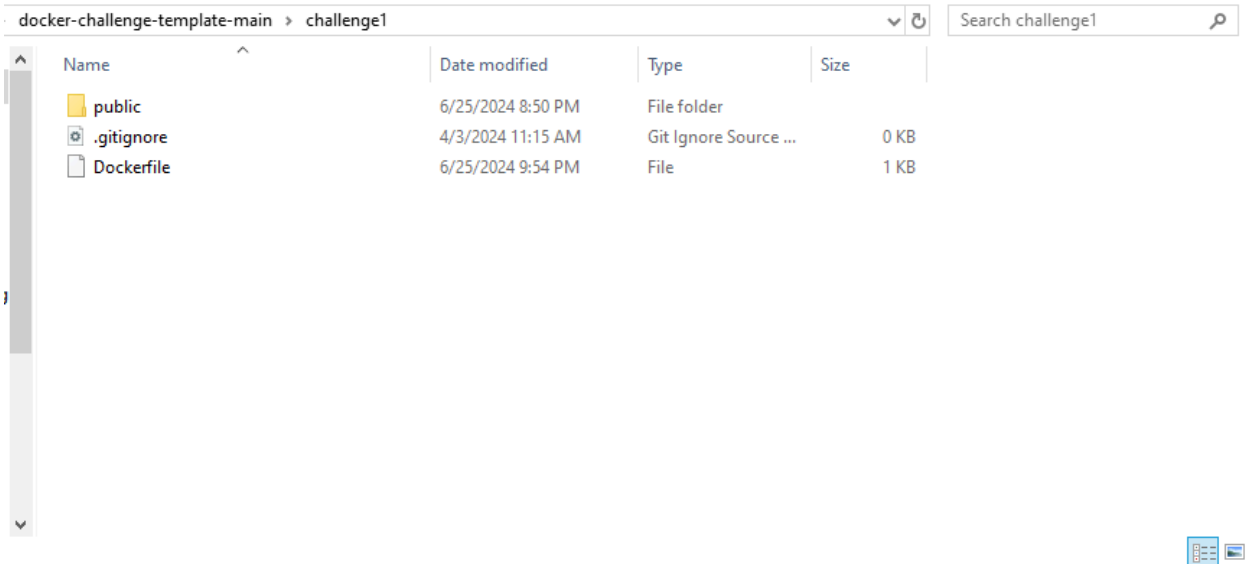
Docker Challenge 1

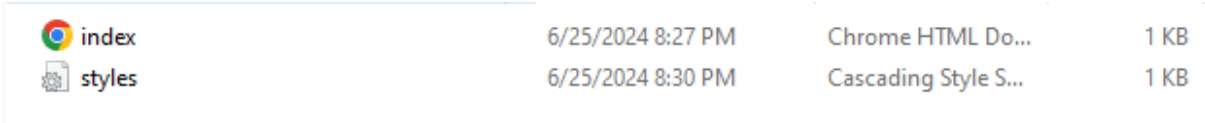
Danny Horning
000702974
June 25 2024

Creating the File folder:



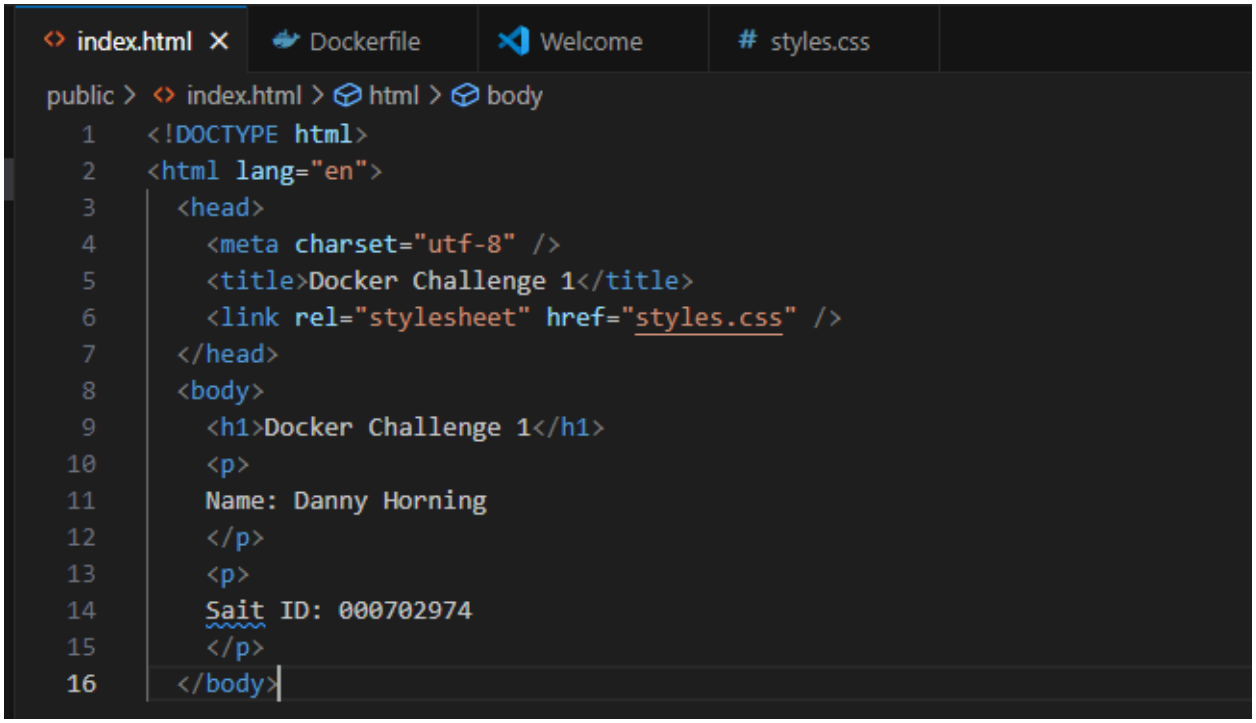
The creation of the folder once I had downloaded the template from the github repository.This was the basic folder structure that I imported into VS Code.

