

Sprawozdanie z pliku: Zadanie1

Wykonał: Viacheslav Peleshok, grupa 6.8

1) Program serwera (Javascript)

```
const http = require('http');
const os = require('os');
const url = require('url');
const moment = require('moment-timezone'); // Wymagana instalacja moment-timezone

// Dane autora serwera
const authorName = 'Viacheslav Peleshok'; // Imię Nazwisko

// Pobierz aktualną datę i godzinę uruchomienia serwera
const serverStartTime = new Date();

// Utwórz serwer HTTP
const server = http.createServer((req, res) => {
  // Pobierz adres IP klienta
  const clientIP = req.connection.remoteAddress;

  // Loguj informacje o uruchomieniu serwera
  console.log(`Serwer uruchomiony przez: ${authorName}`);
  console.log(`Data uruchomienia: ${serverStartTime}`);
  console.log(`Serwer nasłuchuje na porcie: ${server.address().port}`);

  // Pobierz datę i czas w strefie czasowej klienta
  const clientDateTime = moment().tz('Europe/Warsaw').format('YYYY-MM-DD HH:mm:ss');

  // Przygotuj treść odpowiedzi dla klienta
  const responseContent = `
    <html>
    <head><title>Informacje o kliencie</title></head>
    <body>
      <h1>Adres IP klienta: ${clientIP}</h1>
      <h2>Data i godzina w strefie czasowej klienta: ${clientDateTime}</h2>
      <h3> Autor: ${authorName}</h3>
    </body>
    </html>
  `;

  // Ustaw nagłówki odpowiedzi
  res.writeHead(200, {'Content-Type': 'text/html'});

  // Wyślij odpowiedź do klienta
  res.end(responseContent);
});

// Określ port, na którym serwer będzie nasłuchiwał
const port = process.env.PORT || 3000;

// Uruchom serwer i nasłuchuj na określonym porcie
server.listen(port, () => {
  console.log(`Serwer nasłuchuje na porcie ${port}`);
});
```

2) Wymagany Dockerfile do zbudowania obrazu

```
# Etap 1: Budowanie aplikacji
FROM node:14 as builder

# Ustawienie katalogu roboczego
WORKDIR /app

# Skopiowanie plików aplikacji
COPY package.json package-lock.json ./

# Instalacja zależności
RUN npm install

# Utworzenie katalogu build (jeśli nie istnieje)
RUN mkdir -p /app/build

# Skopiowanie kodu aplikacji
COPY . .

# Budowanie aplikacji
RUN npm run build

# Etap 2: Uruchomienie aplikacji
FROM node:14-alpine

# Ustawienie katalogu roboczego
WORKDIR /app

# Skopiowanie plików z etapu budowania
COPY --from=builder /app/build ./build
COPY --from=builder /app/server.js ./server.js

# Skopiowanie plików package.json i package-lock.json
COPY --from=builder /app/package.json ./package.json
COPY --from=builder /app/package-lock.json ./package-lock.json

# Instalacja tylko produkcji zależności
RUN npm install --only=production

# Ustawienie metadanych autora
LABEL author="Viacheslav Peleshok"

# Wystawienie portu
EXPOSE 5000

# Uruchomienie aplikacji
CMD ["node", "server.js"] # Uruchamiamy tutaj server.js
```

3) Polecenia do:

a) zbudowania opracowanego obrazu kontenera:

docker build -t viacheslav1:v1 .

b) uruchomienia kontenera na podstawie zbudowanego obrazu:

docker run -dt -p 3000:3000 viacheslav1:v1

c) Sposobu uzyskania informacji, co generuje serwer przy odpalniu:

docker logs 8fc //8fc to 3 pierwsze znaki nazwy kontenera(info z docker ps, na screenie widać)

```
Air-DATA:zadanie1 data228$ docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS
[
  NAMES
8fce09f6fe29   viacheslav1:v1 "docker-entrypoint.s..." 3 seconds ago  Up 2 seconds  5000/tcp, 0.0.0.0
:3001->3000/tcp   quirky_morse
6c68034f66b6   viacheslav1:v1 "docker-entrypoint.s..." 34 seconds ago Up 32 seconds  0.0.0.0:3000->300
0/tcp, 5000/tcp   pedantic_goldberg
Air-DATA:zadanie1 data228$ docker logs 8fc
Serwer nasłuchuje na porcie 3000
```

a tak wygląda działający serwer w przeglądarce:

```
SPR1_Zadanie1 - Google Документы

Adres IP klienta: ::1

Data i godzina w strefie czasowej klienta: 2024-05-08 17:45:04

Autor: Viacheslav Peleshok
```

4) A tym poleceniami dowiemy się ile warstw posiada obraz:

docker history viacheslav1:v1

```
Air-DATA:zadanie1 data228$ docker history viacheslav1:v1
IMAGE          CREATED          CREATED BY          SIZE      COMMENT
f2bcc9304c1e   13 minutes ago   CMD ["node" "server.js"] 0B         buildkit.dockerfi
le.v0
<missing>      13 minutes ago   EXPOSE map[5000/tcp:{}] 0B         buildkit.dockerfi
le.v0
<missing>      13 minutes ago   LABEL author=Viacheslav Peleshok 0B         buildkit.dockerfi
le.v0
<missing>      13 minutes ago   RUN /bin/sh -c npm install --only=production... 8.32MB     buildkit.dockerfi
le.v0
<missing>      14 minutes ago   COPY /app/package-lock.json ./package-lock.j... 639B       buildkit.dockerfi
le.v0
<missing>      14 minutes ago   COPY /app/package.json ./package.json # buil... 139B       buildkit.dockerfi
le.v0
<missing>      14 minutes ago   COPY /app/server.js ./server.js # buildkit 1.67kB     buildkit.dockerfi
le.v0
<missing>      19 minutes ago   COPY /app/build ./build # buildkit 0B         buildkit.dockerfi
le.v0
<missing>      26 minutes ago   WORKDIR /app 0B         buildkit.dockerfi
le.v0
```