Assignment -4

Assignment Kubernetes / Docker

Assignment Date	2 November 2022
Student Name	Anand Michael M
Student Roll Number	962319104017
Maximum Marks	2 Marks

Question-1:

1.Pull an Image from docker hub and run it in docker playground.

2.Create a docker file for the jobportal application and deploy it in Docker desktop application.

3.Create a IBM container registry and deploy helloworld app or jobportalapp.

4.Create a Kubernetes cluster in IBM cloud and deploy helloworld image or jobportal image and also expose the same app to run in nodeport.

Solution:

Base.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>{% block title %}{% endblock %}</title>
   <link rel="stylesheet" href="{{ url_for('static', filename='css/style.css') }}"/>
   <link rel="preconnect" href="https://fonts.googleapis.com">
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
<link href="https://fonts.googleapis.com/css2?family=Michroma&display=swap" rel="stylesheet">
</head>
<body>
   {% block main %}
   {% endblock %}
</body>
</html>
```

Index.html:

```
{% extends 'base.html' %}

{% block title %}
    Welcome
{% endblock %}

{% block main %}
    <div>
        <h2>Hello World!</h2>
        <p>The Hello World Program is Printed Successfully</p>
    </div>
{% endblock %}
```

__init__.py:

```
from flask import Flask

def create_app():
    app = Flask(__name__)

    # registering the blue print with the app
    from .views import blue_print
    app.register_blueprint(blueprint=blue_print, appendix='/')

    return app
```

Views.py:

```
from flask import Blueprint, render_template

blue_print = Blueprint("print", __name__)

@blue_print.route('/')
def home():
    return render_template('index.html')
```

App.py:

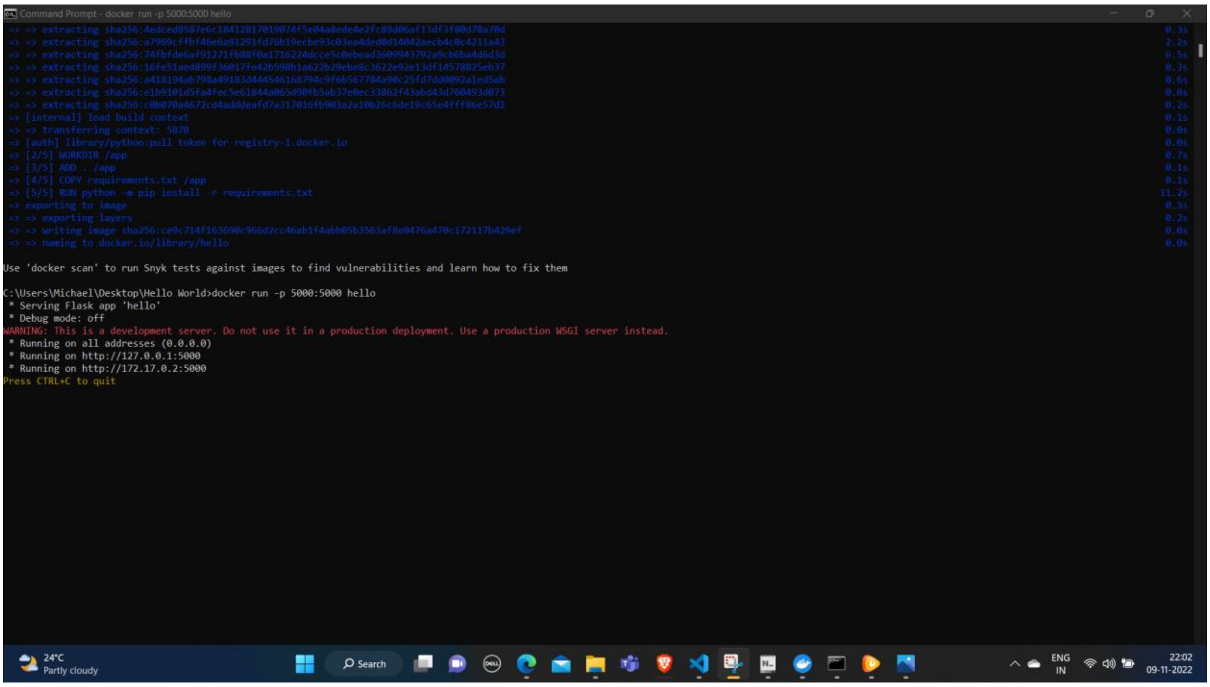
```
from venv import create
from hello import create_app

