Docker files, kontener jako definicja etapu

- Pierwszym krokiem było znalezienie dowolnego repozytorium z kodem , który zawiera Makefile i testy oraz otwartą licencję. Mój wybór padł na kalkulator. Sklonowałam repozytorium.

```
kamila_partyka@Lenovo:~/workspace/MDO2023_INO/INO/GCL2/KP406287/lab02$ git clone https://github.com/alt-romes/programmer-calculator
Cloning into 'programmer-calculator'...
remote: Enumerating objects: 1322, done.
remote: Counting objects: 100% (207/207), done.
remote: Compressing objects: 100% (74/74), done.
remote: Total 1322 (delta 137), reused 177 (delta 129), pack-reused 1115
Receiving objects: 100% (1322/1322), 1.98 MiB | 1.67 MiB/s, done.
Resolving deltas: 100% (809/809), done.
kamila_partyka@Lenovo:~/workspace/MDO2023_INO/INO/GCL2/KP406287/lab02$ ls
programmer-calculator
kamila_partyka@Lenovo:~/workspace/MDO2023_INO/INO/GCL2/KP406287/lab02$ cd programmer-calculator/
```

- Doinstalowałam wymagane biblioteki ncurses. Biblioteka ta umożliwia programistom manipulowanie tekstem i kolorami, tworzenie okien, ramek, przycisków, menu itp. w konsoli tekstowej.

```
kamila_partyka@Lenovo:~/workspace/MDO2023_INO/INO/GCL2/KP406287/lab02/programmer-calculator$ sudo apt-get install libncurses5-dev libncursesw5-dev
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libc-dev-bin libc-devtools libc6-dev libcrypt-dev libncurses-dev libnsl-dev libtirpc-dev linux-libc-dev manpages-dev rpcsvc-proto
Suggested packages:
  glibc-doc ncurses-doc
The following NEW packages will be installed:
  libc-dev-bin libc-devtools libc6-dev libcrypt-dev libncurses-dev libncurses5-dev libncursesw5-dev libnsl-dev libtirpc-dev linux-libc-dev manpages-dev
  rpcsvc-proto
0 upgraded, 12 newly installed, 0 to remove and 35 not upgraded.
Need to get 6612 kB of archives.
After this operation, 29.0 MB of additional disk space will be used.
```

- Zainstalowałam make'a. Skompilowałam kod programu i przeniosłam plik wykonywalny do katalogu osiągalnego przez $PATH.

```
kamila_partyka@Lenovo:~/workspace/MDO2023_INO/INO/GCL2/KP406287/lab02/programmer-calculator$ sudo make install
Installing!
make all
make[1]: Entering directory '/home/kamila_partyka/workspace/MDO2023_INO/INO/GCL2/KP406287/lab02/programmer-calculator'
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -c src/draw.c -o build/draw.o
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -c src/history.c -o build/history.o
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -c src/main.c -o build/main.o
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -c src/numberstack.c -o build/numberstack.o
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -c src/operators.c -o build/operators.o
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -c src/parser.c -o build/parser.o
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -c src/xmalloc.c -o build/xmalloc.o
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -o bin/pcalc build/draw.o build/history.o build/main.o build/numberstack
.o build/operators.o build/parser.o build/xmalloc.o -lncurses #
Executing all complete!
make[1]: Leaving directory '/home/kamila_partyka/workspace/MDO2023_INO/INO/GCL2/KP406287/lab02/programmer-calculator'
```

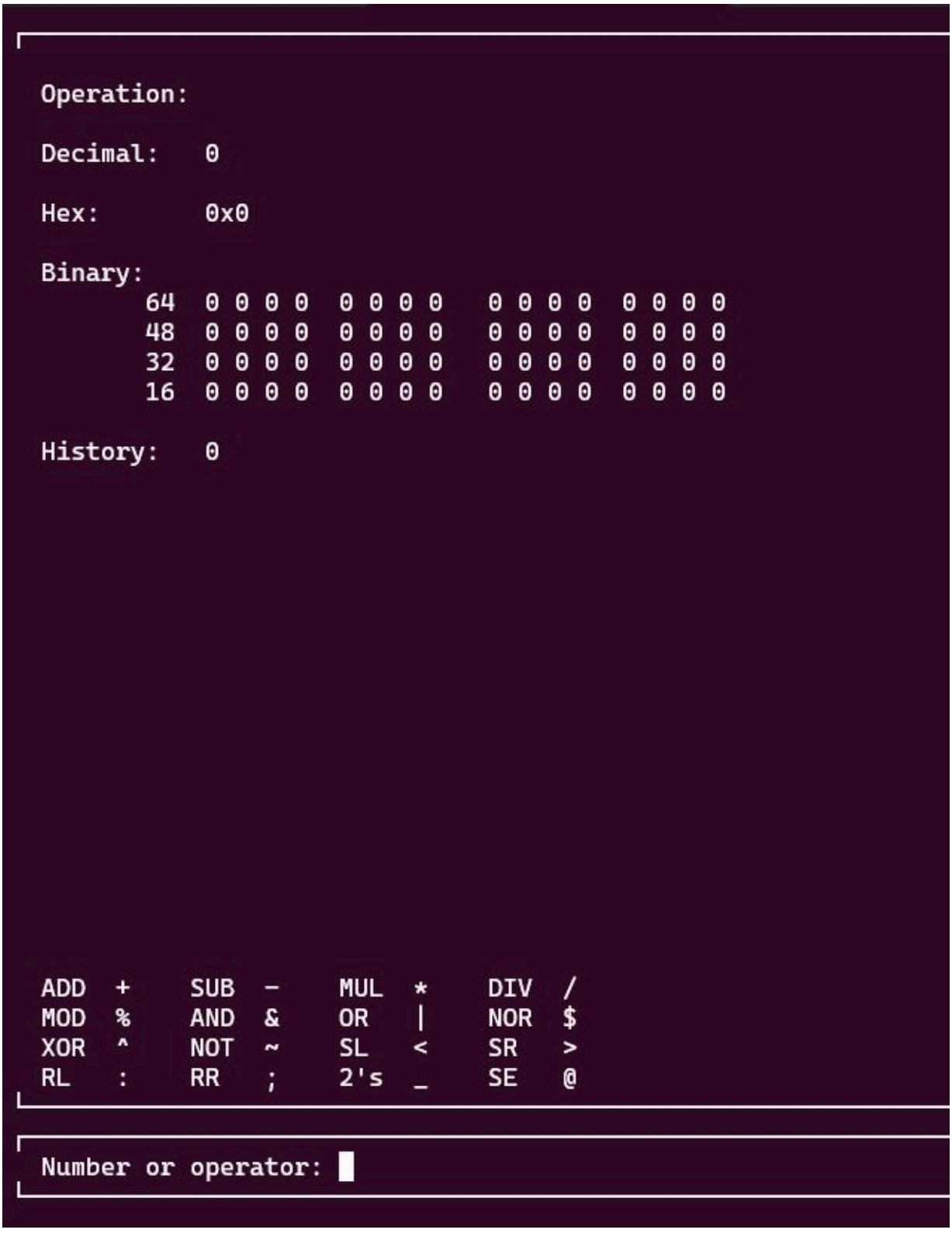
```
kamila_partyka@Lenovo:~/workspace/MDO2023_INO/INO/GCL2/KP406287/lab02/programmer-calculator$ make
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -o bin/pcalc build/draw.o build/history.o build/main.o build/numberstack
.o build/operators.o build/parser.o build/xmalloc.o -lncurses #
Executing all complete!
```

```
kamila_partyka@Lenovo:~/Desktop/lab03/programmer-calculator$ sudo mv -i pcalc /usr/local/bin
```

- Uruchomiłam potrzebne testy.

```
kamila_partyka@Lenovo:~/Desktop/lab03/programmer-calculator$ ./run-tests.sh
All tests passed
kamila_partyka@Lenovo:~/Desktop/lab03/programmer-calculator$ ls
CONTRIBUTING.md  LICENSE  Makefile  README.md  assets  bin  build  docs  how-to-publish.md  include  run-tests.sh  src  tests
```

- A następnie uruchomiłam program.



