Name: Osama Abdul Razzak(2303.KHI.DEG.029)
Peer Name: Rahima Siddiqui(2303.KHI.DEG.030)

Assignment 4.4

Browse to:

tasks/4_microservices_development/day_4_best_practices/
app_that_doesnt_follow_best_practices/

Analyze the application - which Microservice best practices does it not follow?

Think about what needs to be improved first. Have a look at the areas_for_improvement.txt file for hints.

Improve the application.

Improvement required:

he following aspects need to be improved in the application:

- The logs shouldn’t written to a file, but to the container output.
- It should be stateless, so that:
  - it can easily be restarted without loss of data,
  - it is easy to spawn multiple instances of the application.
- Requirements installation should be moved from runtime to build time.
- App should be able to be executed both during development, with debugging enabled, and in production, with debugging disabled.
- The application should be built in such a way that the database can easily be replaced (development with production instance).

Name: Osama Abdul Razzak(2303.KHI.DEG.029)
Peer Name: Rahima Siddiqui(2303.KHI.DEG.030)

Modify the main.py

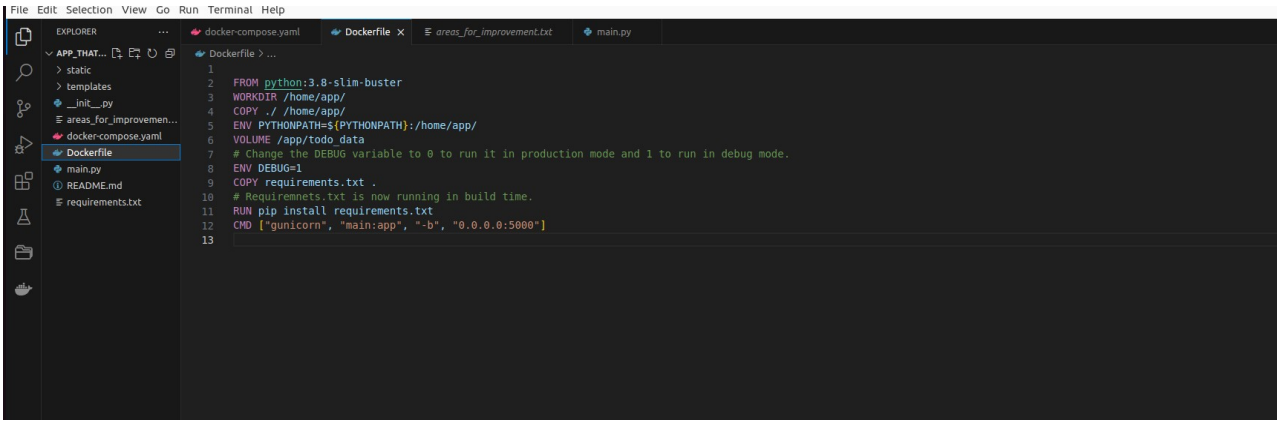
```python
import json
import logging
import os
import sys
from flask import Flask, render_template, request

# Problem  -  App should be able to be executed both during development, with debugging enabled, and in production, with debugging disabled.
app = Flask(__name__)

# This converts the retrieved value to a boolean. If the value is truthy (i.e. not empty, zero, or None),
#it returns True; otherwise, it returns False
app.debug = bool(os.environ.get('DEBUG', False))
# Problem 1 - The logs shouldn't written to a file, but to the container output.


logging.basicConfig(
    stream=sys.stdout,
    level=logging.INFO,
    format="%(asctime)s,%(msecs)d %(name)s %(levelname)s %(message)s",
    datefmt="%H:%M:%S",
)
# the logs will be printed to the console or container output rather than being written to a file

TODO_FILE_NAME = "/app/todo_data/todo.json"
#You can change the Path anytime when you want to switch from test to production

if os.path.exists(TODO_FILE_NAME):
    with open(TODO_FILE_NAME) as f:
        TODO_ITEMS = json.load(f)
else:
    TODO_ITEMS = []


@app.route("/", methods=["GET", "POST"])
def main():
    if request.method == "POST":
        content = request.form["content"]
        TODO_ITEMS.append(content)
        save_todo_items()

    return render_template("index.html", todo_items=TODO_ITEMS)

# This function saves the list on each call making it stateless instead of periodic saving.
def save_todo_items():
    with open(TODO_FILE_NAME, "w") as f:
        json.dump(TODO_ITEMS, f)

# Checking wheather the app is in debug mode or not.
if app.debug:
    app.logger.setLevel(logging.DEBUG)
    app.logger.debug("Debug mode is enabled.")
else:
    app.logger.debug("Debug mode is disabled.")


if __name__ == "__main__":
    app.run(host="0.0.0.0")
```

