COMSATS UNIVERSITY ISLAMABAD

ATTOCK CAMPUS



LAB ASSIGNMENT #1

NAME: Abbas Inayatullah Khan

REG. NO.: SP22-BSE-020

DATE: March 5th, 2025

SUBJECT: DevOps For Cloud Computing

**Overview:**

This assignment is a demonstration of implementation of containerization using Dockers. I created a python file that simply shows a dummy text, then I created an image of that file using docker. Then using docker cli, I packed that image in a docker container. The complete implementation can be seen through the following screenshots.

The complete project can be forked from my GitHub repository through the following link:

[DevOps/Guide to working with Dockers at main · flickShot555/DevOps](https://github.com/flickShot555/DevOps/tree/main/Guide%20to%20working%20with%20Dockers)

**Step 1:**

I had already installed and integrated the docker, so my verification was seamless. Figure 1 shows the command “docker –version”.

**Step 2:**

Then I created a simple python application, that displayed a dummy text. The back-end was python coded and the front-end was rendered using html.

Code for app.py and index.html are as follows:

**Index.html**

<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1.0">  
 <title>Weather App</title>  
</head>

<body style="display:flex; flex-direction:column;align-items:center;">  
 <h1>Weather App</h1>  
 <div style="display:flex; flex-direction:column;align-items:center;">  
 <h2>This is our first docker project.</h2>  
 <h4>congratulations!!</h4>  
 </div>  
</body>  
</html>

**App.py**

from flask import Flask, render\_template  
app = Flask(\_\_name\_\_)

@app.route('/')  
def home():  
 return render\_template('index.html')  
 if \_\_name\_\_ == '\_\_main\_\_':  
 app.run(host='0.0.0.0', port=5000, debug=True)

Figure 2 shows the output of the code in the browser.

**Step 3:**

Then I containerized the python app using a dockerfile file, that had a simple template on how this app will be run when hosted on a virtual machine.

Figure 4 shows the screenshot of the dockerfile.

**Step 4:**

Then I, using docker CLI, prompted the system to build me a container for the image.

The following command created a docker image:

‘docker build -t my-first-docker-proj .’

Then to check the running images, I entered:

‘docker ps’

Figure 5 shows the output of the docker CLI after the container was built, and the total running images.

**Step 5:**

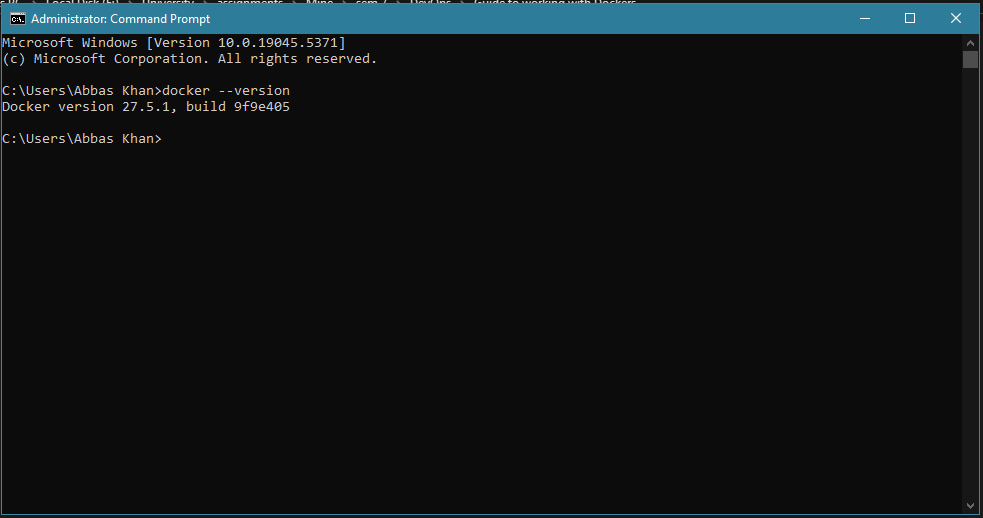
Then I, using the docker-compose file, automated the running of the app without having it live on my host docker desktop. I mapped the host server 5000 to host sever 8081 for running it independently.

Figure 6 & 7 shows the image of the web browser showing the output of the app, and the output of the ‘docker ps’ and ‘docker-compose.yml’ file.

**Step 6:**

This step mostly comprises of cleaning up process and documentation process.

**Screenshot:**

1. 
2. A screenshot of a computer

   AI-generated content may be incorrect.
3. A screenshot of a computer

   AI-generated content may be incorrect.

1. A screenshot of a computer

   AI-generated content may be incorrect.
2. A screenshot of a computer

   AI-generated content may be incorrect.
3. A screenshot of a computer

   AI-generated content may be incorrect.
4. 