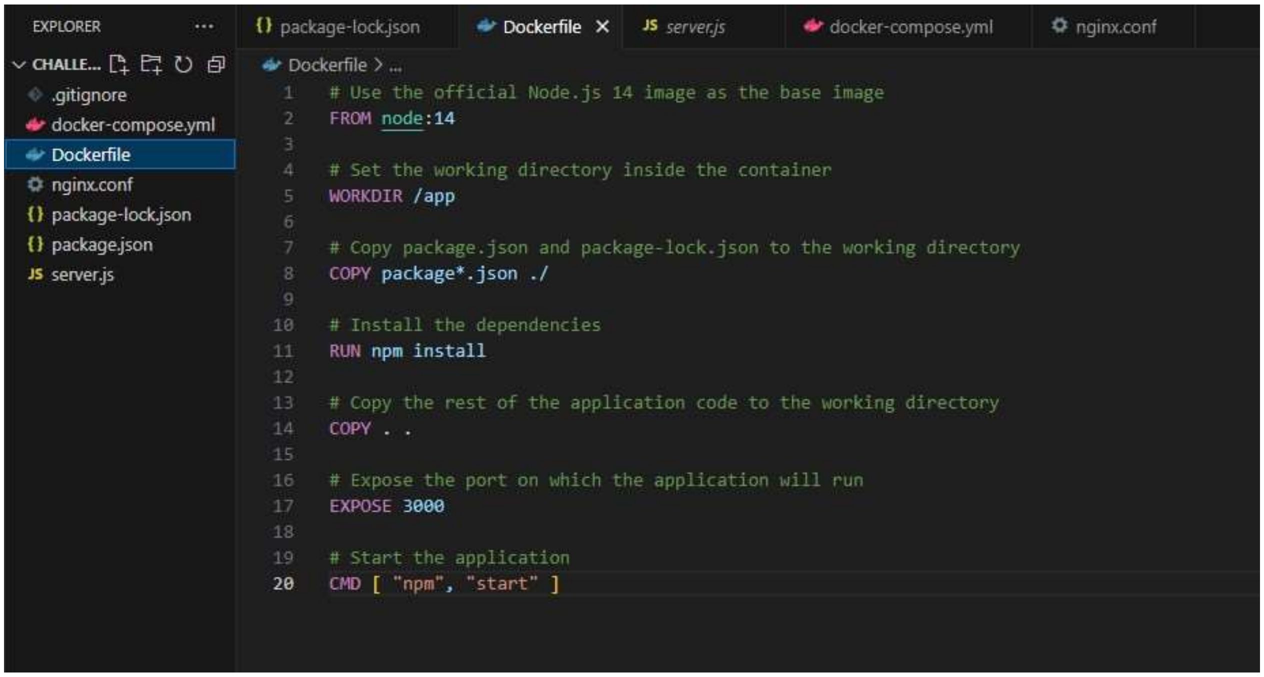
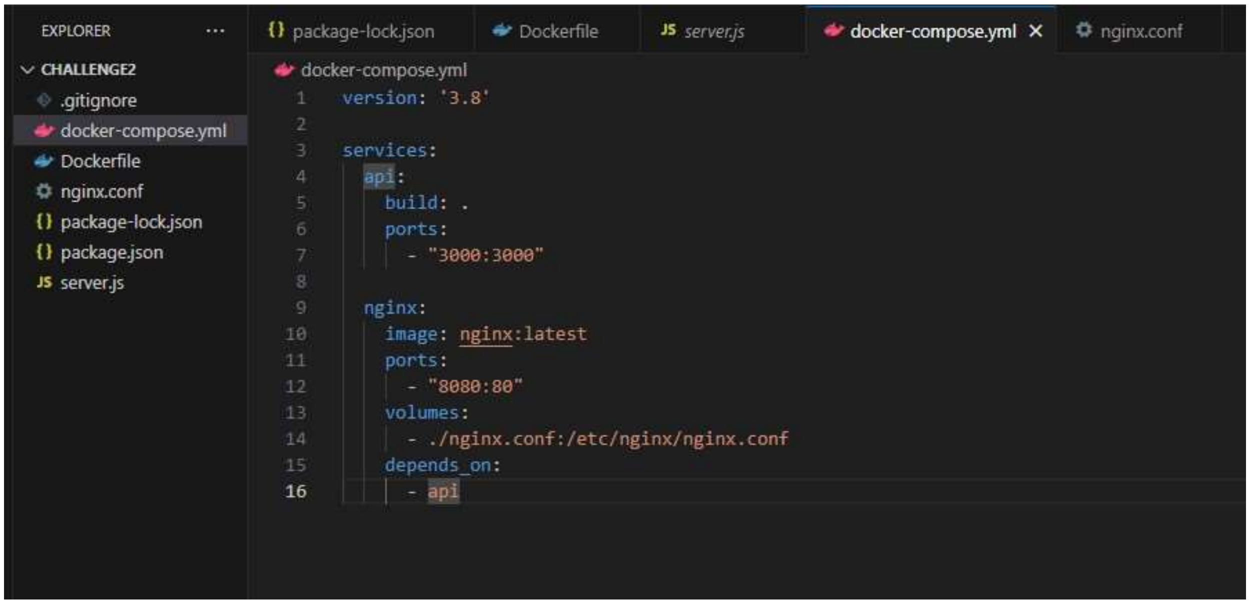
Docker Challenge 2