Create the docker compose.yaml file
Map container port for the replicas
Called our Dockerfile from there
Create volume so it won't lost data, after container kill or restart and save json data in local host

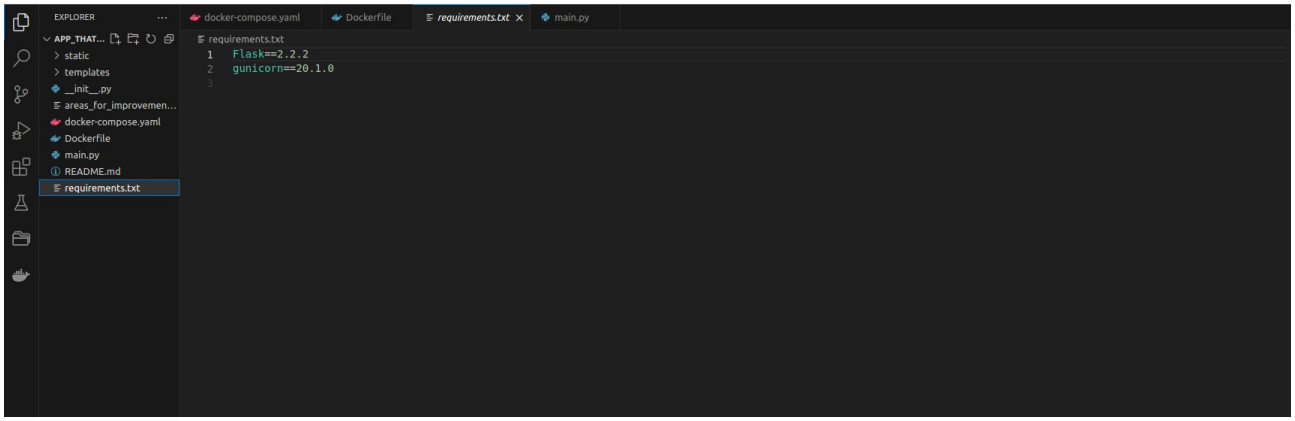
```yaml
app:
  build:
    context: .
    dockerfile: Dockerfile
  ports:
    - '5001-5003:5000'  # Map container port for the  replicas

  volumes:
    - todo_data:/app/todo_data
  deploy:
    replicas: 3

volumes:
  todo_data:

# A persistant storage paired with a dedicated volume is assigned so that app is restarted without the loss of data even when container is stopped.
# You can replicate this container or spawn multiple instances and still expect the same results.
```

Name: Osama Abdul Razzak(2303.KHI.DEG.029)
Peer Name: Rahima Siddiqui(2303.KHI.DEG.030)

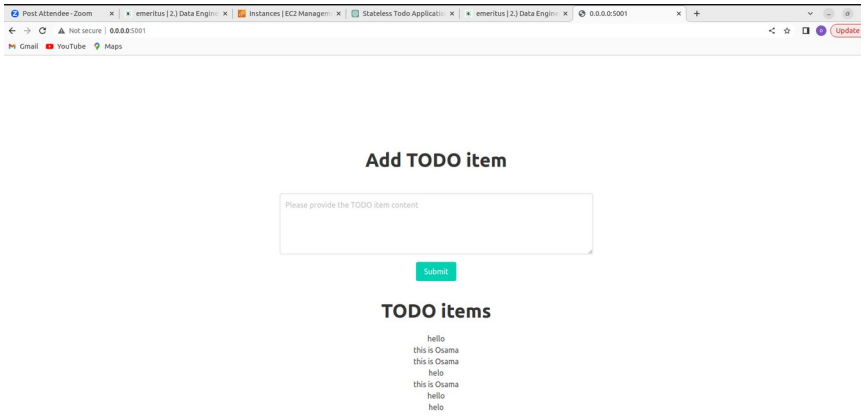
And here is our Dockerfile, which is being called in our Dockercompose file and reqiurement.txt is moved from run to build time



