


ASSIGNMENT-4

Date	02 November 2022
Team ID	PNT2022TMID46808
Project Name	Plasma Donor Application
Maximum Marks	2 Marks


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

 praveenabetes1 / randomapp

Description
This repository does not have a description

Last pushed: a few seconds ago

Tags and scans
This repository contains 1 tag(s).

Tag	OS	Type	Pulled	Pushed
latest		Image	---	a few seconds ago

[See all](#)[Go to Advanced Image Management](#)

Vulnerability Scanning - DISABLED [Enable](#)

Docker commands
To push a new tag to this repository,
`docker push praveenabetes1/randomapp:tagname`

[Public View](#)

Automated Builds
Manually pushing images to Hub? Connect your account to GitHub or Bitbucket to automatically build and tag new images whenever your code is updated, so you can focus your time on creating.
Available with Pro, Team and Business subscriptions.
[Upgrade](#) [Learn more](#)

03:51:24
CLOSE SESSION

Instances  

+ ADD NEW INSTANCE

192.168.0.28
node1

cdqsnus0_cdqsnve0qau0009eda6g

IP
192.168.0.28
OPEN PORT 5000

Memory
28.15% (1.1GiB / 3.906GiB)

CPU
1.21%

SSH
ssh ip172-18-0-55-cdqsnus0qau0009eda60@direct.labs.pla

DELETE EDITOR

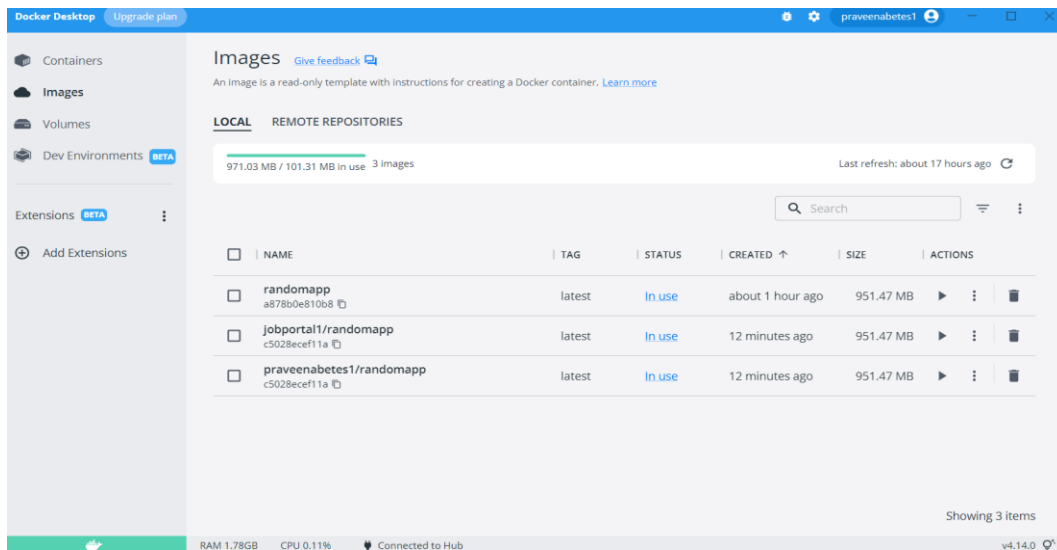
```
[node1] (local) root@192.168.0.28 ~
$ docker container run -p 5000:5000 praveenabetes1/randomapp
* Serving Flask app 'app'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://172.17.0.2:5000
press CTRL+C to quit
```

IBM x CAD-B11-5A1E (Evening Session) x Docker Hub x Docker Playground x 127.0.0.1:5000 x

← → ↺ 127.0.0.1:5000

Everything works fine

2. Create a docker file for the job portal application and deploy it in docker desktop application.



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()
```

Dockerfile

```
FROM python
WORKDIR /app
COPY . .
RUN pip install -r requirements.txt
CMD ["python", "app.py"]
```

Index.html