- Uruchomiłam dockera i sprawdziłam dostępne obrazy:

- Uruchomiłam kontener i podłączyłam się do niego, aby rozpocząć interaktywną pracę oraz zainstalowałam gita. Interaktywna praca pozwala na wykonywanie poleceń wewnątrz kontenera i podejmowanie akcji w czasie rzeczywistym, co jest szczególnie przydatne w przypadku

testowania i rozwiązywania problemów w aplikacjach uruchomionych w kontenerze Docker.

```
kamila_partyka@Lenovo:~/Desktop/lab03/programmer-calculator$ sudo docker run --interactive --tty ubuntu sh
# git --version
sh: 1: git: not found
# apt-get install git
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
E: Unable to locate package git
# apt-get update
Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:2 http://archive.ubuntu.com/ubuntu jammy InRelease [270 kB]
Get:3 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [900 kB]
Get:5 http://archive.ubuntu.com/ubuntu jammy-backports InRelease [107 kB]
Get:6 http://archive.ubuntu.com/ubuntu jammy/main amd64 Packages [1792 kB]
Get:7 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [829 kB]
Get:8 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [23.2 kB]
Get:9 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [868 kB]
Get:10 http://archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [17.5 MB]
Get:11 http://archive.ubuntu.com/ubuntu jammy/restricted amd64 Packages [164 kB]
Get:12 http://archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [266 kB]
Get:13 http://archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [28.6 kB]
Get:14 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1199 kB]
Get:15 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1138 kB]
Get:16 http://archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [885 kB]
Get:17 http://archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [22.4 kB]
Get:18 http://archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [49.0 kB]
Fetched 26.2 MB in 24s (1078 kB/s)
Reading package lists... Done
# apt-get install -y git
Reading package lists... Done
```

- Sklonowałam wcześniej wybrane repozytorium.

```
# git clone https://github.com/alt-romes/programmer-calculator.git
Cloning into 'programmer-calculator'...
remote: Enumerating objects: 1322, done.
remote: Counting objects: 100% (207/207), done.
remote: Compressing objects: 100% (74/74), done.
remote: Total 1322 (delta 137), reused 177 (delta 129), pack-reused 1115
Receiving objects: 100% (1322/1322), 1.98 MiB | 1.32 MiB/s, done.
Resolving deltas: 100% (809/809), done.
# ls
bin  boot  dev  etc  home  lib  lib32  lib64  libx32  media  mnt  opt  proc  programmer-calculator  root  run  sbin  srv  sys  tmp  usr  var
# cd programmer-calculator
# ls
CONTRIBUTING.md  README.md  how-to-publish.md  src
LICENSE          assets     include           tests
Makefile         docs       run-tests.sh
```

- Zainstalowałam resztę potrzebnych rzeczy.

```
# apt-get install make
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:
  make-doc
The following NEW packages will be installed:
  make
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 180 kB of archives.
After this operation, 426 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu jammy/main amd64 make amd64 4.3-4.1build1 [180 kB]
Fetched 180 kB in 1s (143 kB/s)
debconf: delaying package configuration, since apt-utils is not installed
Selecting previously unselected package make.
(Reading database ... 8352 files and directories currently installed.)
Preparing to unpack .../make_4.3-4.1build1_amd64.deb ...
Unpacking make (4.3-4.1build1) ...
Setting up make (4.3-4.1build1) ...
# ls
CONTRIBUTING.md  README.md  how-to-publish.md  src
LICENSE          assets     include           tests
Makefile         docs       run-tests.sh
```

```
# apt-get install gcc
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu cpp
  cpp-11 fontconfig-config fonts-dejavu-core gcc-11
  gcc-11-base libasan6 libatomic1 libbinutils libc-dev-bin
  libc-devtools libc6-dev libcc1-0 libcrypt-dev
  libctf-nobfd0 libctf0 libdeflate0 libfontconfig1
  libfreetype6 libgcc-11-dev libgd3 libgomp1 libisl23
  libitm1 libjbig0 libjpeg-turbo8 libjpeg8 liblsan0 libmpc3
  libmpfr6 libnsl-dev libpng16-16 libquadmath0 libtiff5
  libtirpc-dev libtsan0 libubsan1 libwebp7 libxpm4
  linux-libc-dev manpages manpages-dev rpcsvc-proto ucf
Suggested packages:
  binutils-doc cpp-doc gcc-11-locales gcc-multilib autoconf
  automake libtool flex bison gdb gcc-doc gcc-11-multilib
  gcc-11-doc glibc-doc libgd-tools man-browser
The following NEW packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu cpp
  cpp-11 fontconfig-config fonts-dejavu-core gcc gcc-11
  gcc-11-base libasan6 libatomic1 libbinutils libc-dev-bin
  libc-devtools libc6-dev libcc1-0 libcrypt-dev
  libctf-nobfd0 libctf0 libdeflate0 libfontconfig1
  libfreetype6 libgcc-11-dev libgd3 libgomp1 libisl23
  libitm1 libjbig0 libjpeg-turbo8 libjpeg8 liblsan0 libmpc3
  libmpfr6 libnsl-dev libpng16-16 libquadmath0 libtiff5
  libtirpc-dev libtsan0 libubsan1 libwebp7 libxpm4
  linux-libc-dev manpages manpages-dev rpcsvc-proto ucf
0 upgraded, 48 newly installed, 0 to remove and 0 not upgraded.
Need to get 55.4 MB of archives.
After this operation, 173 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
```

```
# apt-get install libncurses5-dev libncursesw5-dev
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libncurses-dev
Suggested packages:
  ncurses-doc
The following NEW packages will be installed:
  libncurses-dev libncurses5-dev libncursesw5-dev
0 upgraded, 3 newly installed, 0 to remove and 0 not upgraded.
Need to get 382 kB of archives.
After this operation, 2419 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://archive.ubuntu.com/ubuntu jammy/main amd64 libncurses-dev amd64 6.3-2 [380 kB]
Get:2 http://archive.ubuntu.com/ubuntu jammy/main amd64 libncurses5-dev amd64 6.3-2 [780 B]
Get:3 http://archive.ubuntu.com/ubuntu jammy/main amd64 libncursesw5-dev amd64 6.3-2 [784 B]
Fetched 382 kB in 1s (298 kB/s)
debconf: delaying package configuration, since apt-utils is not installed
Selecting previously unselected package libncurses-dev:amd64.(Reading database ... 13354 files
Preparing to unpack .../libncurses-dev_6.3-2_amd64.deb ...
```

