Lab: Getting Started with Docker

This lab will guide you through Docker's core features, including containerization, image management, and networking.

Objective

1. **Understand basic Docker concepts.**
2. **Create and run Docker containers.**
3. **Build and push a custom Docker image.**
4. **Network containers for a multi-container setup.**

Lab Setup

Prerequisites:

- **Install Docker https://docs.docker.com/engine/install/ubuntu/**
- **Familiarity with basic command-line operations**

Step 1: Run Your First Docker Container

1. **Start a Container:**

Use the following command to pull and run an `nginx` container:

```
docker run -d -p 8080:80 --name my-nginx nginx
```

    -
    - **Explanation:**
        - `-d`**: Run in detached mode (background).**
        - `-p 8080:80`**: Map port 80 inside the container to port 8080 on your machine.**
        - `--name my-nginx`**: Name the container `my-nginx`.**
2. **Verify it's Running:**
    - **Run `docker ps` to see your active containers.**
    - **Open a web browser and go to `http://localhost:8080` to see the Nginx welcome page.**
3. **Stop and Remove the Container:**

Stop the container:

```
docker stop my-nginx
```

-

Remove the container:

```
docker rm my-nginx
```

-

Step 2: Building a Custom Docker Image

1. **Create a Simple Web App:**

In a new directory, create a file named `index.html` with the following content:

```
<html>
  <body>
    <h1>Welcome to my custom Docker container!</h1>
  </body>
</html>
```

-

In the same directory, create a `Dockerfile`:
Dockerfile

```
FROM nginx:latest
COPY index.html /usr/share/nginx/html/index.html
```

-

2. **Build the Docker Image:**

Run the following command to build your custom image:

```
docker build -t my-custom-nginx .
```

-
- **Explanation:**
  - `-t my-custom-nginx`**: Tags the image with the name** `my-custom-nginx`**.**

        - `.`: Specifies the current directory as the build context.
3. **Run Your Custom Image:**

Start a container with your new image:

```
docker run -d -p 8081:80 --name custom-nginx my-custom-nginx
```

-
- **Go to `http://localhost:8081` in a web browser to see your custom web page.**

Step 3: Docker Networking

1. **Create a Network:**

Create a new network:

```
docker network create my-network
```

-
2. **Run a Database Container on the Network:**

Run a `mysql` container on the network:

```
docker run -d --name my-mysql --network my-network -e
MYSQL_ROOT_PASSWORD=root mysql
```

-
3. **Link Your App to the Database:**
    - **Run another container that connects to `my-mysql` on the `my-network`.**

Step 4: Push Your Image to Docker Hub

Tag the Image:

```
docker tag my-custom-nginx <your_dockerhub_username>/my-custom-nginx
```

1. **Push the Image:**

Log in to Docker Hub:

```
docker login
```

Push the image:

```
docker push <your_dockerhub_username>/my-custom-nginx
```

Resources

https://www.digitalocean.com/community/tutorials/how-to-install-and-use-docker-on-ubuntu-20-04
https://docs.docker.com/engine/install/ubuntu/