Отчёт. Часть 1.

Для начала был написан сайт на HTML:

<!DOCTYPE html>

<html lang="ru-RU">

<head>

<meta charset="utf-8" />

<title>My SIT WebServer Page</title>

</head>

<body>

<h1>КОТИКИ - МИЛЕНЬКО</h1>

<img src="images/kitty.jpg" alt="Тут должен быть котик :3" />

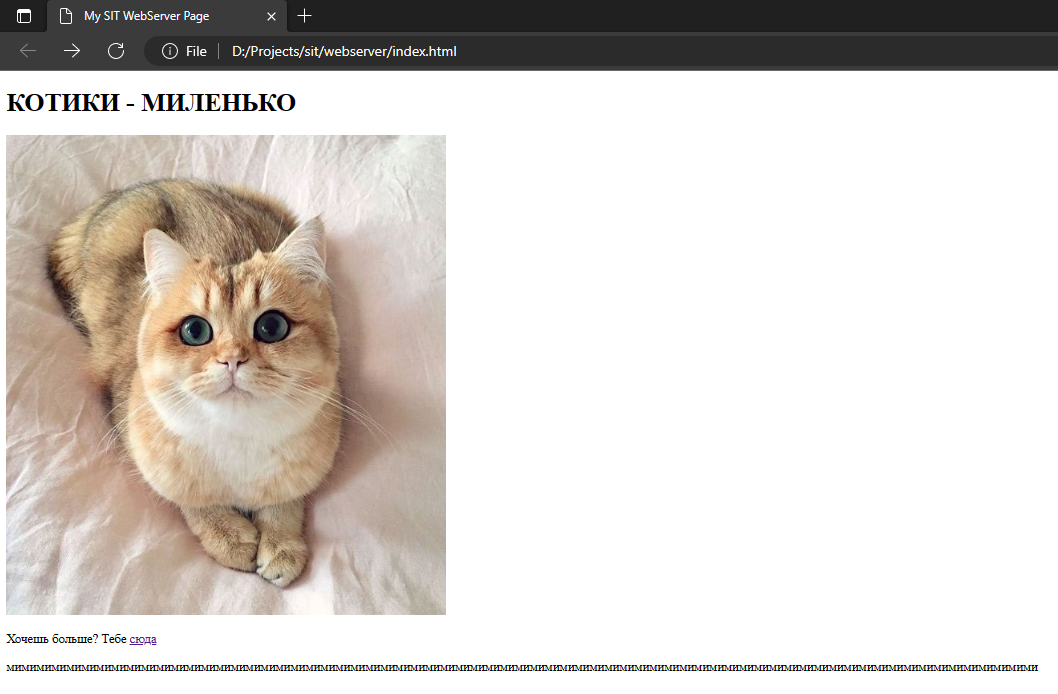
<p class="par1">Хочешь больше? Тебе <a href="https://yandex.ru/images/search?from=tabbar&text=%D0%BE%D1%87%D0%B5%D0%BD%D1%8C%20%D0%BC%D0%B8%D0%BB%D1%8B%D0%B5%20%D0%BA%D0%BE%D1%82%D0%B8%D0%BA%D0%B8">сюда</a></p>

<p>мимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимими</p>

</body>

</html>

Затем была проверена его работоспособность. Для этого файл index.html был открыт в браузере Edge:



Как мы видим, сайт работает успешно, весь контент отображается, как и должен. После был добавлен CSS файл:

<!DOCTYPE html>

<html lang="ru-RU">

<head>

<meta charset="utf-8" />

<title>My SIT WebServer Page</title>

<link href="styles/style.css" rel="stylesheet" />

</head>

<body>

<h1>КОТИКИ - МИЛЕНЬКО</h1>

<img src="images/kitty.jpg" alt="Тут должен быть котик :3" />

<p class="par1">Хочешь больше? Тебе <a href="https://yandex.ru/images/search?from=tabbar&text=%D0%BE%D1%87%D0%B5%D0%BD%D1%8C%20%D0%BC%D0%B8%D0%BB%D1%8B%D0%B5%20%D0%BA%D0%BE%D1%82%D0%B8%D0%BA%D0%B8">сюда</a></p>