Abdel Mouzahir
000891579
June 28th 2024



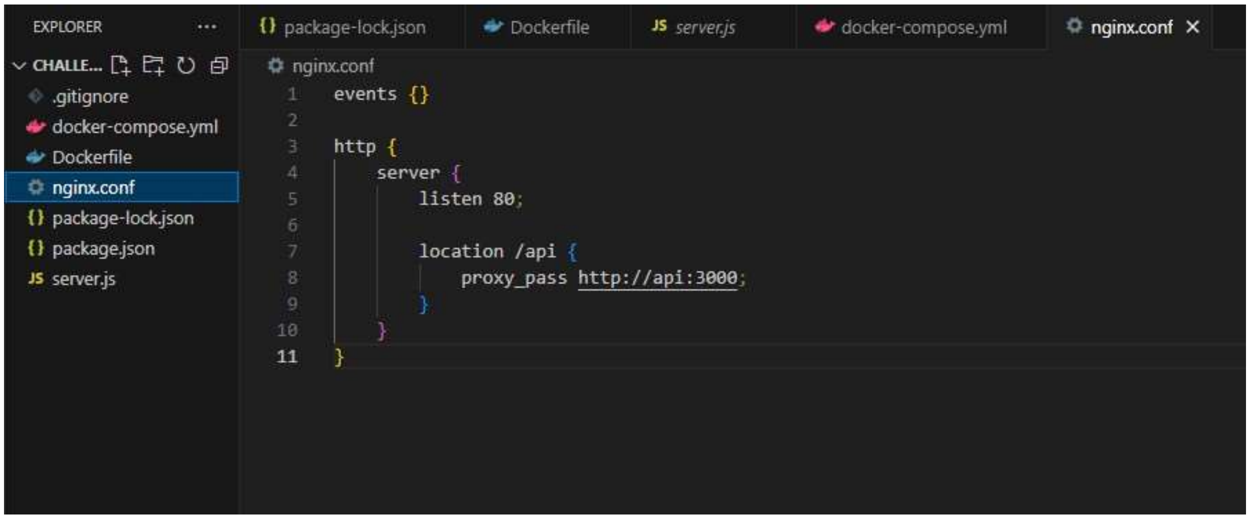
```dockerfile
# Use the official Node.js 14 image as the base image
FROM node:14

# Set the working directory inside the container
WORKDIR /app

# Copy package.json and package-lock.json to the working directory
COPY package*.json ./

# Install the dependencies
RUN npm install

# Copy the rest of the application code to the working directory
COPY . .

# Expose the port on which the application will run
EXPOSE 3000

# Start the application
CMD [ "npm", "start" ]
```

While setting up this project, I first unzipped the required json files and server into the appropriate. The first step I took was to create the dockerfile, in which we use the node version 14 image to package everything. We then insert the json files into the working directory. Next, to get everything working properly, we need to install node package manager. Finally we tell the application to listen on port 3000 (where the server will be) and to start.