if __name__ == "__main__":
    app = create_app()
    app.run(host="0.0.0.0", port=5000)
```

Dockerfile:

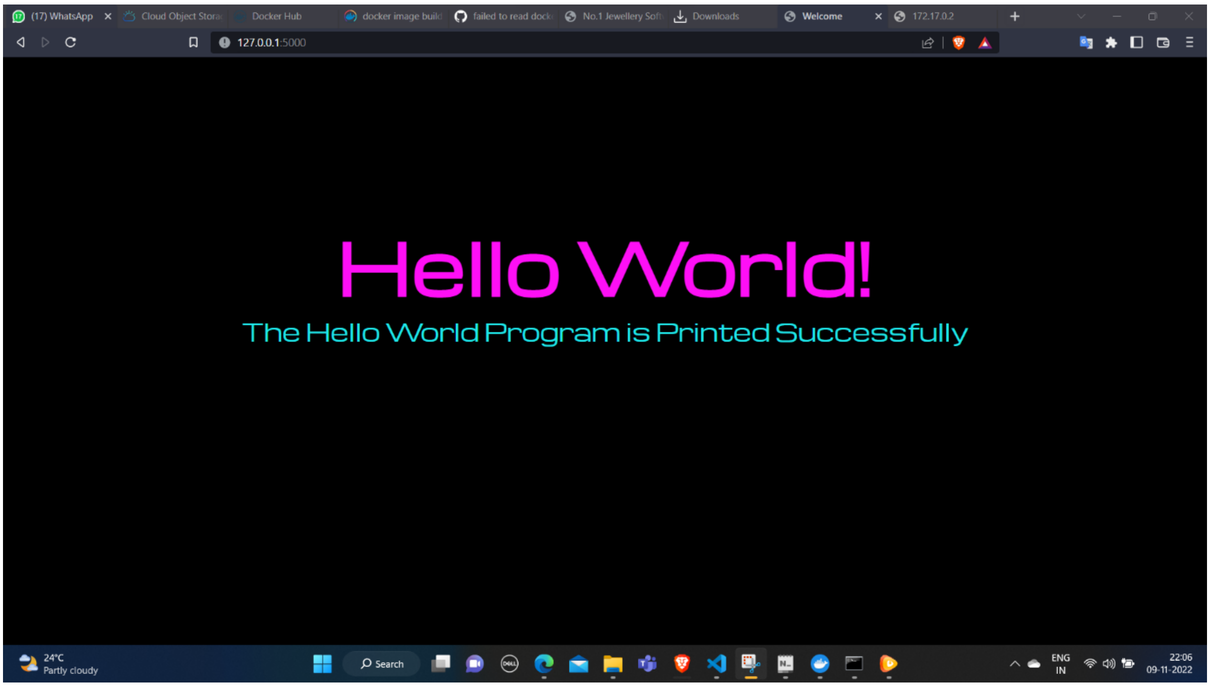
```
FROM python:3.7
WORKDIR /app
ADD . /app
COPY requirements.txt /app
RUN python -m pip install -r requirements.txt
EXPOSE 5000
ENTRYPOINT [ "python" ]
CMD [ "app.py" ]
```

Requirements.txt:

