

Assignment-4

PLASMA DONOR APPLICATION

Team ID : PNT2022TMID16512

Questions:

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 docker file for the jobportal application and deploy it in Docker desktop application.
4. Create a Kubernetes cluster in IBM cloud and deploy hello world image or job portal image and also expose the same app to run in node port.

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

```
[node1] (local) root@192.168.0.8 ~
$ docker run -dp 80:80 docker/getting-started:pwd
Unable to find image 'docker/getting-started:pwd' locally
pwd: Pulling from docker/getting-started
89d9c30c1d48: Pull complete
24f1c4f0b2f4: Pull complete
16542569a10d: Pull complete
08396939143d: Pull complete
Digest: sha256:9156d395e7e41498d5348e95513d61fc7929db720393448306c5d7263d7f2696
Status: Downloaded newer image for docker/getting-started:pwd
afe9c7b8d51769ca33d7c846644aad1ceefa4372fde68b8c18f4b27503f0a756
[node1] (local) root@192.168.0.8 ~
$ sudo docker images
REPOSITORY          TAG             IMAGE ID        CREATED         SIZE
world                latest          45328bdf05eb   13 months ago  25.8MB
[node1] (local) root@192.168.0.8 ~
```

File Edit Selection View Go Run Terminal Help

EXPLORER

- IBM
 - Assignment
 - JOB PORTAL
 - static
 - templates
 - app.py 2, U
 - CAD
 - deployment.yaml
 - DigiCertGlobalRootCA...
 - Dockerfile
 - README.md
 - requirements.txt
 - sendemail.py
 - service.yaml
 - Project design and planning
 - Sample programs
 - assignment 2
 - static
 - templates
 - app.py
 - DigiCertGlobalRootCA.crt

OUTLINE

TIMELINE

requirements.txt

app.py 2, U

JOB PORTAL > app.py > apply

```
82
83     return render_template('dashboard.html')
84
85 @app.route('/apply', methods = ['GET', 'POST'])
86 def apply():
87     msg = ''
88     if request.method == 'POST' :
89         username = request.form['username']
90         email = request.form['email']
```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

sudo - JOB PORTAL

Step 1/8 : FROM python:3.6
--> 54260638d07c
Step 2/8 : WORKDIR /app
--> Using cache
--> 993215fe524e
Step 3/8 : ADD . /app
--> 4351b5c29fdb
Step 4/8 : COPY requirements.txt /app
--> 45acc8d4f27f
Step 5/8 : RUN python3 -m pip install -r requirements.txt
--> Running in 8e223a861382

Ln 97, Col 44 Spaces: 4 UTF-8 CRLF Python 3.8.10 64-bit Go Live

File Edit Selection View Go Run Terminal Help

EXPLORER

- IBM
 - Assignment
 - JOB PORTAL
 - static
 - templates
 - app.py 2, U
 - CAD
 - deployment.yaml
 - DigiCertGlobalRootCA...
 - Dockerfile
 - README.md
 - requirements.txt
 - sendemail.py
 - service.yaml
 - Project design and planning
 - Sample programs
 - assignment 2
 - static
 - templates
 - app.py
 - DigiCertGlobalRootCA.crt

OUTLINE

TIMELINE

requirements.txt

app.py 2, U

JOB PORTAL > app.py > apply

```
82
83     return render_template('dashboard.html')
84
85 @app.route('/apply', methods = ['GET', 'POST'])
86 def apply():
87     msg = ''
88     if request.method == 'POST' :
89         username = request.form['username']
90         email = request.form['email']
```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

bash - JOB PORTAL

Step 8/8 : CMD ["python", "app.py"]
--> Running in e76a612bbca1
Removing intermediate container e76a612bbca1
--> 8b022ea43a31
Successfully built 8b022ea43a31

Ln 97, Col 44 Spaces: 4 UTF-8 CRLF Python 3.8.10 64-bit Go Live

Aboutus

Mission

SMARTBRIDGE is an edTech organization with a vision to bridge the gap between academia & industry. Our outcome-based experiential learning programs on emerging technologies (Internet of Things, Machine Learning, Data Science, Artificial Intelligence, Robotics) are building skilled entry - level engineers, for the corporate world. .