- Następnie uruchomiłam program poleceniem make.

```
# ls
CONTRIBUTING.md  assets  how-to-publish.md  tests
LICENSE          bin     include
Makefile         build   run-tests.sh
README.md        docs    src
# make
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -c src/draw.c -o build/draw.o
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -c src/history.c -o build/history.o
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -c src/main.c -o build/main.o
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -c src/numberstack.c -o build/numberstack.o
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -c src/operators.c -o build/operators.o
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -c src/parser.c -o build/parser.o
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -c src/xmalloc.c -o build/xmalloc.o
undant-decls -Iinclude -o bin/pcalc build/draw.o build/history.o build/main.o build/numberstack.o build/operators.o build/parser
 #
Executing all complete!
```

- Ponownie uruchomiłam testy.

```
# bash ./run-tests.sh
All tests passed
# exit
```

- Później należało stworzyć dwa pliki Dockerfile automatyzujące wcześniej wykonane kroki. Pliki Dockerfile służą do automatyzowania procesu budowania obrazów kontenerów w Dockerze. Pozwalają one na zdefiniowanie kroków, które mają być wykonane podczas budowania obrazu i zapisanie ich w jednym pliku tekstowym. Pierwszy z kontenerów przeprowadza wszystkie kroki aż do builda.

```
 GNU nano 6.2                                                dockerFile1
FROM ubuntu:latest
RUN apt-get update -y
RUN apt-get install git -y
RUN apt-get install make -y
RUN apt-get install gcc -y
RUN apt-get install libncurses5-dev libncursesw5-dev -y
RUN git clone https://github.com/alt-romes/programmer-calculator.git
WORKDIR programmer-calculator
RUN make
```

FROM- oznacza wybranie konkretnego obrazu

RUN- uruchomienie instalacji, klonowania repozytorium i builda

WORKDIR- definiowanie katalogu roboczego kontenera

- Utworzyłam obraz z pliku dockerFile1:

```
kamila_partyka@Lenovo:~/Desktop/lab03/dockerfiles$ sudo docker build -t img1:latest . -f ./dockerFile1
[+] Building 40.4s (13/13) FINISHED
 => [internal] load build definition from dockerFile1
 => => transferring dockerfile: 334B
 => [internal] load .dockerignore
 => => transferring context: 2B
 => [internal] load metadata for docker.io/library/ubuntu:latest
 => [1/9] FROM docker.io/library/ubuntu:latest
 => CACHED [2/9] RUN apt-get update -y
 => CACHED [3/9] RUN apt-get install git -y
 => CACHED [4/9] RUN apt-get install make -y
 => CACHED [5/9] RUN apt-get install gcc -y
 => CACHED [6/9] RUN apt-get install libncurses5-dev libncursesw5-dev -y
 => [7/9] RUN git clone https://github.com/alt-romes/programmer-calculator.git
 => [8/9] WORKDIR programmer-calculator
 => [9/9] RUN make
 => exporting to image
 => => exporting layers
 => => writing image sha256:6dcfa9b1e60defdd02d95532579ab4b9f346b6c041b360e7df2f9be183f2d14f
 => => naming to docker.io/library/img1:latest
```

```
kamila_partyka@Lenovo:~/Desktop/lab03/dockerfiles$ sudo docker images
REPOSITORY    TAG       IMAGE ID       CREATED              SIZE
img1          latest    6dcfa9b1e60d   About a minute ago   368MB
ubuntu        latest    08d22c0ceb15   12 days ago          77.8MB
hello-world   latest    feb5d9fea6a5   18 months ago        13.3kB
```

- Utworzyłam drugi plik Dockerfile, w którym wybieram aktualny obraz utworzony wcześniej, następnie uruchamiane są testy.

```
 GNU nano 6.2                                                dockerFile2

FROM img1:latest
RUN bash ./run-tests.sh
```

- Utworzyłam obraz z pliku dockerFile2:

```
kamila_partyka@Lenovo:~/Desktop/lab03/dockerfiles$ sudo docker build -t img2:latest . -f ./dockerFile2
[+] Building 24.8s (6/6) FINISHED
 => [internal] load build definition from dockerFile2
 => => transferring dockerfile: 86B
 => [internal] load .dockerignore
 => => transferring context: 2B
 => [internal] load metadata for docker.io/library/img1:latest
 => [1/2] FROM docker.io/library/img1:latest
 => [2/2] RUN bash ./run-tests.sh
 => exporting to image
 => => exporting layers
 => => writing image sha256:6d663f21191a3cf3fa701896e15757d3549a94ee5ca867817dcc9d4c52ae858f
 => => naming to docker.io/library/img2:latest
```

```
kamila_partyka@Lenovo:~/Desktop/lab03/dockerfiles$ sudo docker images
REPOSITORY    TAG       IMAGE ID       CREATED          SIZE
img2          latest    6d663f21191a   38 seconds ago   368MB
img1          latest    6dcfa9b1e60d   7 minutes ago    368MB
ubuntu        latest    08d22c0ceb15   12 days ago      77.8MB
hello-world   latest    feb5d9fea6a5   18 months ago    13.3kB
```

ZACHOWANIE STANU

- Przygotowano woluminy wejściowy i wyjściowy o nazwach volin oraz volout. Wolumin wejściowy to pojemnik na dane, które będą przetwarzane, natomiast wolumin wyjściowy to miejsce, gdzie

przetworzone dane zostaną zapisane. Wykorzystano do tego komendę sudo docker volume create.

```
kamila_partyka@Lenovo:~/Desktop/lab04$ sudo docker volume create --name volin
[sudo] password for kamila_partyka:
volin
kamila_partyka@Lenovo:~/Desktop/lab04$ sudo docker volume create --name volout
volout
kamila_partyka@Lenovo:~/Desktop/lab04$ docker volume ls
DRIVER    VOLUME NAME
local     a8ce4f54460a841d8d506ab77617bbdabbd8506c15fce4ead571e7cdc0e83384
local     volin
local     volout
```

- Następnie uruchomiono kontener z woluminami za pomocą sudo docker run. i wyświetlono odpowiednie katalogi vin i vout.

-„it" oznacza, że kontener będzie uruchomiony w trybie interaktywnym i utworzy terminal w kontenerze

```
kamila_partyka@Lenovo:~/Desktop/lab04$ sudo docker run -it --name first --mount source=volin
,target=/vin --mount source=volout,target=/vout ubuntu
root@0a561e3ddb16:/# ls
bin  boot  dev  etc  home  lib  lib32  lib64  libx32  media  mnt  opt  proc  root  run  sbin
  srv  sys  tmp  usr  var  vin  vout
root@0a561e3ddb16:/# ls v*
var:
backups  cache  lib  local  lock  log  mail  opt  run  spool  tmp

vin:

vout:
root@0a561e3ddb16:/#
```

- Wyświetlono szczegóły o obu woluminach oraz ścieżkę, która będzie potrzebna w następnym kroku. Wykorzystano komendę sudo docker volume inspect. Komenda ta wyświetli JSON zawierający różne informacje o woluminie, takie jak nazwa, typ, opcje konfiguracyjne, punkty montowania i inne.

