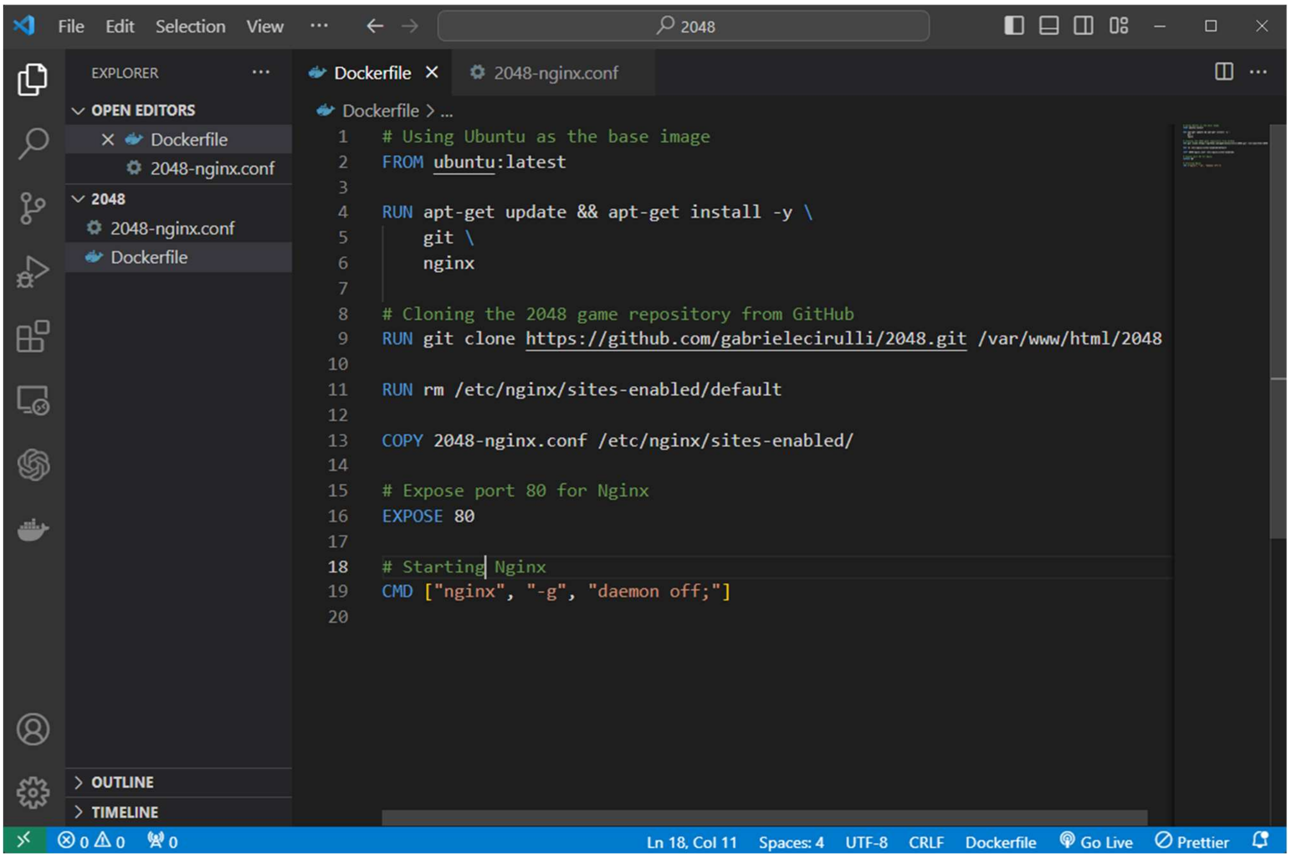
# Deploying 2048-Game on Nginx Server

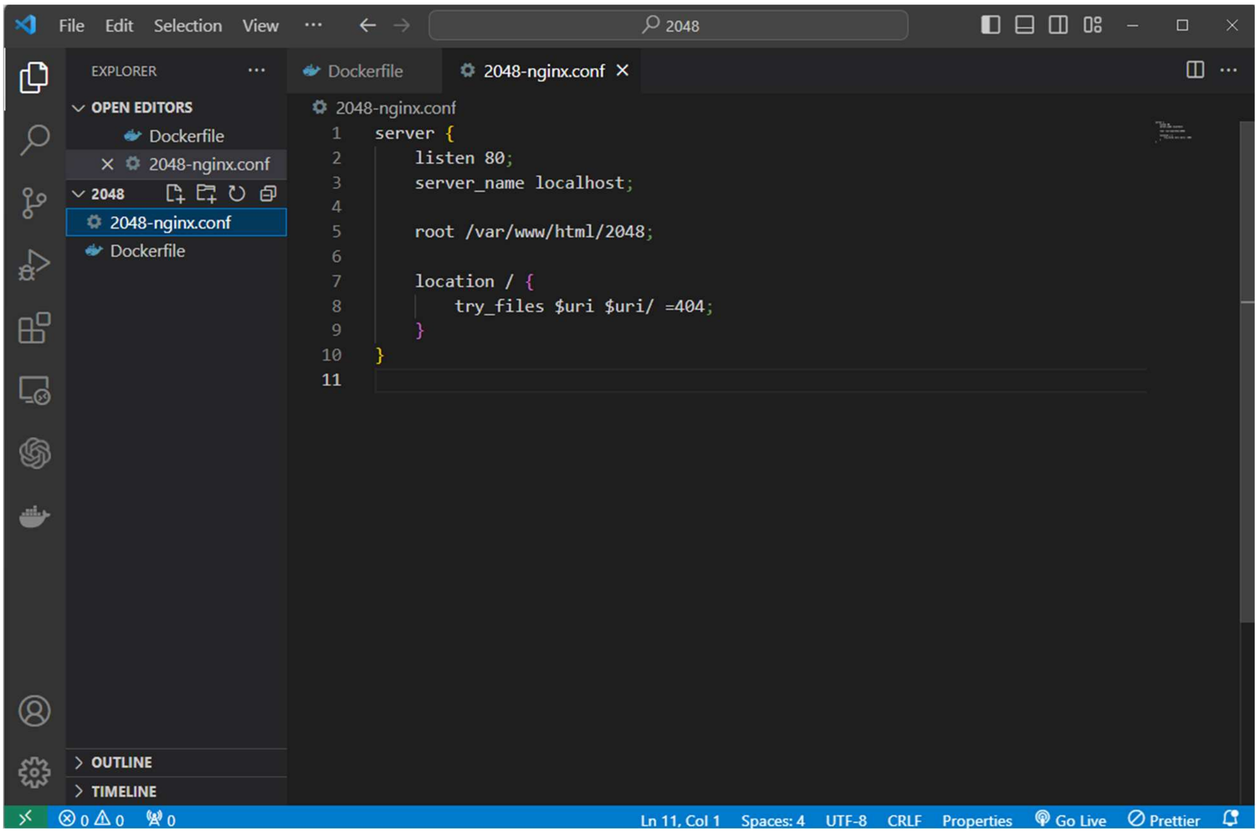
- Snippet for the Dockerfile that will be used to build the image for the project is attached below.

- Ubuntu is used as the base Image
- Nginx Server is installed
- Code for the 2048 game is cloned from the GitHub repository
- Configurations of Nginx are updated to allow it to listen
- Port 80 of the Nginx server is exposed.



```
# Using Ubuntu as the base image
FROM ubuntu:latest

RUN apt-get update && apt-get install -y \
    git \
    nginx

# Cloning the 2048 game repository from GitHub
RUN git clone https://github.com/gabrielecirulli/2048.git /var/www/html/2048

RUN rm /etc/nginx/sites-enabled/default

COPY 2048-nginx.conf /etc/nginx/sites-enabled/

# Expose port 80 for Nginx
EXPOSE 80

# Starting Nginx
CMD ["nginx", "-g", "daemon off;"]