Vission

Our main objective is to bridge the existing gaps between prevailing industry standards and what the academics offer to the graduates while passing out of university. SmartBridge offers suitable skill deployment and training to the young talent before on boarding their first job. Our skill development programs are designed considering the present expectations in the industry.

Objective

Well directed career guidance programs for educational institutions
Appropriate certification courses that suit the industry need
Train the trainers; expanded awareness about the current industry standards
Liaise with corporates to offer niche internships
Establish technology development centers in colleges
Specialised incubation centers in collaboration with corporates

JobPortal

Lorem ipsum dolor sit amet consectetur adipisicing elit. Voluptatum quis, reiciendis id magni magnam, accusamus nobis in, temporibus molestias ab placeat rerum aperiam illum perspiciatis ducimus non! Fugiat, odit ducimus.

Get in Touch

- jobportal@gmail.com
- +91 8977787657

The screenshot displays a web browser window at the top, showing the 'Aboutus' page of a website. The page has a dark header with navigation links: 'LOGIN', 'REGISTER', and 'CONTACT US'. The main content area is divided into three columns: 'Mission', 'Vission', and 'Objective'. The 'Mission' column describes SMARTBRIDGE as an edTech organization. The 'Vission' column states the main objective is to bridge gaps between industry standards and academia. The 'Objective' column lists several goals, including career guidance, certification courses, training trainers, awareness expansion, niche internships, technology development centers, and specialized incubation centers. Below these columns are sections for 'JobPortal' and 'Get in Touch'. The 'JobPortal' section contains a placeholder text. The 'Get in Touch' section provides contact information: an email address 'jobportal@gmail.com' and a phone number '+91 8977787657'.

Below the browser window is a code editor showing the source code for the 'JobPortal' application. The code is written in Python using the Flask framework. It defines a Flask app, sets the template path, and defines a route for '/apply'. The route handles both GET and POST requests. For GET requests, it renders the 'dashboard.html' template. For POST requests, it takes 'username' and 'email' as input and prints them. The code is as follows:

```

1  from flask import Flask
2  app = Flask(__name__)
3  app.config['TEMPLATES_AUTO_RELOAD'] = True
4
5  @app.route('/')
6  def index():
7      return render_template('dashboard.html')
8
9  @app.route('/apply', methods=['GET', 'POST'])
10 def apply():
11     msg = ''
12     if request.method == 'POST':
13         username = request.form['username']
14         email = request.form['email']
15
16     return render_template('dashboard.html', msg=msg)