Możemy również dodać opcję --format do tej komendy, aby określić format wyjściowy.

```
kamila_partyka@Lenovo:~/Desktop/lab04$ sudo docker inspect volin
[
    {
        "CreatedAt": "2023-04-01T18:10:16Z",
        "Driver": "local",
        "Labels": {},
        "Mountpoint": "/var/lib/docker/volumes/volin/_data",
        "Name": "volin",
        "Options": {},
        "Scope": "local"
    }
]
kamila_partyka@Lenovo:~/Desktop/lab04$ sudo docker volume inspect volout
[
    {
        "CreatedAt": "2023-04-01T18:10:22Z",
        "Driver": "local",
        "Labels": {},
        "Mountpoint": "/var/lib/docker/volumes/volout/_data",
        "Name": "volout",
        "Options": {},
        "Scope": "local"
    }
]
```

- Z poziomu roota skopiowano repozytorium na wolumin wejściowy volin z wykorzystaniem ścieżki z poprzedniego kroku oraz polecenia git clone.

```
root@Lenovo:/var/lib# ls
PackageKit          docker-desktop  pam           snmp                  update-notifier
apt                 dpkg            plymouth      sudo                  upower
aspell              emacsen-common  polkit-1      systemd               usbutils
colord              ghostscript     private       ubuntu-advantage      vim
command-not-found   git             python        ubuntu-fan            xfonts
dbus                ispell          saned         ubuntu-release-upgrader xkb
dhcp                logrotate       sgml-base     ucf                   xml-core
dictionaries-common man-db          shells.state  unattended-upgrades
docker              misc            snapd         update-manager
root@Lenovo:/var/lib# cd docker
root@Lenovo:/var/lib/docker# cd volumes
root@Lenovo:/var/lib/docker/volumes# ls
backingFsBlockDev  metadata.db  volin  volout
root@Lenovo:/var/lib/docker/volumes# cd volin
root@Lenovo:/var/lib/docker/volumes/volin# ls
_data
root@Lenovo:/var/lib/docker/volumes/volin#
root@Lenovo:/var/lib/docker/volumes/volin# ls
_data
root@Lenovo:/var/lib/docker/volumes/volin# cd _data
root@Lenovo:/var/lib/docker/volumes/volin/_data# git clone https://github.com/alt-romes/programmer-calculator.git
Cloning into 'programmer-calculator'...
remote: Enumerating objects: 1326, done.
remote: Counting objects: 100% (211/211), done.
remote: Compressing objects: 100% (73/73), done.
remote: Total 1326 (delta 140), reused 182 (delta 134), pack-reused 1115
Receiving objects: 100% (1326/1326), 1.98 MiB | 1.10 MiB/s, done.
Resolving deltas: 100% (812/812), done.
root@Lenovo:/var/lib/docker/volumes/volin/_data# ls
programmer-calculator
root@Lenovo:/var/lib/docker/volumes/volin/_data#
```

- W kontenerze widoczne jest sklonowane repozytorium.

```
root@Lenovo:/home/kamila_partyka/Desktop/lab04# sudo docker run -it --name first --mount source=volin,target=/vin --mount source=volout,target=/vout ubuntu
root@be13d8eb9bfe:/# ls
bin  boot  dev  etc  home  lib  lib32  lib64  libx32  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var  vin  vout
root@be13d8eb9bfe:/# ls v*
var:
backups  cache  lib  local  lock  log  mail  opt  run  spool  tmp

vin:
programmer-calculator

vout:
root@be13d8eb9bfe:/# cd vin
root@be13d8eb9bfe:/vin# ls
programmer-calculator
root@be13d8eb9bfe:/vin# cd programmer-calculator
root@be13d8eb9bfe:/vin/programmer-calculator# ls
CONTRIBUTING.md  LICENSE  Makefile  README.md  assets  docs  how-to-publish.md  include  run-tests.sh  src  tests
root@be13d8eb9bfe:/vin/programmer-calculator#
```

- Do wykonania builda w kontenerze konieczne było doinstalowanie make, gcc oraz biblioteki ncurses.

```
root@be13d8eb9bfe:/vin# apt update
Get:1 http://archive.ubuntu.com/ubuntu jammy InRelease [270 kB]
Get:2 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:3 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [914 kB]
Get:4 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [908 kB]
Get:6 http://archive.ubuntu.com/ubuntu jammy-backports InRelease [108 kB]
Get:7 http://archive.ubuntu.com/ubuntu jammy/main amd64 Packages [1792 kB]
Get:8 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [907 kB]
Get:9 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [23.2 kB]
Get:10 http://archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [266 kB]
Get:11 http://archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [17.5 MB]
Get:12 http://archive.ubuntu.com/ubuntu jammy/restricted amd64 Packages [164 kB]
Get:13 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1148 kB]
Get:14 http://archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [28.6 kB]
Get:15 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1251 kB]
Get:16 http://archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [962 kB]
Get:17 http://archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [23.3 kB]
Get:18 http://archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [49.0 kB]
Fetched 26.5 MB in 18s (1475 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.
root@be13d8eb9bfe:/vin#
```

```
root@be13d8eb9bfe:/vin# apt-get install make gcc libncurses5-dev libncursesw5-dev -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu cpp cpp-11 fontconfig-config fonts-dejavu-core
  libbrotli1 libbsd0 libc-dev-bin libc-devtools libc6-dev libcc1-0 libcrypt-dev libctf-nobfd0 libct
  libgcc-11-dev libgd3 libgomp1 libisl23 libitm1 libjbig0 libjpeg-turbo8 libjpeg8 liblsan0 libmd0 l
  libquadmath0 libtiff5 libtirpc-dev libtsan0 libubsan1 libwebp7 libx11-6 libx11-data libxau6 libxc
  manpages-dev rpcsvc-proto ucf
Suggested packages:
```

- Zbudowano program poleceniem make.

```
root@be13d8eb9bfe:/vin/programmer-calculator# make
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -c src/draw.c -o build/draw.o
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -c src/history.c -o build/history.o
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -c src/main.c -o build/main.o
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -c src/numberstack.c -o build/numberstack.o
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -c src/operators.c -o build/operators.o
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -c src/parser.c -o build/parser.o
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -c src/xmalloc.c -o build/xmalloc.o
gcc -Wall -Wextra -g -Werror=missing-declarations -Werror=redundant-decls -Iinclude -o bin/pcalc build/draw.o build/history.o bu
.o build/operators.o build/parser.o build/xmalloc.o -lncurses #
Executing all complete!
root@be13d8eb9bfe:/vin/programmer-calculator#
```