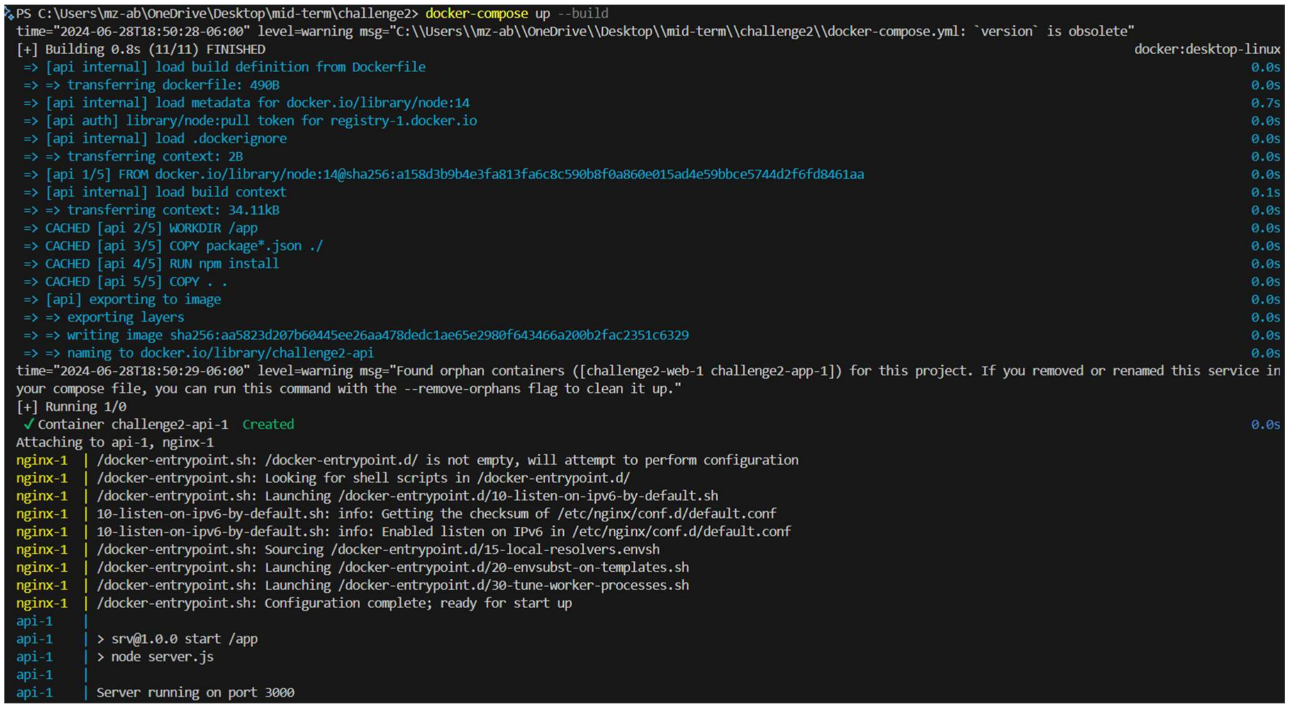
```yaml
version: '3.8'

services:
  api:
    build: .
    ports:
      - "3000:3000"

  nginx:
    image: nginx:latest
    ports:
      - "8080:80"
    volumes:
      - ./nginx.conf:/etc/nginx/nginx.conf
    depends_on:
      - api
```

This file is going to manage our separate containers used in our application. The api represents the Node.js application. The ports connect the host machine ports to the container ports. The nginx does the same, except we want the configuration file to override the default nginx. This allows the container to proxy requests to the API service.