```

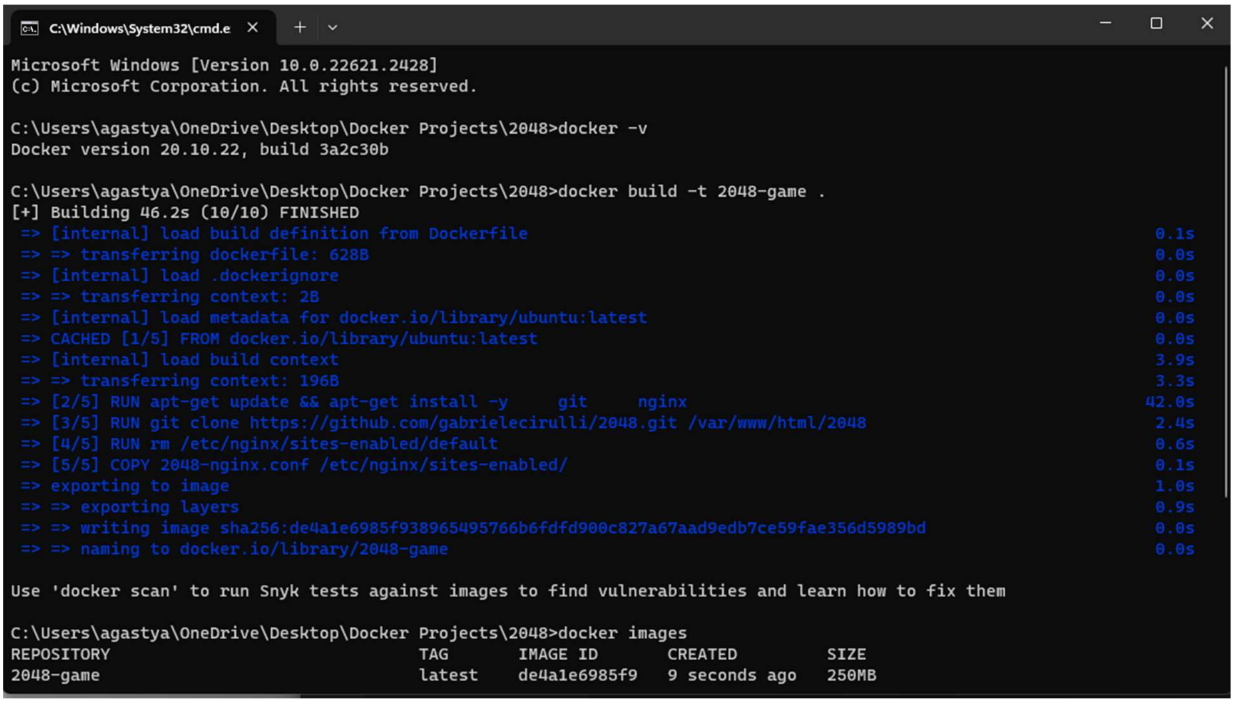
- Setting Up the configurations of Nginx server to listen on port 80:



2048-nginx.conf

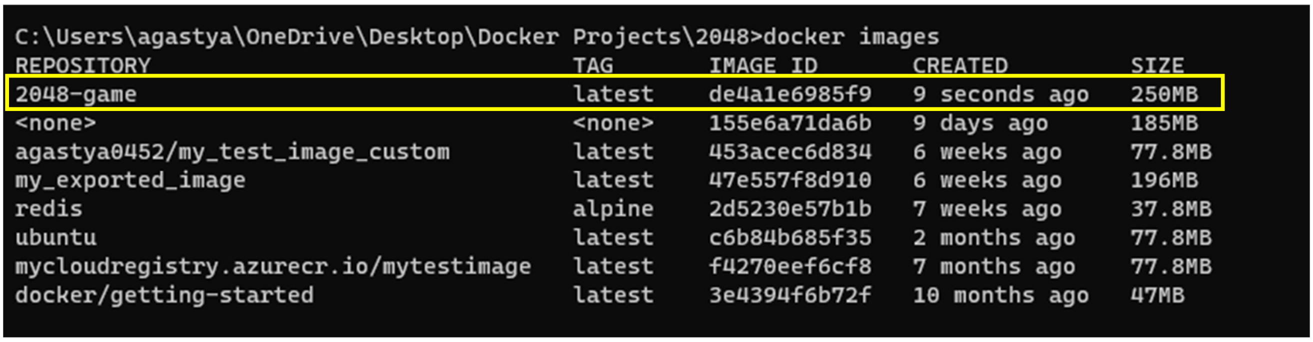
```
server {
    listen 80;
    server_name localhost;

    root /var/www/html/2048;

    location / {
        try_files $uri $uri/ =404;
    }
}

```

- Image is built using the docker command: '*docker build -t 2048-game .*':



