Student Name – Matriculation Number: Ayesha Begum – A0227071M

GitHub repository: <https://github.com/Ayesh02/Task1A>

**Task 1.1: Dockerize the given node app in the “app” folder**

Instructions on how to run the Docker container:

* Create Dockerfile in the same directory as package.json
* Run “**docker build -t ayesha/node-web-app**” in the app directory. Tagging it for ease of running it later
* Run “**docker images**” to see if the image is added
* Run “**docker run -p 3000:8080 -d ayesha/node-web-app**” to run the image
* Browse “**http://localhost:3000**”

Graphical user interface, application

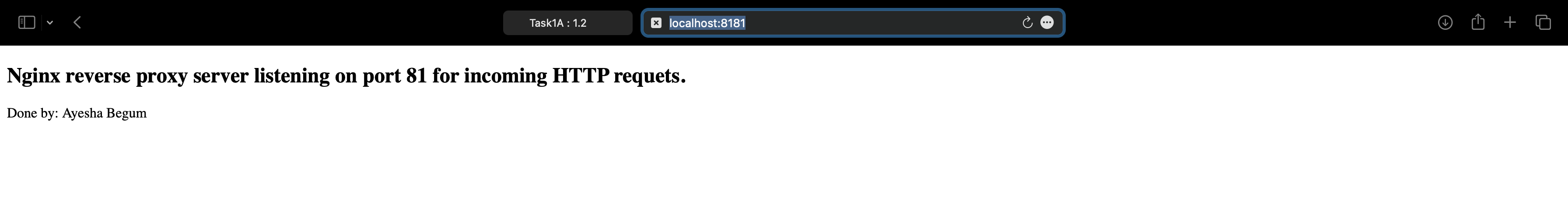
Description automatically generated

**Task 1.2: Dockerize NGINX to serve the static html page in the “nginx-sample” folder**

Instructions on how to run the Docker container:

1. Serving static html AS **single-html**:

* Create Dockerfile inside nginx-sample folder
* Run “**docker run -d -p 8181:80 –name single-html static-nginx**”
* Browse “**http://localhost:8181**”



1. Running NGINX with reverse proxy AS **nginx-base**:

* Run “**sudo docker run -d --name nginx-base -p 81:80 nginx:latest**”
* Run “**sudo docker cp nginx-base-2:/etc/nginx/conf.d/default.conf ~/Desktop/default.conf**”
* Run “**sudo nano default.conf**”
* Add upstream server pointing to the single-html and add location /sample calling upstream via proxy-pass
* Run “**sudo docker cp ~/Desktop/default.conf nginx-base-2:/etc/nginx/conf.d/**”
* Run “**sudo docker exec nginx-base nginx -t**”
* Run “**sudo docker exec nginx-base nginx -s reload**”
* Browse “**http://localhost:81/sample**”

Graphical user interface, application

Description automatically generated

**Task 1.3: Task 1.3: Combining knowledge of docker and NGINX reverse proxy, use NGINX (in “nginx” folder) to serve the node app (“app” folder) using docker-compose**

Instructions on how to run the Docker container:

1. Add Dockerfile and nginx.conf inside nginx folder
2. Add docker-compose.yml at root folder
3. Run “**docker-compose up -d**”
4. Browse “[**http://localhost:8200**](http://localhost:8200)” 🡨 nodejs application
5. Browse “[**http://localhost:82**](http://localhost:82)” 🡨 nginx running nodejs application