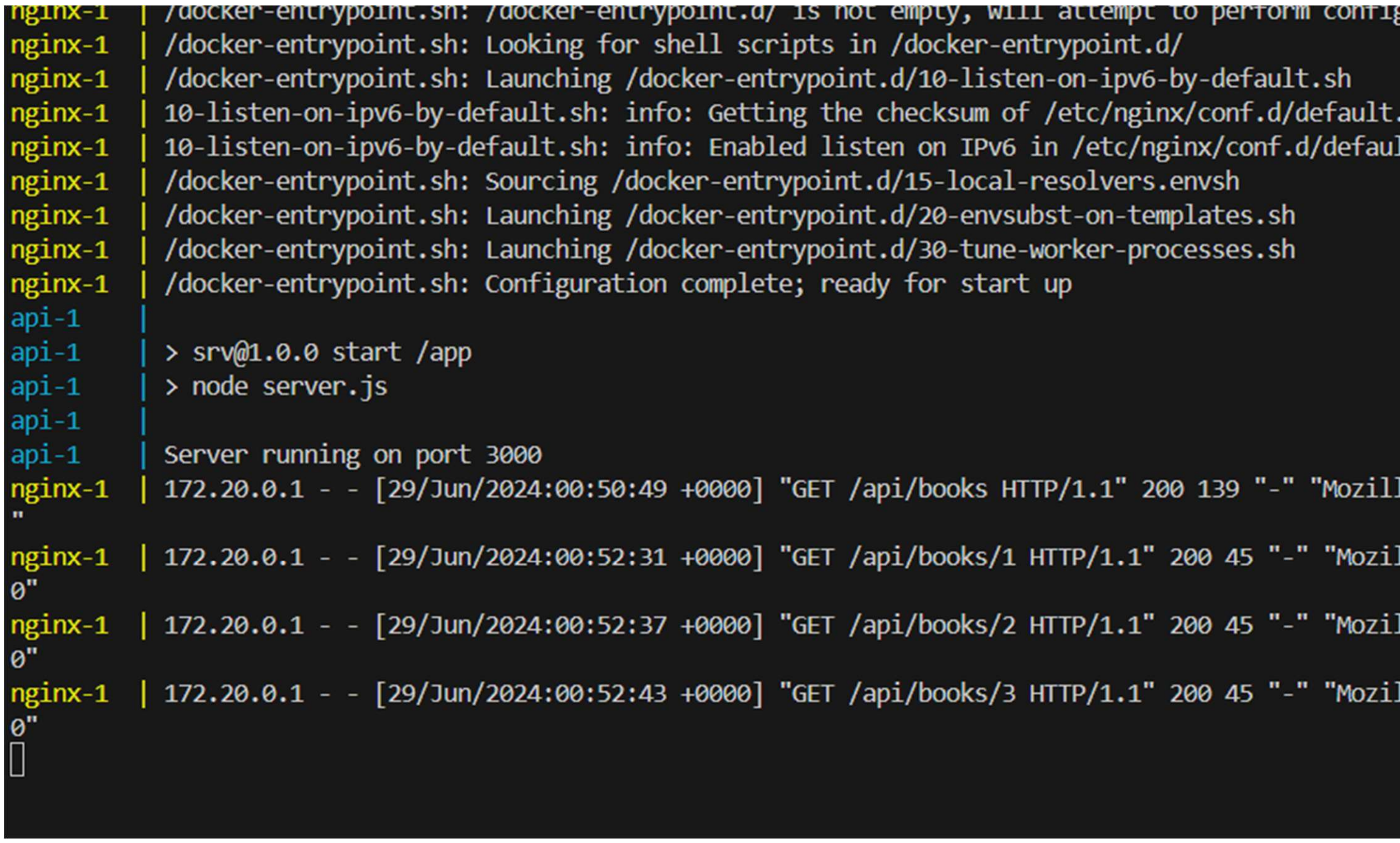


```
events {}

http {
    server {
        listen 80;

        location /api {
            proxy_pass http://api:3000;
        }
    }
}
```

This file overrides the default setting of the NGinx configuration, and tells the application to redirect any request that matches the '/api' to port 3000

