Assignment: Containerization with Docker Report

## Personal

* Name: Zackaria Osman
* ID: 000885686
* Email:
  + Primary: [cobalt.zr86@gmail.com](mailto:cobalt.zr86@gmail.com)
  + Secondary: [zackaria.osman@edu.sait.ca](mailto:zackaria.osman@edu.sait.ca)

## Links

GitHub repository: <https://github.com/ZackariaOsman/docker-challenge-template>

## Basic introduction of Docker

* Why is it important for me, in the context of software development?
* Docker commands have you used?

\*\*\* ’.’ = current working directory \*\*\*

* docker build
  + Builds an image from a Dockerfile in the current directory.
* docker images
  + Lists all available Docker images.
* docker build -t <repository\_name>
  + Builds an image and tags it with a repository name.
* docker run -d -p <host\_port>:<container\_port> <image\_id>
  + Runs a container in detached mode, mapping ports between host and container.
* docker ps
  + Lists running containers.
* docker stop <container\_id> or docker stop <container\_name>:
  + Stops a running container.
* docker ps -a
  + Lists all containers (both running and stopped).
* docker rm <container\_id> or docker rm <container\_name>
  + Removes a container.
* docker rmi <image\_id> or docker rmi <image\_name>
  + Removes an image.

## Which steps I went through to make it work

**Installation**

1. **Install Docker**: Downloaded the Docker Engine from the Docker website followed the instructions for the installation process.

**Configuration**

1. Creating a DockerFile:

# To use the Nginx image from Docker Hub

FROM nginx:alpine

# Copy  contents of  public folder to Nginx html directory

COPY public/ /usr/share/nginx/html/

# Expose port 80 but it is not mandatory

#EXPOSE 80

# Start Nginx when container starts

#Also not mandatory

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

#For Nginx in debug mode

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

1. **Building an Image**:
   * Use docker build . to build an image from a Dockerfile located in the current directory (.).
   * Optionally, you can tag the image with a repository name using docker build -t <repository\_name> ..
2. **Listing Images**:
   * After building, use docker images to list all locally available Docker images.

**Creation of Files**

1. **Creating and Running Containers**:
   * Create a container from an image with docker run:
     + Use docker run -d -p <host\_port>:<container\_port> <image\_id> to run a container in detached mode (-d) and map ports (-p). Replace <host\_port> and <container\_port> with appropriate values.
     + Example: docker run -d -p 80:80 8b0a2c43a149 runs a container from image 8b0a2c43a149, mapping host port 80 to container port 80.
2. **Managing Containers**:
   * View running containers with docker ps.
   * Stop a container using docker stop <container\_id> or docker stop <container\_name>.
   * List all containers (including stopped ones) with docker ps -a.
   * Remove a container with docker rm <container\_id> or docker rm <container\_name>.
3. **Managing Images**:
   * Remove an image using docker rmi <image\_id> or docker rmi <image\_name>.

Screenshots

Png1:

A screen shot of a computer screen

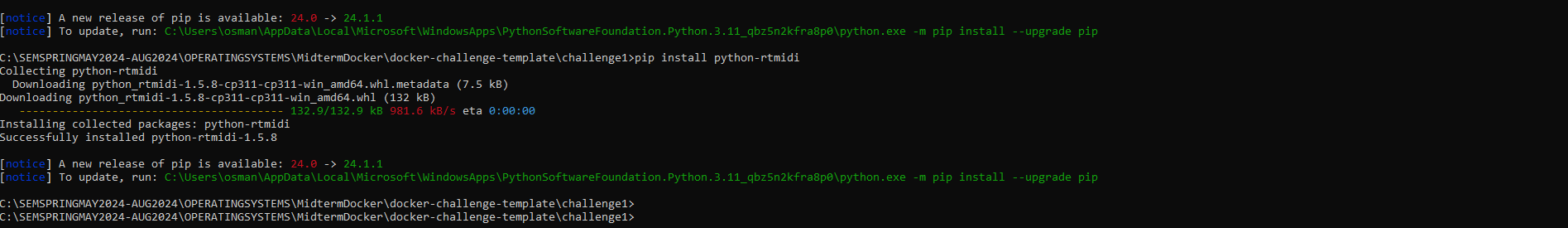
Description automatically generated

Png2:

A screen shot of a computer program

Description automatically generated

Png3:



## Lessons learned

* Something went wrong during your work?
  + Mention it.
  + How do you overcome the issues?

## References

* YouTube: <https://www.youtube.com/watch?v=SnSH8Ht3MIc&list=LL&index=7&t=929s>