- Zapisano pliki w woluminie wyjściowym.

```
root@be13d8eb9bfe:/vin/programmer-calculator/bin# ls
pcalc  programmer-calculator
root@be13d8eb9bfe:/vin/programmer-calculator/bin# cp pcalc /vin/programmer-calculator
root@be13d8eb9bfe:/vin/programmer-calculator/bin# ls
pcalc  programmer-calculator
root@be13d8eb9bfe:/vin/programmer-calculator/bin# cd ..
root@be13d8eb9bfe:/vin/programmer-calculator# ls
CONTRIBUTING.md  LICENSE  Makefile  README.md  assets  bin  build  docs  how-to-publish.md  include  pcalc  run-tests.sh  src
root@be13d8eb9bfe:/vin/programmer-calculator# cd ..
root@be13d8eb9bfe:/vin# ls
programmer-calculator
root@be13d8eb9bfe:/vin# cd ..
root@be13d8eb9bfe:/# ls
bin  boot  dev  etc  home  lib  lib32  lib64  libx32  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var  vin  vou
root@be13d8eb9bfe:/# cd vout
root@be13d8eb9bfe:/vout# ls
pcalc
```

EKSPORTOWANIE PORTU

- Pierwszym krokiem było pobranie i uruchomienie odpowiedniego obrazu dockerowego. Aby uzyskać obraz Dockerowy z oprogramowaniem iperf3, można skorzystać z oficjalnego obrazu iperf3 dostępnego na Docker Hub.

```
kamila_partyka@Lenovo:~$ sudo docker pull networkstatic/iperf3
[sudo] password for kamila_partyka:
Using default tag: latest
latest: Pulling from networkstatic/iperf3
bb263680fed1: Pull complete
f5ea6522a4b6: Pull complete
Digest: sha256:39fb418e92188f4f906da2a1582d4b82565ac72e98679d2914ab4546b19bc119
Status: Downloaded newer image for networkstatic/iperf3:latest
docker.io/networkstatic/iperf3:latest
kamila_partyka@Lenovo:~$ sudo docker run -it --rm --name=iperf3-server -p 5201:5201 networkstatic/iperf3 -s
-----------------------------------------------------------
Server listening on 5201
-----------------------------------------------------------
```

```
kamila_partyka@Lenovo:~$ sudo docker images
[sudo] password for kamila_partyka:
REPOSITORY              TAG       IMAGE ID       CREATED       SIZE
fedora                  latest    e4c5c8cc5d55   2 weeks ago   184MB
ubuntu                  latest    08d22c0ceb15   3 weeks ago   77.8MB
networkstatic/iperf3    latest    7e67b9e829c0   5 weeks ago   81.8MB
```

- Sprawdzono za pomocą sudo netstat nasłuchiwanie serwera.

```
kamila_partyka@Lenovo:~$ sudo netstat -utnpa | grep LISTEN
tcp6       0      0 :::5201                 :::*                    LISTEN      10558/iperf3
```

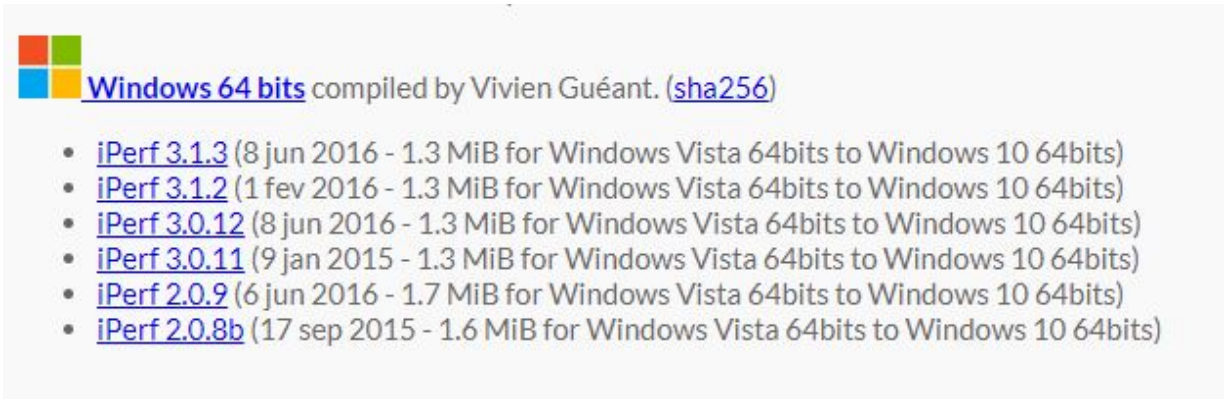
- Sprawdzono IP kontenera.

```
kamila_partyka@Lenovo:~$ docker inspect -f '{{range .NetworkSettings.Networks}}{{.IPAddress}}{{end}}' first
172.17.0.2
```

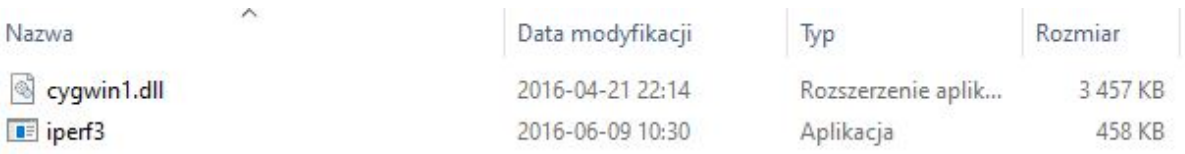
- W celu połączenia się spoza kontenera i spoza hosta zainstalowano iperf3.

```
kamila_partyka@Lenovo:~$ sudo apt install iperf3
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  bridge-utils dns-root-data dnsmasq-base ubuntu-fan
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  libiperf0 libsctp1
Suggested packages:
  lksctp-tools
The following NEW packages will be installed:
  iperf3 libiperf0 libsctp1
0 upgraded, 3 newly installed, 0 to remove and 64 not upgraded.
Need to get 106 kB of archives.
After this operation, 346 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://archive.ubuntu.com/ubuntu jammy/main amd64 libsctp1 amd64 1.0.19+dfsg-1build1 [9370 B]
Get:2 http://archive.ubuntu.com/ubuntu jammy/universe amd64 libiperf0 amd64 3.9-1build1 [81.5 kB]
Get:3 http://archive.ubuntu.com/ubuntu jammy/universe amd64 iperf3 amd64 3.9-1build1 [14.6 kB]
```