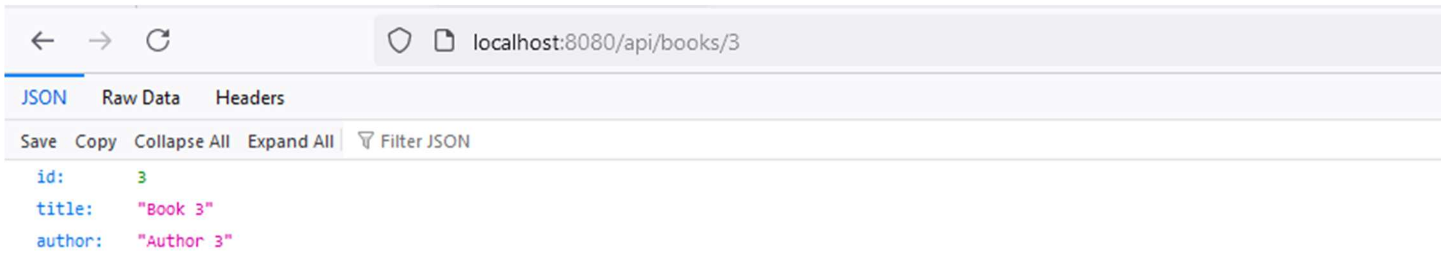


```
PS C:\Users\mz-ab\OneDrive\Desktop\mid-term\challenge2> docker-compose up --build
time="2024-06-28T18:50:28-06:00" level=warning msg="C:\\Users\\mz-ab\\OneDrive\\Desktop\\mid-term\\challenge2\\docker-compose.yml: `version` is obsolete"
[+] Building 0.8s (11/11) FINISHED                                                                                      docker:desktop-linux
 => [api internal] load build definition from Dockerfile                                                                               0.0s
 => => transferring dockerfile: 490B                                                                                                   0.0s
 => [api internal] load metadata for docker.io/library/node:14                                                                         0.7s
 => [api auth] library/node:pull token for registry-1.docker.io                                                                        0.0s
 => [api internal] load .dockerignore                                                                                                  0.0s
 => => transferring context: 2B                                                                                                        0.0s
 => [api 1/5] FROM docker.io/library/node:14@sha256:a158d3b9b4e3fa813fa6c8c590b8f0a860e015ad4e59bbce5744d2f6fd8461aa                  0.0s
 => [api internal] load build context                                                                                                  0.1s
 => => transferring context: 34.11kB                                                                                                   0.0s
 => CACHED [api 2/5] WORKDIR /app                                                                                                      0.0s
 => CACHED [api 3/5] COPY package*.json ./                                                                                             0.0s
 => CACHED [api 4/5] RUN npm install                                                                                                   0.0s
 => CACHED [api 5/5] COPY . .                                                                                                          0.0s
 => [api] exporting to image                                                                                                           0.0s
 => => exporting layers                                                                                                                0.0s
 => => writing image sha256:aa5823d207b60445ee26aa478dedc1ae65e2980f643466a200b2fac2351c6329                                           0.0s
 => => naming to docker.io/library/challenge2-api                                                                                      0.0s
time="2024-06-28T18:50:29-06:00" level=warning msg="Found orphan containers ([challenge2-web-1 challenge2-app-1]) for this project. If you removed or renamed this service in
your compose file, you can run this command with the --remove-orphans flag to clean it up."
[+] Running 1/0
 ✓ Container challenge2-api-1  Created                                                                                                 0.0s
Attaching to api-1, nginx-1
nginx-1  | /docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
nginx-1  | /docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
nginx-1  | /docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
nginx-1  | 10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
nginx-1  | 10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
nginx-1  | /docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
nginx-1  | /docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
nginx-1  | /docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
nginx-1  | /docker-entrypoint.sh: Configuration complete; ready for start up
api-1    |
api-1    | > srv@1.0.0 start /app
api-1    | > node server.js
api-1    |
api-1    | Server running on port 3000
```

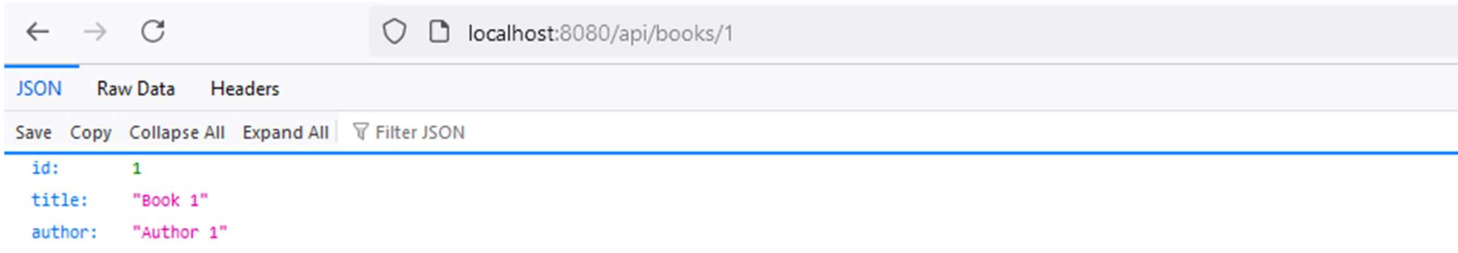


```
nginx-1  | /docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform config
nginx-1  | /docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
nginx-1  | /docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
nginx-1  | 10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.
nginx-1  | 10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/defaul
nginx-1  | /docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
nginx-1  | /docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
nginx-1  | /docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
nginx-1  | /docker-entrypoint.sh: Configuration complete; ready for start up
api-1    |
api-1    | > srv@1.0.0 start /app
api-1    | > node server.js
api-1    |
api-1    | Server running on port 3000
nginx-1  | 172.20.0.1 - - [29/Jun/2024:00:50:49 +0000] "GET /api/books HTTP/1.1" 200 139 "-" "Mozill
"
nginx-1  | 172.20.0.1 - - [29/Jun/2024:00:52:31 +0000] "GET /api/books/1 HTTP/1.1" 200 45 "-" "Mozil
0"
nginx-1  | 172.20.0.1 - - [29/Jun/2024:00:52:37 +0000] "GET /api/books/2 HTTP/1.1" 200 45 "-" "Mozil
0"
nginx-1  | 172.20.0.1 - - [29/Jun/2024:00:52:43 +0000] "GET /api/books/3 HTTP/1.1" 200 45 "-" "Mozil
0"
```

