b55c90a     starter-postgres-express-react_default  bridge  local

```

```
ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker network create -  
-driver host my_host  
Error response from daemon: only one instance of "host" network is allowed
```

```
ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker push docker-compose-react-nodejs-pos  
tgres-client:latest  
The push refers to repository [docker.io/library/docker-compose-react-nodejs-postgres-client]  
e97434324e9a: Preparing  
1e41fc496ad4: Preparing  
30860e6e88d2: Preparing  
cd2b6a45a969: Preparing  
7ed19a98ee64: Preparing  
98195222df3e: Waiting  
b19b86d1bbdb: Waiting  
e4857f94a65d: Waiting  
f270deee6385: Waiting  
e129ba4f1574: Waiting  
780658909072: Waiting  
eae0a84b6ca2: Waiting  
4b017a36fd9c: Waiting  
20a9b386e10e: Waiting  
f8217d7865d2: Waiting  
01c9a2a5f237: Waiting  
denied: requested access to the resource is denied  
ilyamzy@ilyamzy:~/353502_MALINOUSKI_19/IGI/LR2/docker-compose-react-nodejs-postgres$ docker push docker-compose-react-nodejs-pos  
tgres-server:latest  
The push refers to repository [docker.io/library/docker-compose-react-nodejs-postgres-server]  
d5121b6f409e: Preparing  
b90d39e678ae: Preparing  
751ea16358fd: Preparing  
32dea0c4127a: Preparing  
1fc178678329: Preparing  
1eced85b5efd: Waiting  
f270deee6385: Waiting  
e129ba4f1574: Waiting  
780658909072: Waiting  
eae0a84b6ca2: Waiting  
4b017a36fd9c: Waiting  
20a9b386e10e: Waiting  
f8217d7865d2: Waiting  
01c9a2a5f237: Waiting  
denied: requested access to the resource is denied
```