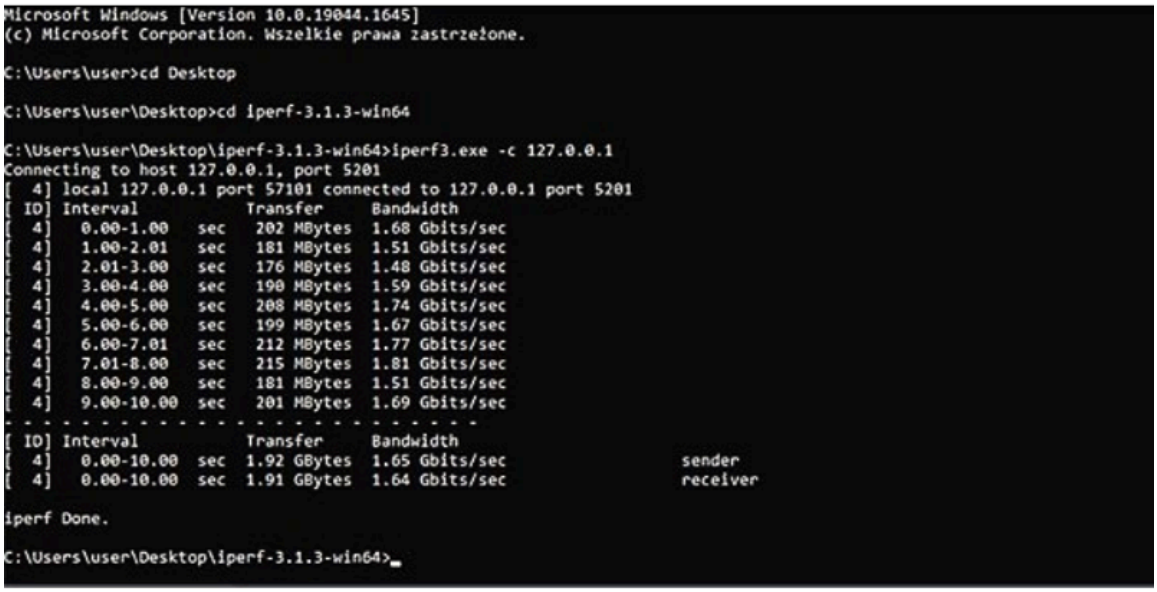
- Połączenie spoza hosta. Pobranie Iperf3 dla komputera z systemem Windows 10.

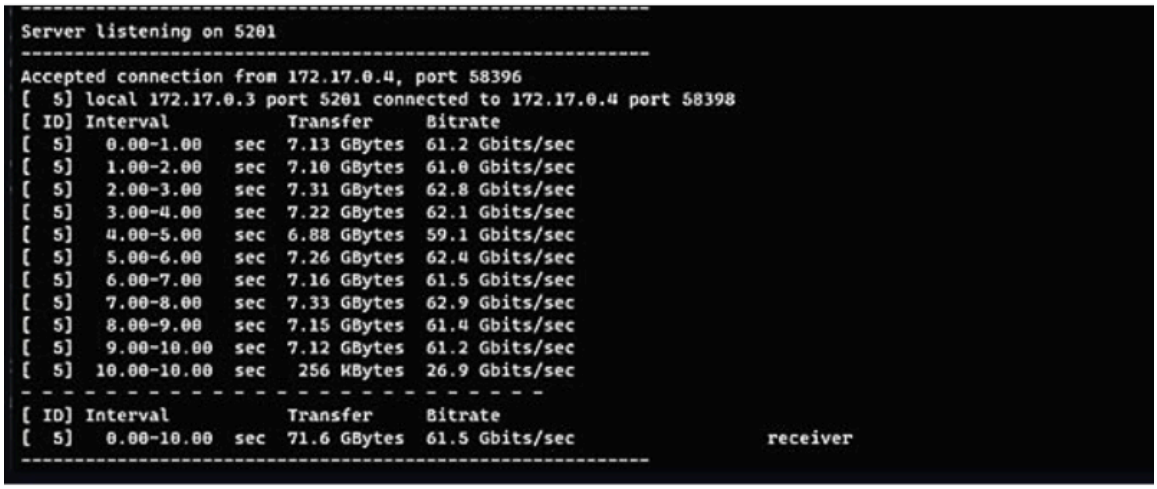


Po wypakowaniu:



- Wyciągnięto logi z kontenera aby przedstawić przepustowość komunikacji.





INSTANCJA JENKINS

- Utworzono nową sieć dockerową oraz pobrano i uruchomiono obraz DIND kontenera. Można również dodać dodatkowe opcje do polecenia docker network create aby skonfigurować sieć, na przykład określić, czy kontenery mają otrzymywać adresy IP automatycznie.

```
kamila_partyka@Lenovo:~$ sudo docker network create jenkins
[sudo] password for kamila_partyka:
4f46ffd559107f1d9152b99200d8ad5a7b38ddcae8acc4d520089f387958b0ed
```

- Utworzono Dockerfile.

```
kamila_partyka@Lenovo:~$ sudo docker run --name jenkins-docker --rm --detach --privileged --network jenkins
--network-alias docker --env DOCKER_TLS_CERTDIR=/certs --volume jenkins-docker-certs:/certs/client --volume
jenkins-data:/var/jenkins_home --publish  2376:2376 docker:dind \
> --storage-driver overlay2
Unable to find image 'docker:dind' locally
dind: Pulling from library/docker
f56be85fc22e: Pull complete
7ed9ddfd3b8f: Pull complete
788e1ab5616a: Pull complete
634492f0d979: Pull complete
0374339fff0d: Pull complete
140b66e8d860: Pull complete
81e994128f0f: Pull complete
56a555f71dff: Pull complete
11783ff70e21: Pull complete
7c7ba1df9096: Pull complete
a792e2c6a551: Pull complete
751a02a9c959: Pull complete
56f5a341abd5: Pull complete
14705636e8a9: Pull complete
Digest: sha256:29f11c7c63c84a51c45901701c2d16336c9abc6365d3a69136884b3cf15e2cc9
Status: Downloaded newer image for docker:dind
82864090ba07992f17ff7f1b430a4da0bd44d0b8f4cd42cbec205f285090f455
```

- Zbudowano nowy obraz dockera z pliku Dockerfile. Opcja -t pozwala na nadanie nazwy i opcjonalnie tagu dla nowo utworzonego obrazu. W przypadku gdy tag nie jest podany, domyślnie przypisywany jest tag latest.

```
kamila_partyka@Lenovo: $ docker build -t myjenkins-blueocean:2.387.1-1 .
[+] Building 258.2s (10/10) FINISHED
 => [internal] load build definition from Dockerfile                                                0.3s
 => => transferring dockerfile: 597B                                                                0.2s
 => [internal] load .dockerignore                                                                   0.3s
 => => transferring context: 2B                                                                     0.0s
 => [internal] load metadata for docker.io/jenkins/jenkins:2.387.1                                  3.5s
 => [1/6] FROM docker.io/jenkins/jenkins:2.387.1@sha256:0944e18261a6547e89b700cec432949281a7419a616  130.0s
 => => resolve docker.io/jenkins/jenkins:2.387.1@sha256:0944e18261a6547e89b700cec432949281a7419a6165a  0.1s
 => => sha256:d5ed2ceef0ec08e9044ebb39812f211d64dbcdfce775cc6b0460ca289193416f 13.13kB / 13.13kB     0.0s
 => => sha256:0944e18261a6547e89b700cec432949281a7419a6165a3906e78c97efde3bc86 2.36kB / 2.36kB       0.0s
 => => sha256:32fb02163b6bb519a30f909008e85235ddae10bdfd6b3d190dbdfe8f15403ea0 55.05MB / 55.05MB   66.8s
```