The public folder contents:



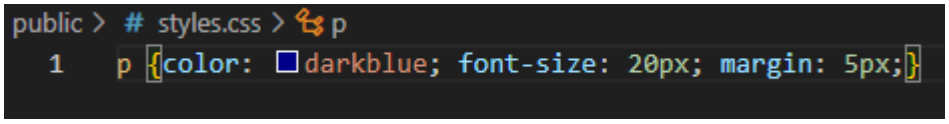
In the public folder I placed my HTML file, as well as a styles.css, purely because I wanted to experiment with having multiple files in the folder.

The index file:



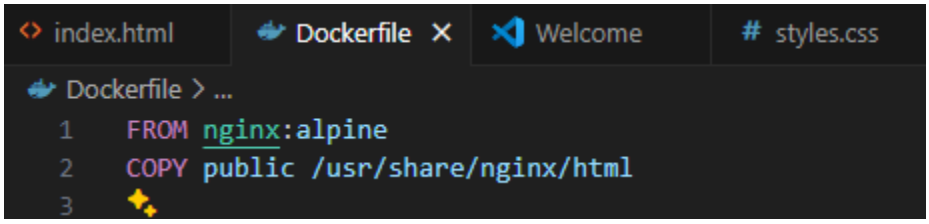
The index file was extremely basic. The header had the challenge name, and the <p> tags had my name and sait id.

The styles file:



The styles simply just changed the colour of my name and id to blue, changed the size, and the spacing between them

The docker file:



The docker file simply used an NGinx (alpine was used to make the file smaller) and copied the contents of the public folder into the NGinx html.

Docker build:

```
PS C:\users\danny\desktop\docker-challenge-template-main\challenge1> docker build -t docker-challenge-1 .
[+] Building 6.6s (8/8) FINISHED
 => [internal] load build definition from Dockerfile
 => => transferring dockerfile: 93B
 => [internal] load metadata for docker.io/library/nginx:alpine
 => [auth] library/nginx:pull token for registry-1.docker.io
 => [internal] load .dockerignore
 => => transferring context: 2B
 => [internal] load build context
 => => transferring context: 101B
 => CACHED [1/2] FROM docker.io/library/nginx:alpine@sha256:a45ee5d042aaa9e81e013f97ae40c3dda26fbe98f22b6251acdf28e579560d55
 => [2/2] COPY public /usr/share/nginx/html
 => exporting to image
 => => exporting layers
 => => writing image sha256:4d53878e1b158136bbbc07687e3473737ce4d4ae5b580bcd1cfd326029919dbb
 => => naming to docker.io/library/docker-challenge-1

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/hziw5psshhnmkfqxg5yofx50x
```

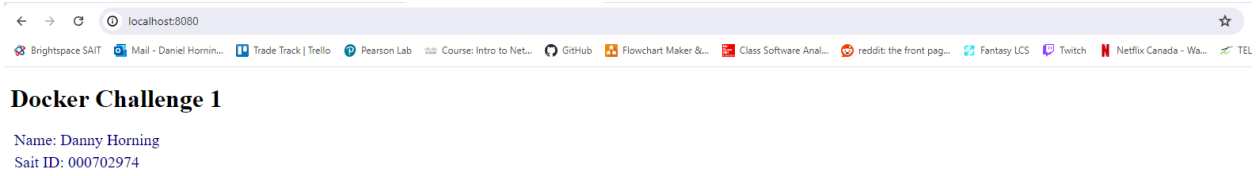
Now it was time to build the container. I used the command docker build -t docker-challenge-1 to build the container. Once the container was built, we could run the container.

Docker run:

```
PS C:\users\danny\desktop\docker-challenge-template-main\challenge1> docker run -d -p 8080:80 docker-challenge-1
c9244551d15e67c33f0e5b01ea6e85df32b5537f313bbed5d01eb1ccb9bf4e3a
```

Here I ran the container using ‘docker run -d -p 8080:80 docker-challenge-1’. This command ran the container on the host machine port 8080.

Output:



Visiting the port 8080 localhost we can now see the HTML document as well as the simple styling changes

References

https://mav-rana.medium.com/containerize-a-simple-static-website-a381c5e49633

https://hub.docker.com/_/nginx

https://www.tutorialspoint.com/docker/docker_setting_nginx.htm

https://www.youtube.com/watch?v=pTFZFxd4hOI&ab_channel=ProgrammingwithMosh

https://docs.docker.com/reference/cli/docker/image/build/