# Qoala Report
# Tarak Agarwal 21ucc102

## Issues Faced

### 1. Python Dockerfile Issues:

- **Incorrect Syntax:** The line **`WORKDIR/appp`** was incorrectly specified. This was corrected to **`WORKDIR/app`**.

- **File Copying Errors:** The command **`COPY appp.py/app`** contained a typographical error; it was rectified to **`COPY app.py/app`.**

- **Package Name Typo:** The package name **`netiface`** was corrected to **`netifaces`**.

- **Port Specification:** The port `8000` was originally formatted as a string; this was corrected to an integer format.

- **CMD Syntax Error:** The command `CMD ["python", "app.py"]` was corrected for proper syntax.

### 2. Nginx Dockerfile Issues:

- **Syntax Errors:** In the Nginx Dockerfile, the term **`latests`** was corrected to **`latest`**, and directory paths were updated accordingly.

- **Port Exposure Syntax:** The syntax for `EXPOSE` was changed from **`eighty`** to **`80`**.

- **CMD Syntax Correction:** The command was changed from **`daemon of`** to **`daemon off`**.

### 3. Directory Structure Issue:

- Following the Nginx build, an error indicated the absence of a directory named `html` at the specified path. In response, I removed the **`COPY ./html /usr/share/nginx/html`** line.

### 4. Nginx Configuration File Errors:

- An error was identified in the `nginx.conf` file at line 3, specifically regarding the directive `worker_process`. Subsequent examination revealed several syntax errors:

- The directive **`worker_connection`** was corrected to **`worker_connections`**.

- The line **`include /etc/nginx/mime.typess;`** was updated to **`include /etc/nginx/mime.types;`**.

- The directive **`default_typ`** was corrected to **`default_type`**.

**5. `docker-compose.yaml` Errors:**

- Ports in services were correctly mapped to 80

- Typo in the container name (python-app vs python_app).

**<u>Resolution Steps</u>**

**1. Building the Python Image:**

- After rectifying the aforementioned issues in the Python Dockerfile, I executed the command **`docker build -t local-python-app`,** resulting in a successful image build.

**2. Building the Nginx Image:**

- Following the corrections made to the Nginx Dockerfile, I attempted to build the image. Upon addressing the directory structure issue and verifying the contents of the `nginx.conf`, the Nginx image was successfully built.

**3. Container Management:**

- After resolving all issues, I rebuilt and restarted the containers. I verified their status by executing `docker ps`.

**4. Testing Application Communication:**

- I tested the communication between the Nginx and Python application by executing `curl http://localhost:80`. This confirmed that the Nginx server was functioning correctly and forwarding requests to the Python application.

**5. Logging Verification:**

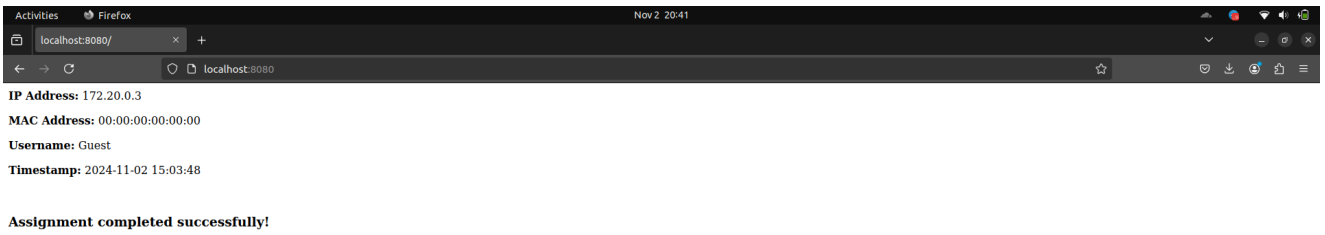
- Finally, I reviewed the logs for both the Nginx container using **`docker logs local-nginx`** and for the Python application using **`docker logs local-python-app`** to ensure that there were no further errors and that the applications were running as expected.

```
tarak@tarak-Inspiron-5415:~/Projects/qoala$ docker logs nginx_app
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: IPv6 listen already enabled
/docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
tarak@tarak-Inspiron-5415:~/Projects/qoala$
```

Nginx Logs

```
tarak@tarak-Inspiron-5415:~/Projects/qoala$ docker logs python_app
 * Serving Flask app 'app'
 * Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server ins
tead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:8000
 * Running on http://172.21.0.2:8000
Press CTRL+C to quit
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 523-075-557
172.21.0.3 - - [02/Nov/2024 15:09:42] "GET / HTTP/1.0" 200 -
 * Serving Flask app 'app'
 * Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server ins
tead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:8000
 * Running on http://172.21.0.2:8000
Press CTRL+C to quit
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 523-075-557
172.21.0.3 - - [02/Nov/2024 18:59:53] "GET / HTTP/1.0" 200 -
172.21.0.3 - - [02/Nov/2024 18:59:53] "GET /favicon.ico HTTP/1.0" 404 -
```

Python App logs




Application Running on the Browser