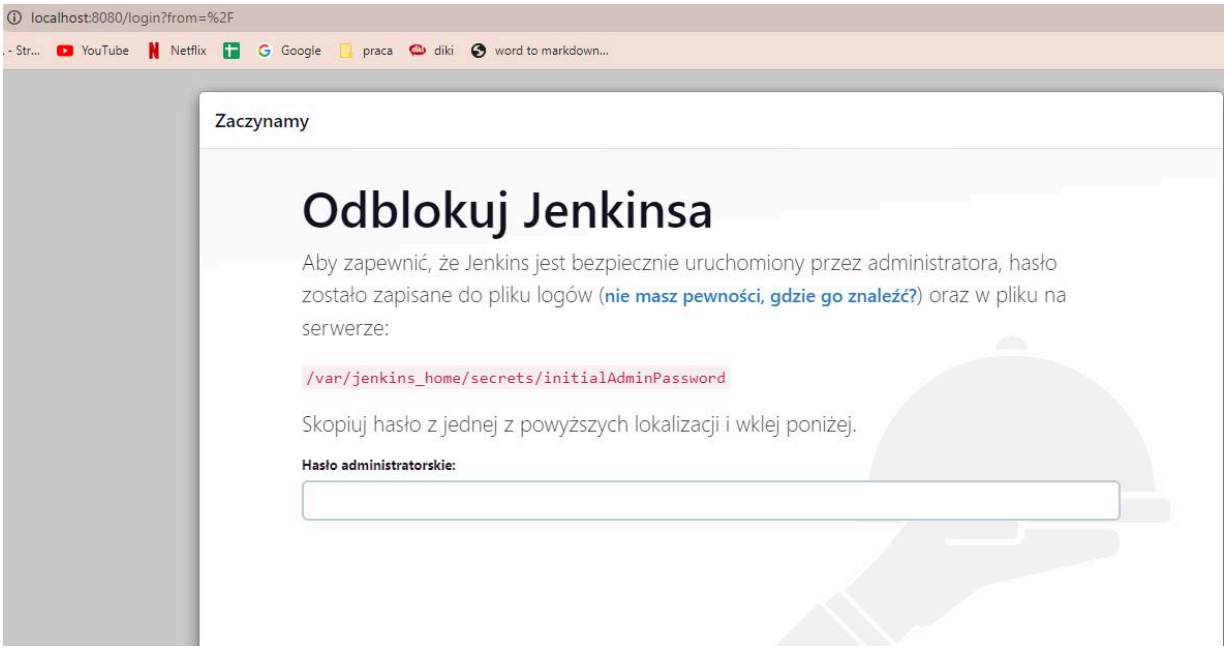
Efekt:

```
 => => naming to docker.io/library/myjenkins-blueocean:2.387.1-1
kamila_partyka@Lenovo:~$ sudo docker images
[sudo] password for kamila_partyka:
REPOSITORY            TAG         IMAGE ID       CREATED         SIZE
myjenkins-blueocean   2.387.1-1   76eee9a33d6b   4 minutes ago   811MB
docker                dind        1d27856a797d   3 days ago      318MB
img2                  latest      6d663f21191a   12 days ago     368MB
img1                  latest      6dcfa9b1e60d   12 days ago     368MB
ubuntu                latest      08d22c0ceb15   3 weeks ago     77.8MB
hello-world           latest      feb5d9fea6a5   18 months ago   13.3kB
```

- Uruchomiono stworzony obraz.

```
kamila_partyka@Lenovo:~$ sudo docker images
[sudo] password for kamila_partyka:
REPOSITORY            TAG         IMAGE ID       CREATED         SIZE
myjenkins-blueocean   2.387.1-1   76eee9a33d6b   4 minutes ago   811MB
docker                dind        1d27856a797d   3 days ago      318MB
img2                  latest      6d663f21191a   12 days ago     368MB
img1                  latest      6dcfa9b1e60d   12 days ago     368MB
ubuntu                latest      08d22c0ceb15   3 weeks ago     77.8MB
hello-world           latest      feb5d9fea6a5   18 months ago   13.3kB
kamila_partyka@Lenovo:~$ sudo docker run \
> --name jenkins-blueocean \
> --restart=on-failure \
> --detach \
> --network jenkins \
> --env DOCKER_HOST=tcp://docker:2376 \
> --env DOCKER_CERT_PATH=/certs/client \
> --env DOCKER_TLS_VERIFY=1 \
> --publish 8080:8080 \
> --publish 50000:50000 \
> --volume jenkins-data:/var/jenkins_home \
> --volume jenkins-docker-certs:/certs/client:ro \
> myjenkins-blueocean:2.387.1-1
ecc6def21b4b91afb1e0e7b22db05e6ae60dfb37188eeae580ae99a242d72eae
```

- Sprawdzono adres localhost:8080 i odblokowano za pomocą hasła. Localhost:8080 to adres URL, który wskazuje na aplikację lub serwer nasłuchujący na porcie 8080 na komputerze lokalnym. W kontekście Dockera, jeśli wewnątrz kontenera jest uruchomiona aplikacja, która nasłuchuje na porcie 8080, to można ją osiągnąć poprzez przekierowanie portu na hoście do portu w kontenerze. Czyli dokładniej, uruchamiając kontener z aplikacją, która nasłuchuje na porcie 8080, można przekierować port 8080 z kontenera na port 8080 na hoście (czyli komputerze lokalnym) za pomocą opcji -p w poleceniu docker run.



- W celu odczytania hasła uruchomiono interaktywny terminal za pomocą sudo docker exec. Polecenie sudo docker exec służy do uruchamiania polecenia w kontenerze, który już działa. Umożliwia to interaktywną pracę z kontenerem bez potrzeby uruchamiania go ponownie. Polecenie cat initialAdminPassword pozwoliło na hasła. W kontekście Dockera, plik

initialAdminPassword zwykle znajduje się w katalogu var/jenkins_home/secrets wewnątrz kontenera z Jenkins, a jego zawartość zawiera hasło administratora, które jest wymagane podczas pierwszego logowania do Jenkinsa.

```
kamila_partyka@Lenovo: $ sudo docker exec -it jenkins-blueocean bash
jenkins@ecc6def21b4b:/$ ls
bin   certs  etc   lib    media  opt   root  sbin  sys  usr
boot  dev    home  lib64  mnt    proc  run   srv   tmp  var
jenkins@ecc6def21b4b:/$ cd var
jenkins@ecc6def21b4b:/var$ cd jenkins_home
jenkins@ecc6def21b4b:~$ ls
config.xml                     jenkins.telemetry.Correlator.xml  plugins                 userContent
copy_reference_file.log        jobs                              secret.key              users
hudson.model.UpdateCenter.xml  logs                              secret.key.not-so-secret  war
hudson.plugins.git.GitTool.xml  nodeMonitors.xml                 secrets
identity.key.enc               nodes                             updates
jenkins@ecc6def21b4b:~$ cd secrets
jenkins@ecc6def21b4b:~/secrets$ cat initialAdminPassword
72cc72f45d5a4a09a6d11cb7171c23ce
jenkins@ecc6def21b4b:~/secrets$
```

- Podczas instalacji Jenkinsa zainstalowałam sugerowane wtyczki, które są zalecane do poprawnego działania niektórych funkcjonalności Jenkinsa.