```
flask
```

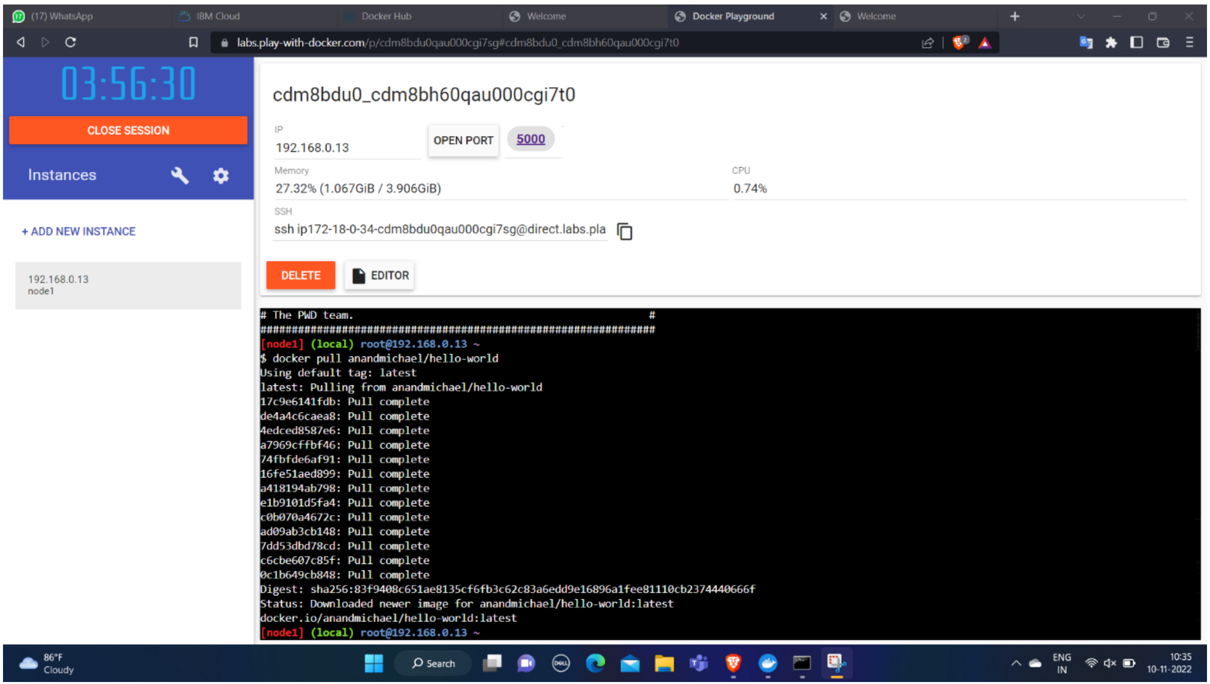


```
 => => extracting sha256:4edced8587e6c1841281701907 4f5e84a8ede4e2fc89d06af13df3f80d78a70d      0.3s
 => => extracting sha256:a7969cffbf46e6a91291fd76b19ecbe93c03ea4ded0d14042aecb4c0c4211a43      2.2s
 => => extracting sha256:74fbfde6af91271fb88f0a1716224dcce5c0ebead3609943792a9cb6ba4d6d3d      6.5s
 => => extracting sha256:16fe51aed899f36017fe42b598b1a622b29ebe8c3622e92e13df14578825eb37      0.3s
 => => extracting sha256:a418194ab798a49183d444546168794c9f6b567784a90c25fd7dd0092a1ed5ab      0.6s
 => => extracting sha256:e1b9101d5fa4fec5e61844a065d90fb5ab37e0ec33862f43abd43d760493d073      0.0s
 => => extracting sha256:c0b070a4672cd4adddeafd7a317016fb903a2a10b26c6de19c65e4fff86e57d2      0.2s
 => [internal] load build context                                                            0.1s
 => => transferring context: 587B                                                            0.0s
 => [auth] library/python:pull token for registry-1.docker.io                                0.0s
 => [2/5] WORKDIR /app                                                                       0.7s
 => [3/5] ADD . /app                                                                         0.1s
 => [4/5] COPY requirements.txt /app                                                         0.1s
 => [5/5] RUN python -m pip install -r requirements.txt                                     11.2s
 => exporting to image                                                                       0.3s
 => => exporting layers                                                                      0.2s
 => => writing image sha256:ce9c714f163690c966d2cc46ab1f4abb05b3563af8e0476a470c172117b429ef  0.0s
 => => naming to docker.io/library/hello                                                     0.0s