<p>мимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимимими</p>

</body>

</html>

html {

font-size: 20px;

font-family: "Comic Sans", "Comic Sans MS", cursive;

}

h1 {

background-color: #deb3e6;

text-align: center;

font-size: 40px;

}

img {

display: block;

margin: 0 auto;

}

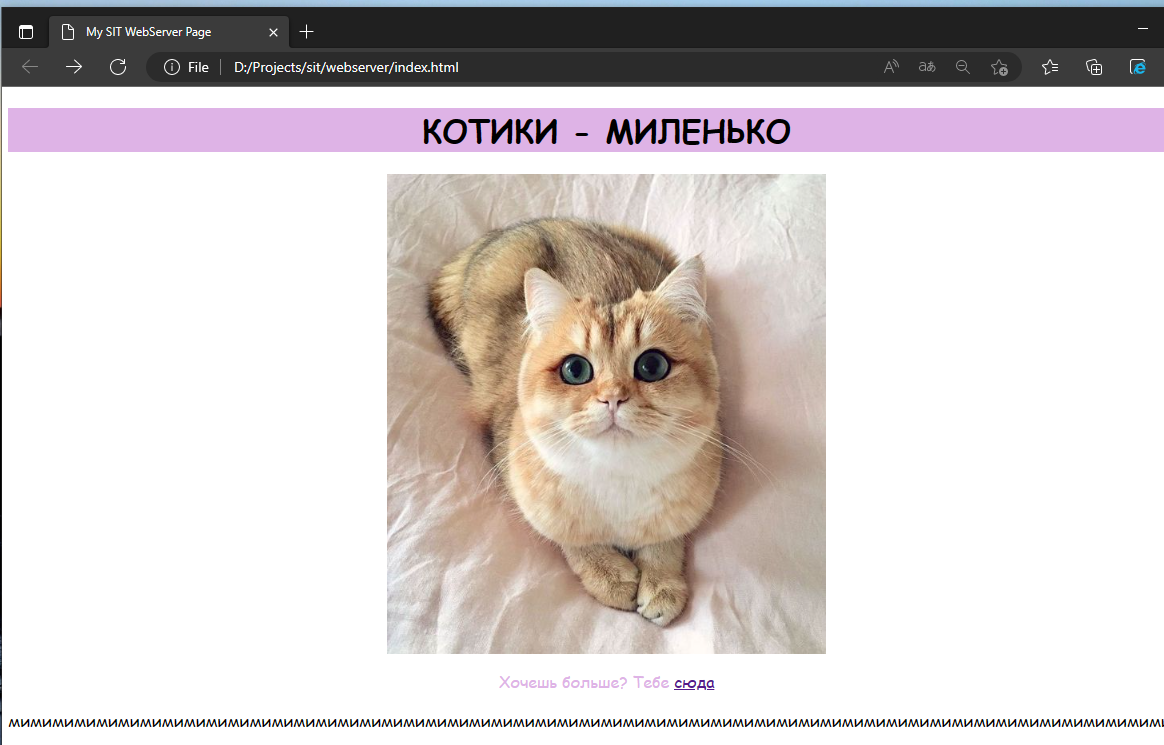
p.par1 {

text-align: center;