And here is requirement.txt file

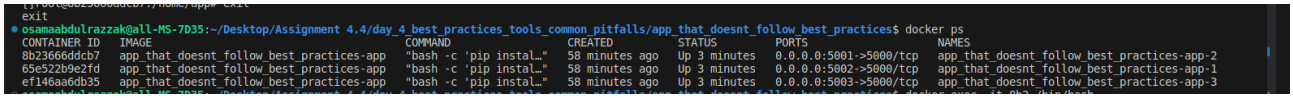


here is web ui



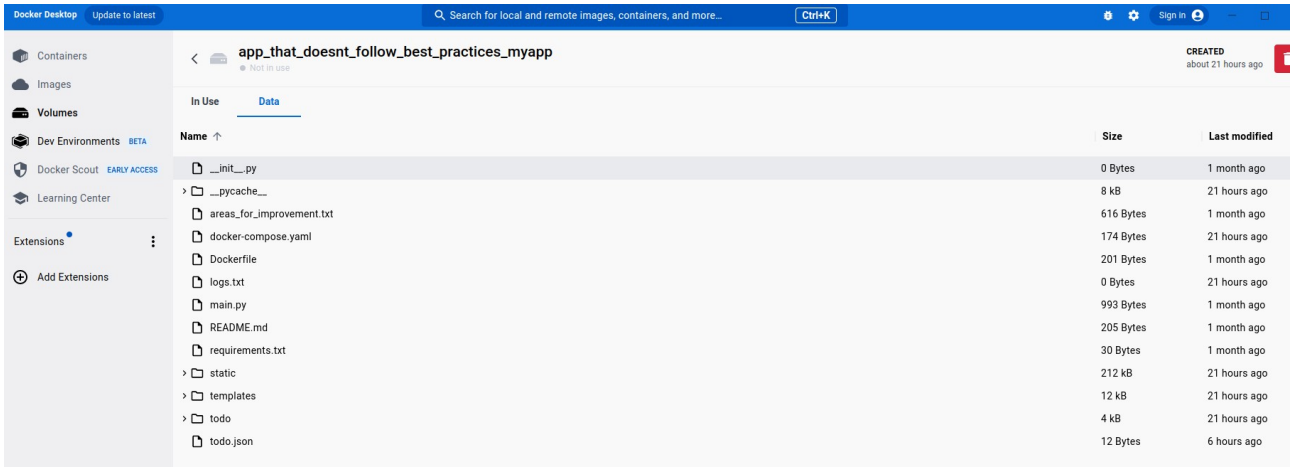
Name: Osama Abdul Razzak(2303.KHI.DEG.029)
Peer Name: Rahima Siddiqui(2303.KHI.DEG.030)

here is replicas of port



```
exit
osamaabdulrazzak@all-MS-7D35:~/Desktop/Assignment 4.4/day_4_best_practices_tools_common_pitfalls/app_that_doesnt_follow_best_practices$ docker ps
CONTAINER ID   IMAGE                                   COMMAND                  CREATED        STATUS         PORTS                    NAMES
8b23666ddcb7   app_that_doesnt_follow_best_practices-app   "bash -c 'pip instal…"   58 minutes ago   Up 3 minutes   0.0.0.0:5001->5000/tcp   app_that_doesnt_follow_best_practices-app-2
65e522b9e2fd   app_that_doesnt_follow_best_practices-app   "bash -c 'pip instal…"   58 minutes ago   Up 3 minutes   0.0.0.0:5002->5000/tcp   app_that_doesnt_follow_best_practices-app-1
ef146aa6db35   app_that_doesnt_follow_best_practices-app   "bash -c 'pip instal…"   58 minutes ago   Up 3 minutes   0.0.0.0:5003->5000/tcp   app_that_doesnt_follow_best_practices-app-3
```

and it will store value in todo.json and it will not lost after terminating or restart the container



and for your reference file we also share all related file