```

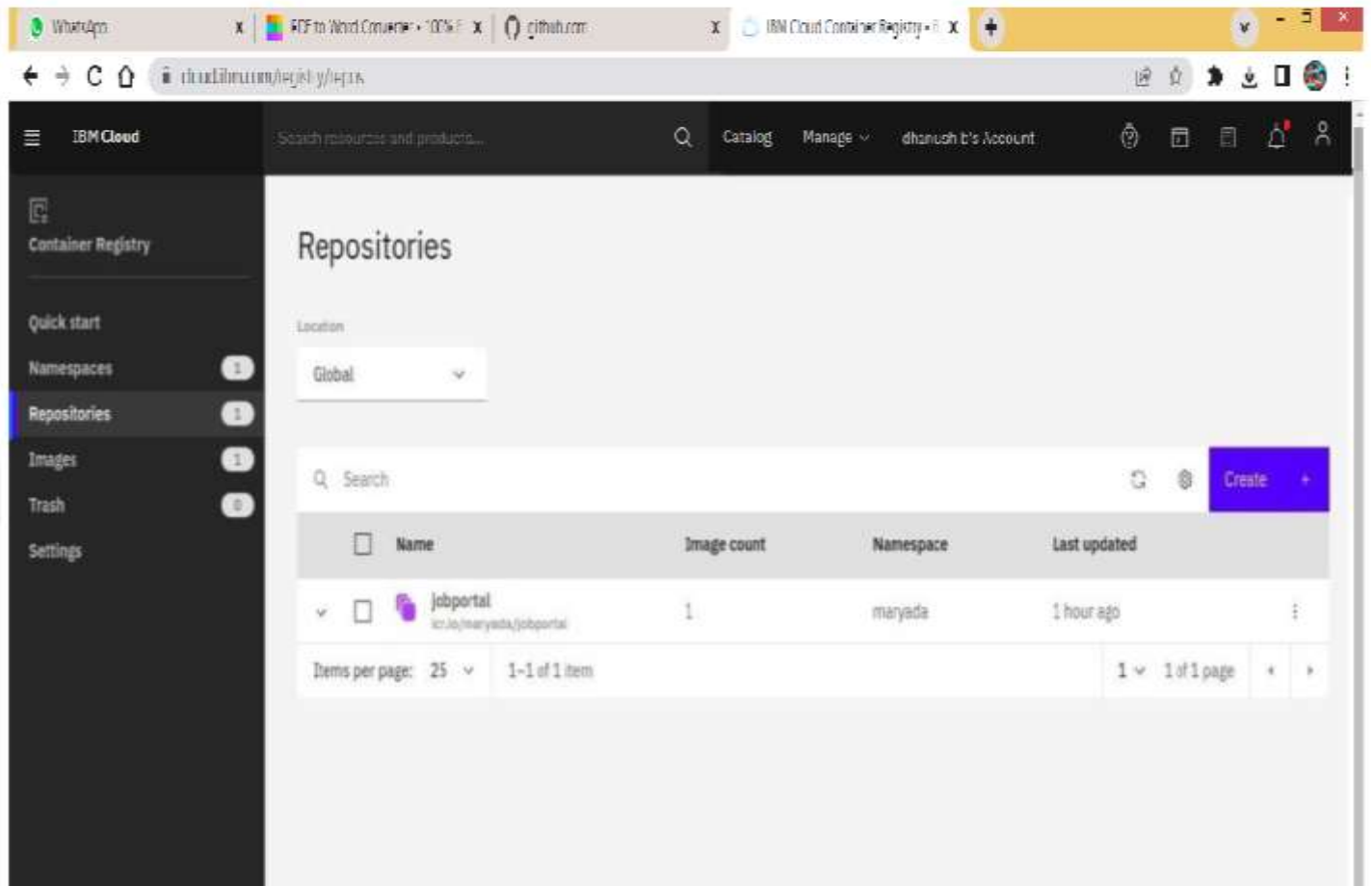
The code editor also shows a terminal window at the bottom, displaying the output of the application. It shows the Flask app running on http://172.17.0.2:5000/ and serving the 'app' (lazy loading). It also shows the environment (production) and debug mode (off). The terminal output includes several log messages indicating successful requests to the application:

```

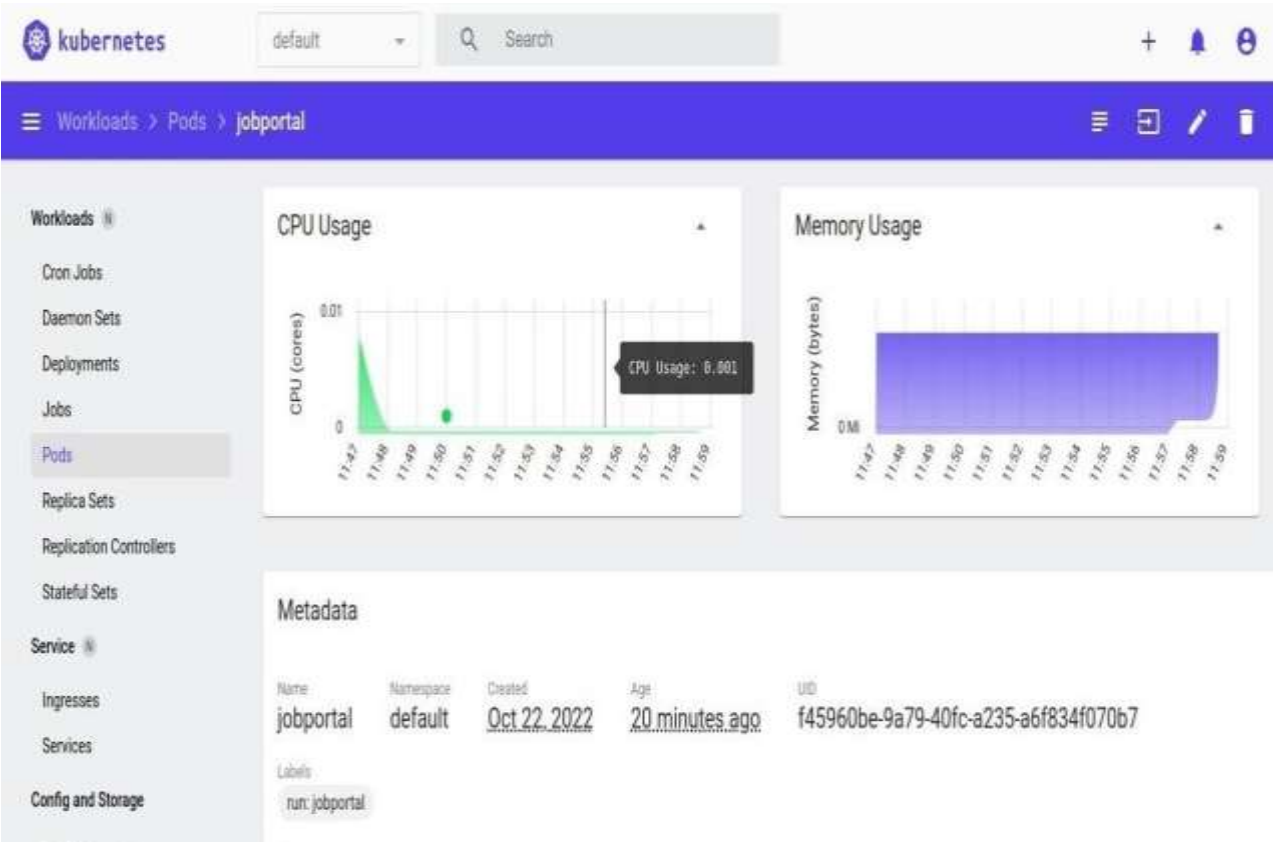
* Serving Flask app 'app' (lazy loading)
* Environment: production
WARNING: This is a development server. Do not use it in a production deployment.
Use a production WSGI server instead.
* Debug mode: off
* Running on all addresses.
WARNING: This is a development server. Do not use it in a production deployment.
* Running on http://172.17.0.2:5000/ (Press CTRL+C to quit)
172.17.0.1 - - [22/Oct/2022 05:06:38] "GET / HTTP/1.1" 200 -
172.17.0.1 - - [22/Oct/2022 05:06:38] "GET /css/style.css HTTP/1.1" 404 -