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them

C:\Users\Michael\Desktop\Hello World>docker run -p 5000:5000 hello
 * Serving Flask app 'hello'
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:5000
 * Running on http://172.17.0.2:5000
Press CTRL+C to quit
```



Hello World!

The Hello World Program is Printed Successfully



cdm8bdu0_cdm8bh60qau000cgi7t0

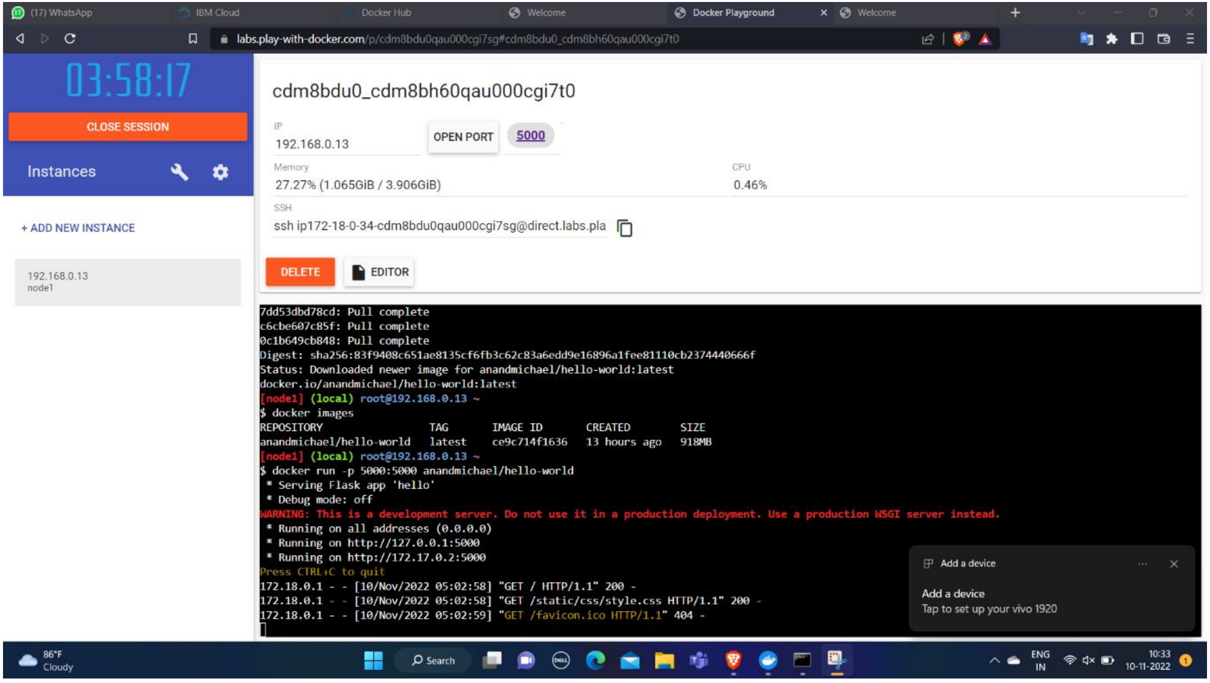
IP: 192.168.0.13

Memory: 27.32% (1.067GiB / 3.906GiB)

CPU: 0.74%

SSH: ssh ip172-18-0-34-cdm8bdu0qau000cgi7sg@direct.labs.pla