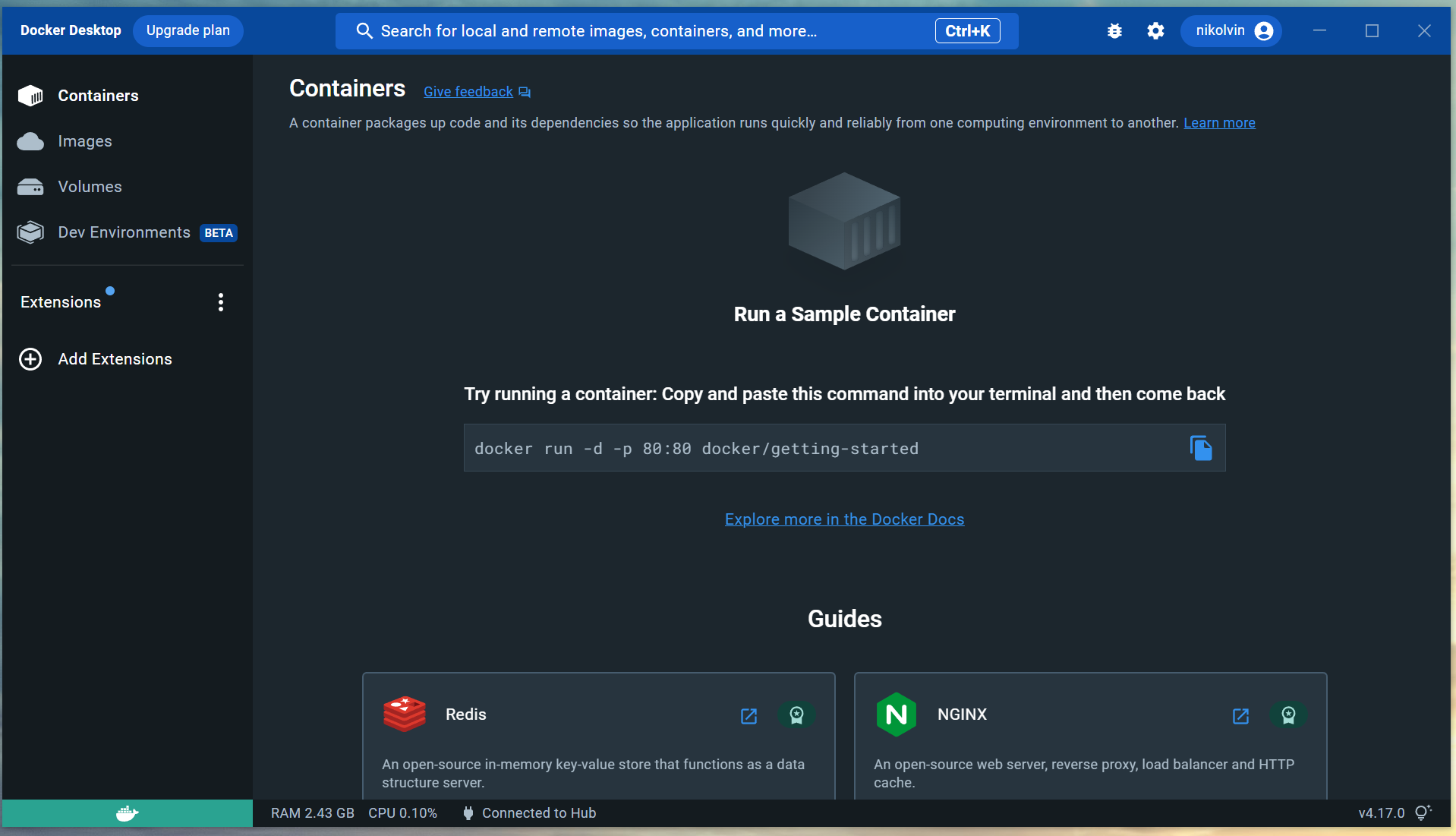
color: #deb3e6;

}

После этого снова проверяем работоспособность сайта в браузере Edge:



Как видим, базовый сайт работает. Можно переходить к подготовке контейнера. Включим компонент WSL, обновим его и установим Docker. После успешной установки отображается следующее меню:



Подготовим Dockerfile:

FROM nginx:latest

COPY ./webserver/index.html /usr/share/nginx/html/index.html

COPY ./webserver/styles/style.css /usr/share/nginx/html/styles/style.css

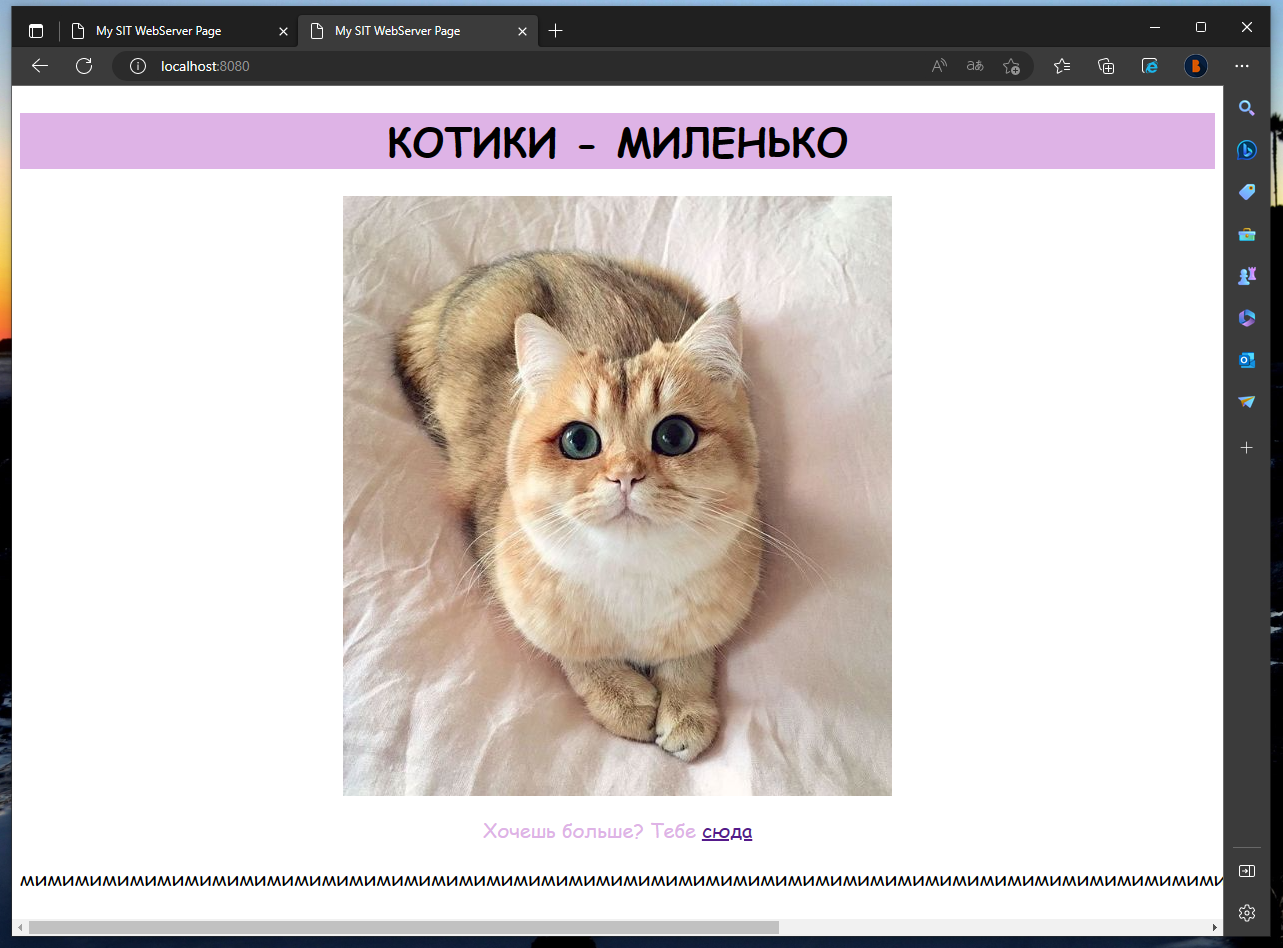
COPY ./webserver/images/kitty.jpg /usr/share/nginx/html/images/kitty.jpg

И запустим его следующим скриптом:

docker build -t webserver .

docker run -it --rm -d -p 8080:80 --name web webserver

Теперь попробуем подключится по адресу `localhost:8080`:



Как мы видим, конейнер успешно работает.