```
<html>
<body>
  <div class="col-md-6 offset-md-3 mt-5">

    <br>
    <a target="_blank" href="https://getform.io?ref=codepenHTML" class="mt-3 d-flex">Getform.io | Get your free endpoint now</a>
    <h1>Job Portal</h1>
    <form accept-charset="UTF-8" action="https://getform.io/f/{your-form-endpoint-goes-here}" method="POST" enctype="multipart/form-data" target="_blank">
      <div class="form-group">
        <label for="exampleInputName">Full Name</label>
```

```
        <input type="text" name="fullname" class="form-control"
id="exampleInputName" placeholder="Enter your name and surname"
required="required">
    </div>
    <div class="form-group">
        <label for="exampleInputEmail1" required="required">Email
address</label>
        <input type="email" name="email" class="form-control"
id="exampleInputEmail1" aria-describedby="emailHelp" placeholder="Enter your
email address">
    </div>
    <div class="form-group">
        <label for="inputAddress">Address</label>
        <input type="text" name="address" class="form-control"
id="inputAddress" placeholder="1234 Main St">
    </div>
    <div class="form-row">
        <div class="form-group col-md-6">
            <label for="inputCity">City</label>
            <input type="text" name="city" class="form-control" id="inputCity"
placeholder="Istanbul">
        </div>
        <div class="form-group col-md-2">
            <label for="inputZip">Zip</label>
            <input type="text" name="zip" class="form-control" id="inputZip"
placeholder="34000">
        </div>
    </div>
    <div class="form-group">
        <label for="example-tel-input" class="col-2 col-form-
label">Telephone</label>
        <div class="col-10">
            <input class="form-control" name="tel" type="tel" value="1-(555)-
555-5555" id="example-tel-input">
        </div>
    </div>
    <div class="form-group">
        <label for="example-date-input" class="col-3 col-form-label">Start
Date</label>
        <div class="col-10">
            <input class="form-control" name="starting_date" type="date"
value="2020-02-01" id="example-date-input">
        </div>
    </div>
    <div class="form-group mt-3">
        <label class="mr-4">Upload your CV:</label>
        <input type="file" name="file">
    </div>
```

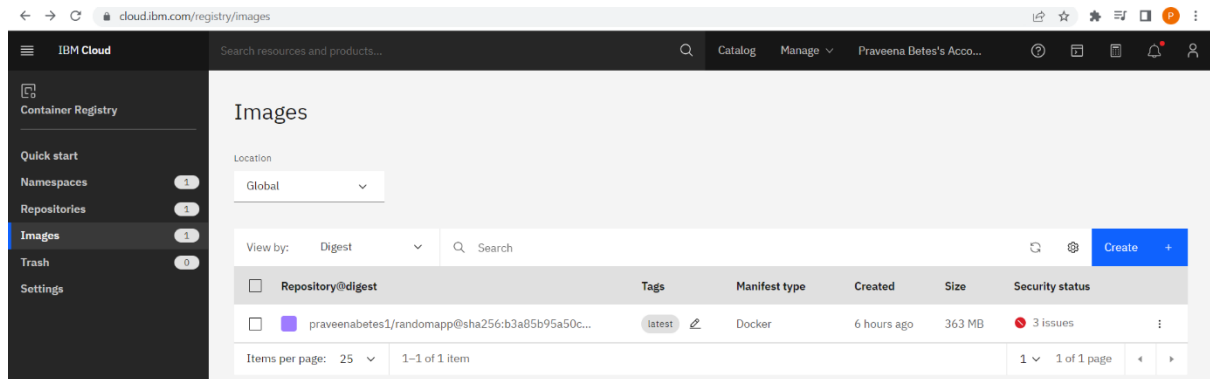
```

    <button type="submit" class="btn btn-primary">Submit</button>
  </form>
</div>
</body>

</html>

```

3. Create a ibm container registry and deploy helloworld app or jobportal app.



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