172.17.0.1 - - [22/Oct/2022 05:06:38] "GET /static/img/smartinternz.png HTTP/1.1" 404 -
172.17.0.1 - - [22/Oct/2022 05:06:38] "GET /assets/img/favicon-32x32.png HTTP/1.1" 404 -

```

3. Create a IBM container registry and deploy hello world app or job portal app.



4. Create a Kubernetes cluster in IBM cloud and deploy hello world image or job portal image and also expose the same app to run in node port.



Workloads > Pods

Workloads

- Cron Jobs
- Daemon Sets
- Deployments
- Jobs
- Pods**
- Replica Sets
- Replication Controllers
- Stateful Sets

Service

- Ingresses
- Services

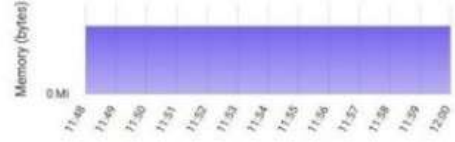
Config and Storage

- Config Maps

CPU Usage



Memory Usage



Pods

Name	Images	Labels	Node	Status	Restarts	CPU Usage (cores)
jobportal	Show all	Show all	10.144.216.52	Running	0	1.00m
lb4-simple-web-app-deployment	Show all	Show all	10.144.216.52	ImagePullBackOff	0	-