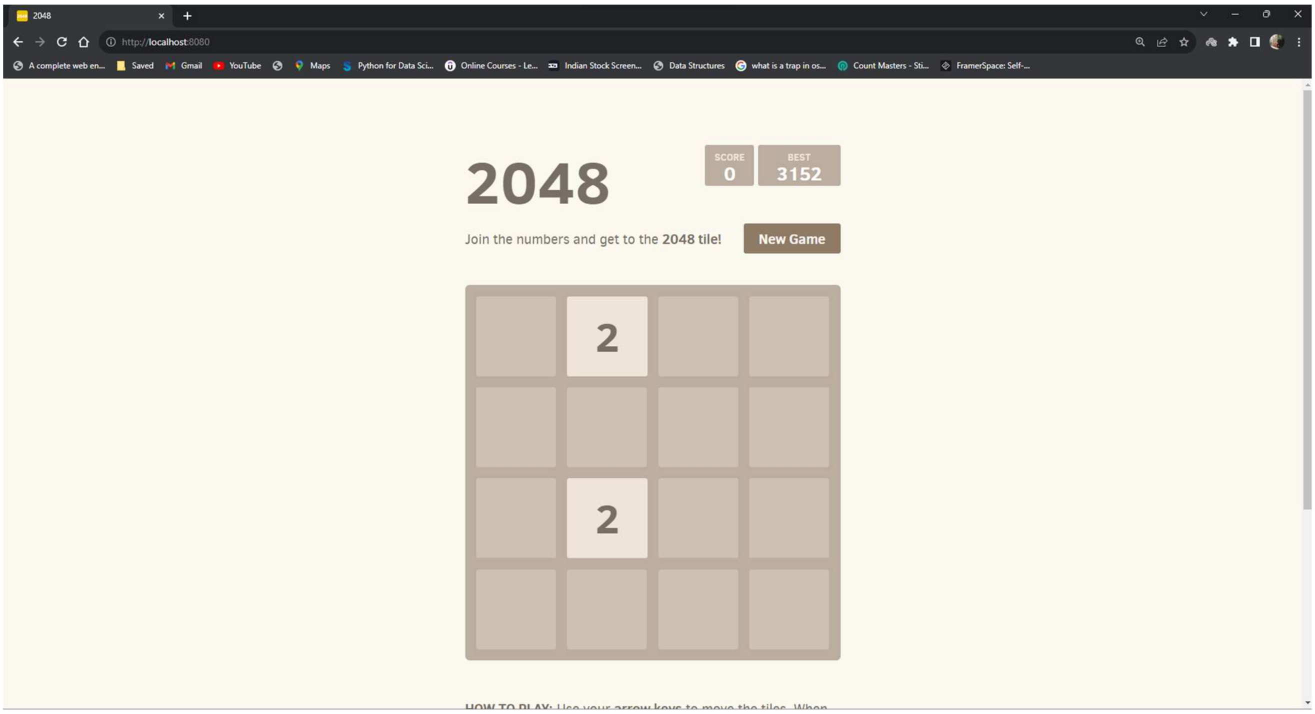
```
Microsoft Windows [Version 10.0.22621.2428]
(c) Microsoft Corporation. All rights reserved.

C:\Users\agastya\OneDrive\Desktop\Docker Projects\2048>docker -v
Docker version 20.10.22, build 3a2c30b

C:\Users\agastya\OneDrive\Desktop\Docker Projects\2048>docker build -t 2048-game .
[+] Building 46.2s (10/10) FINISHED
 => [internal] load build definition from Dockerfile                                                     0.1s
 => => transferring dockerfile: 628B                                                                     0.0s
 => [internal] load .dockerignore                                                                        0.0s
 => => transferring context: 2B                                                                          0.0s
 => [internal] load metadata for docker.io/library/ubuntu:latest                                         0.0s
 => CACHED [1/5] FROM docker.io/library/ubuntu:latest                                                    0.0s
 => [internal] load build context                                                                        3.9s
 => => transferring context: 196B                                                                        3.3s
 => [2/5] RUN apt-get update && apt-get install -y     git     nginx                                    42.0s
 => [3/5] RUN git clone https://github.com/gabrielecirulli/2048.git /var/www/html/2048                   2.4s
 => [4/5] RUN rm /etc/nginx/sites-enabled/default                                                        0.6s
 => [5/5] COPY 2048-nginx.conf /etc/nginx/sites-enabled/                                                 0.1s
 => exporting to image                                                                                   1.0s
 => => exporting layers                                                                                  0.9s
 => => writing image sha256:de4a1e6985f938965495766b6fdfd900c827a67aad9edb7ce59fae356d5989bd             0.0s
 => => naming to docker.io/library/2048-game                                                             0.0s

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them

C:\Users\agastya\OneDrive\Desktop\Docker Projects\2048>docker images
REPOSITORY                                TAG       IMAGE ID       CREATED          SIZE
2048-game                                 latest    de4a1e6985f9   9 seconds ago    250MB
```

- Image is built and ready to be run in the container:



```
C:\Users\agastya\OneDrive\Desktop\Docker Projects\2048>docker images
REPOSITORY                               TAG       IMAGE ID       CREATED         SIZE
2048-game                                latest    de4a1e6985f9   9 seconds ago   250MB
<none>                                   <none>    155e6a71da6b   9 days ago      185MB
agastya0452/my_test_image_custom         latest    453acec6d834   6 weeks ago     77.8MB
my_exported_image                        latest    47e557f8d910   6 weeks ago     196MB
redis                                    alpine    2d5230e57b1b   7 weeks ago     37.8MB
ubuntu                                   latest    c6b84b685f35   2 months ago    77.8MB
mycloudregistry.azurecr.io/mytestimage   latest    f4270eef6cf8   7 months ago    77.8MB
docker/getting-started                   latest    3e4394f6b72f   10 months ago   47MB
```

- Now checking port 8080 to see if the game is ready to play and accessible :



Join the numbers and get to the **2048 tile!**

HOW TO PLAY: Use your arrow keys to move the tiles. When

So, the game is up and running on '*localhost:8080*' through Nginx server.