```
# The PWD team.                                                    #
###################################################################
[node1] (local) root@192.168.0.13 ~
$ docker pull anandmichael/hello-world
Using default tag: latest
latest: Pulling from anandmichael/hello-world
17c9e6141fdb: Pull complete
de4a4c6caea8: Pull complete
4edced8587e6: Pull complete
a7969cffbf46: Pull complete
74fbfde6af91: Pull complete
16fe51aed899: Pull complete
a418194ab798: Pull complete
e1b9101d5fa4: Pull complete
c0b070a4672c: Pull complete
ad09ab3cb148: Pull complete
7dd53dbd78cd: Pull complete
c6cbe607c85f: Pull complete
0c1b649cb848: Pull complete
Digest: sha256:83f9408c651ae8135cf6fb3c62c83a6edd9e16896a1fee81110cb2374440666f
Status: Downloaded newer image for anandmichael/hello-world:latest
docker.io/anandmichael/hello-world:latest
[node1] (local) root@192.168.0.13 ~
```



cdm8bdu0_cdm8bh60qau000cgi7t0

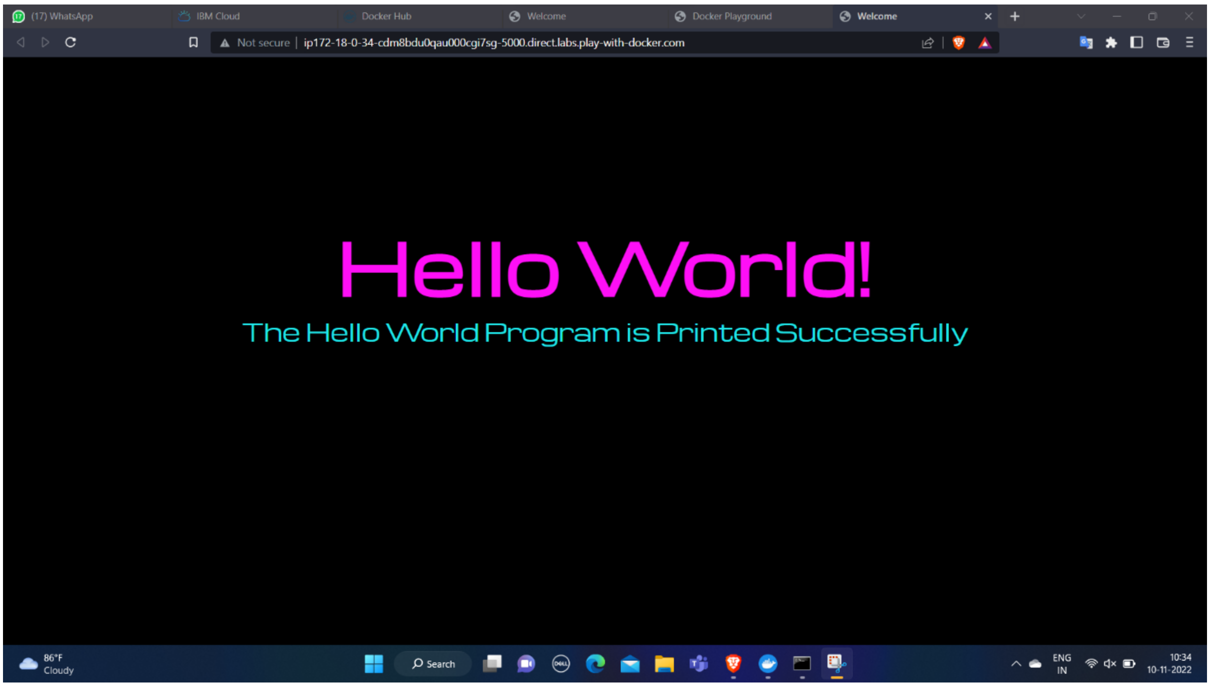
IP: 192.168.0.13

Memory: 27.27% (1.065GiB / 3.906GiB)

CPU: 0.46%

SSH: ssh ip172-18-0-34-cdm8bdu0qau000cgi7sg@direct.labs.pla

```
7dd53dbd78cd: Pull complete
c6cbe607c85f: Pull complete
0c1b649cb848: Pull complete
Digest: sha256:83f9408c651ae8135cf6fb3c62c83a6edd9e16896a1fee81110cb2374440666f
Status: Downloaded newer image for anandmichael/hello-world:latest
docker.io/anandmichael/hello-world:latest
[node1] (local) root@192.168.0.13 ~
$ docker images
REPOSITORY                 TAG       IMAGE ID       CREATED        SIZE
anandmichael/hello-world   latest    ce9c714f1636   13 hours ago   918MB
[node1] (local) root@192.168.0.13 ~
$ docker run -p 5000:5000 anandmichael/hello-world
 * Serving Flask app 'hello'
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on all addresses (0.0.0.0)
 * Running on http://127.0.0.1:5000
 * Running on http://172.17.0.2:5000
Press CTRL+C to quit
172.18.0.1 - - [10/Nov/2022 05:02:58] "GET / HTTP/1.1" 200 -
172.18.0.1 - - [10/Nov/2022 05:02:58] "GET /static/css/style.css HTTP/1.1" 200 -
172.18.0.1 - - [10/Nov/2022 05:02:59] "GET /favicon.ico HTTP/1.1" 404 -
```



Hello World!

The Hello World Program is Printed Successfully