- Użyto polecenia sudo docker ps , jest to polecenie, które wyświetla listę działających kontenerów. Domyślnie polecenie to wyświetla tylko informacje o kontenerach, które zostały uruchomione i są w trakcie działania. Działanie polecenia sudo docker ps polega na wyświetleniu następujących informacji dla każdego działającego kontenera:

CONTAINER ID : unikalny identyfikator kontenera

IMAGE : nazwa obrazu, na podstawie którego został utworzony kontener

COMMAND : polecenie, które zostało uruchomione wewnątrz kontenera

CREATED : czas, kiedy kontener został utworzony

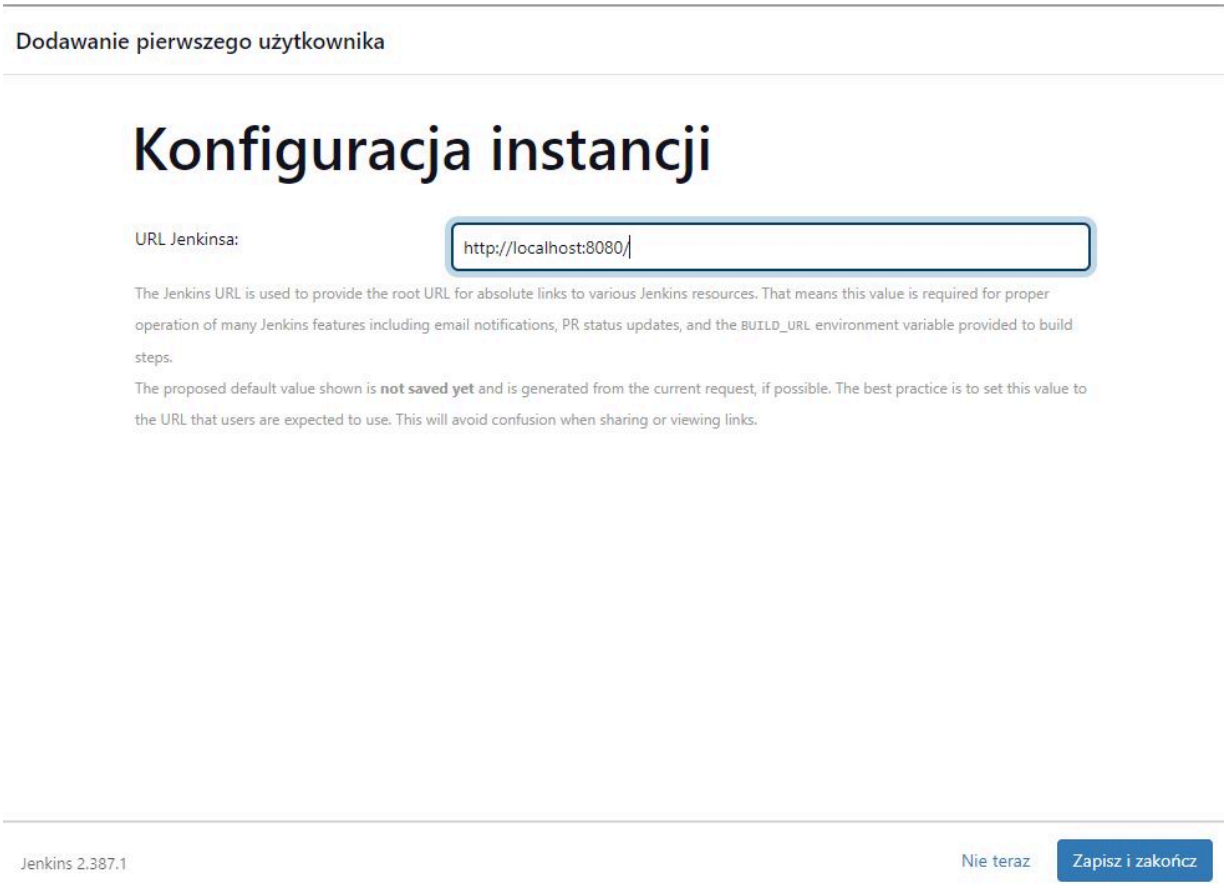
STATUS : aktualny status kontenera

PORTS : lista przekierowanych portów

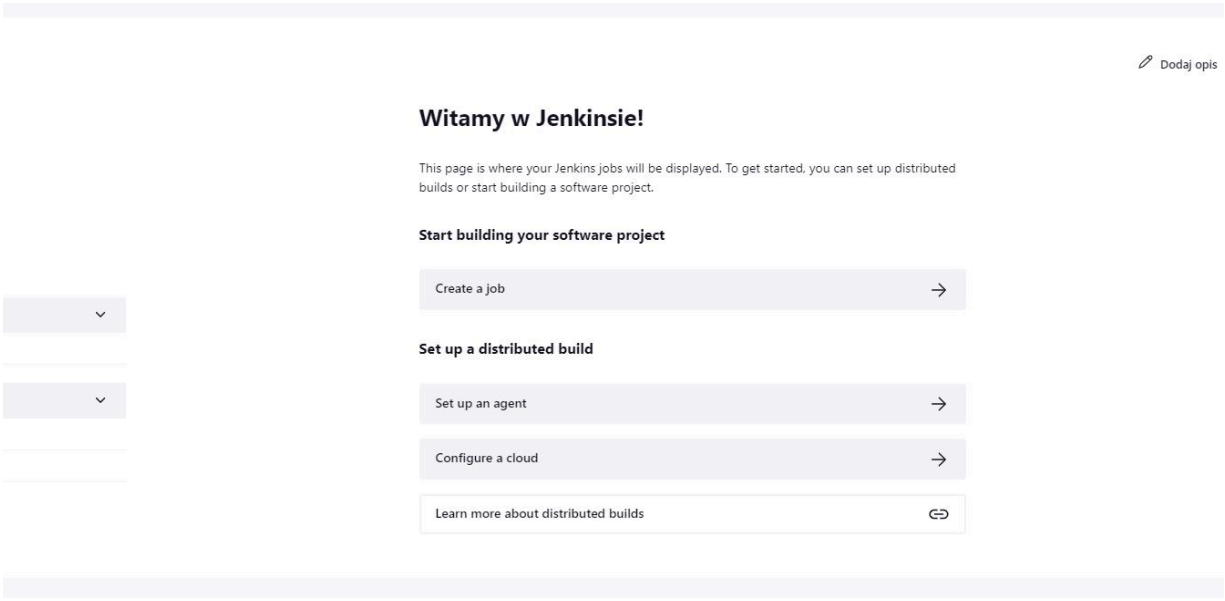
NAMES : nadana przez użytkownika nazwa kontenera (lub automatycznie nadana nazwa)

```
kamila_partyka@Lenovo:~$ sudo docker ps
[sudo] password for kamila_partyka:
CONTAINER ID   IMAGE                              COMMAND                  CREATED          STATUS          PORTS
NAMES
ecc6def21b4b   myjenkins-blueocean:2.387.1-1      "/usr/bin/tini -- /u…"   10 minutes ago   Up 10 minutes   0.0.0.0:8080->8080/tcp, 0.0.0.0:50000->50000/tcp
jenkins-blueocean
82864090ba07   docker:dind                        "dockerd-entrypoint.…"   32 minutes ago   Up 32 minutes   2375/tcp, 0.0.0.0:2376->2376/tcp
jenkins-docker
```

- Dodanie pierwszego użytkownika.



Teraz można korzystać z Jenkinsa jako nowy użytkownik.


Dodaj opis
Witamy w Jenkinsie!
This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.
Start building your software project
Create a job
Set up a distributed build
Set up an agent
Configure a cloud
Learn more about distributed builds