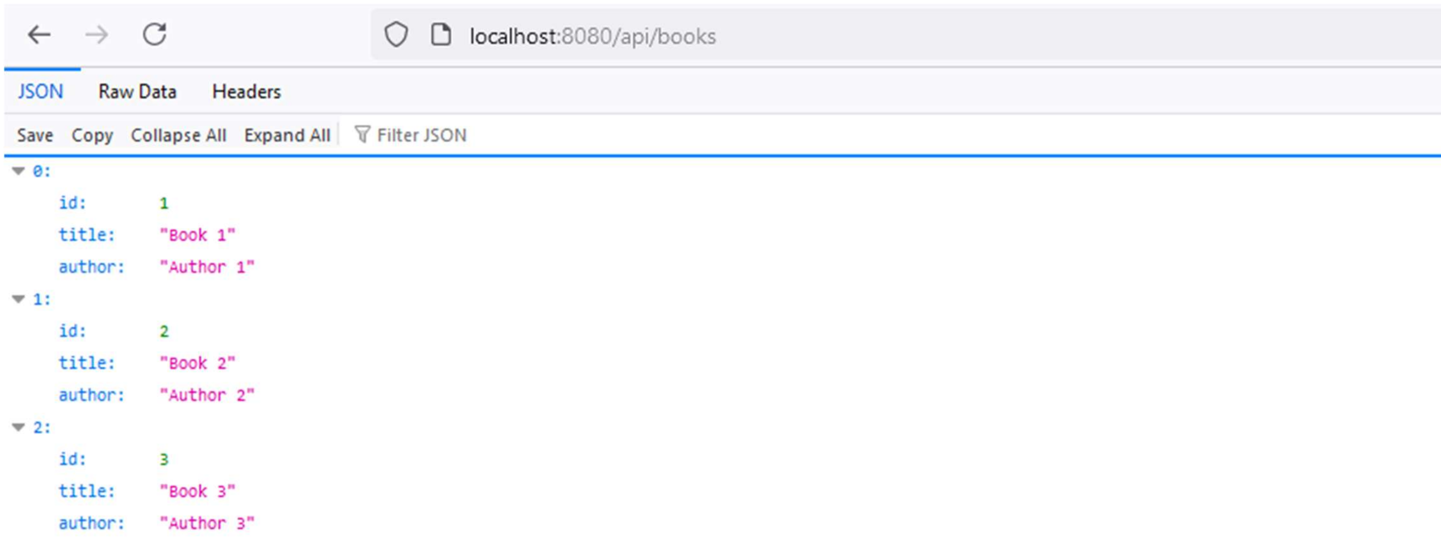
Finally we open docker desktop, then we can open a command line in powershell. I navigated to the challenge 2 folder, and used the command ‘docker-compose up - -build’ to build the container


localhost:8080/api/books/3

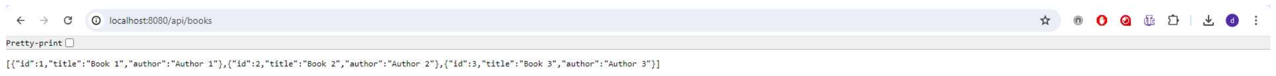
```
id:       3
title:    "Book 3"
author:   "Author 3"
```


localhost:8080/api/books/1

```
id:       1
title:    "Book 1"
author:   "Author 1"
```


localhost:8080/api/books

```
▼ 0:
    id:       1
    title:    "Book 1"
    author:   "Author 1"
▼ 1:
    id:       2
    title:    "Book 2"
    author:   "Author 2"
▼ 2:
    id:       3
    title:    "Book 3"
    author:   "Author 3"
```


localhost:8080/api/books

```
[{"id":1,"title":"Book 1","author":"Author 1"},{"id":2,"title":"Book 2","author":"Author 2"},{"id":3,"title":"Book 3","author":"Author 3"}]
```

Once the container had finished building, I opened a browser (in this case google chrome) and tested both the localhost:8080/api/books and the localhost:8080/api/books/1 to ensure that I was receiving the correct json call.

References:

https://docs.docker.com/compose/

https://www.simplilearn.com/tutorials/docker-tutorial/docker-compose

https://www.baeldung.com/ops/docker-compose

https://www.youtube.com/watch?v=JiJeZOHx0ow&ab_channel=AmichaiMantinband