

ASSIGNMENT 4

Kubernetes:

i) dashboard admin.yaml code:

apiVersion : v1

kind: ServiceAccount

metadata:

name: admin-user

namespace: kubernetes-dashboard

apiVersion: v1

kind: Secret

metadata:

name: admin-user-token

namespace: kubernetes-dashboard

annotations:

kubernetes.io/service-account.name: admin-user

type: kubernetes.io/service-account-token

apiVersion: rbac.authorization.k8s.io/v1

kind: ClusterRoleBinding

metadata:

name: admin-user

roleRef:

apiGroup: rbac.authorization.k8s.io

kind: ClusterRole

name: cluster-admin

subjects:

- kind: ServiceAccount

name: admin-user

namespace: Kubernetes - dashboard

ii) [flask_deployment .yaml code](#):

apiversion : apps/v1

kind: Deployment

metadata:

name: flask-app

spec:

replicas: 3

selector:

matchLabels:

app: flask-app

template:

metadata:

labels:

app: flask-app

spec:

containers:

- name: flask-app-container

image: project3:latest

imagePullPolicy: Never

ports:

- containerPort: 5000

Protocol:tcp

iii)[flask_ingress.yaml code](#):

apiversion : networking.k8s.io/v1

kind: Ingress

metadata:

name: flask-app-ingress

annotations:

kubernetes.io/ingress.class: nginx

nginx.ingress.kubernetes.io/ssl-redirect: "false"

spec:

ingressClassName: nginx

rules:

- http:

paths:

- backend:

service:

name: flask-app-service

port:

number: 5000

path: /

pathType: Prefix

iv) [flask_service](#) :

apiVersion: v1

kind: Service

metadata:

name: flask-app-service

spec:

type: ClusterIP

ports:

- port: 5000

selector:

app:flask-app

v)ibm deploy code:

apiversion: apps/v1

kind: Deployment

metadata:

name: flask-app

spec:

replicas: 3

selector:

matchLabels:

app: flask-app

template:

metadata:

labels:

app: flask-app

spec:

containers:

- name: flask-app-container
- image: jp.icr.io/ibmtrainning/flask-app-testing
- imagePullPolicy: Always
- ports:
 - containerPort: 5000

Protocol :tcp

Templates:

job. html code:

```
<!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>JOB PORTAL</title>

</head>
<style>
form {text-align: center;}
<p style = "background-image: url('jp.jpg');">
```

</style>

<body style="background-color:#dcf3f5">

<div class="header">

<h1>JOB PORTAL</h1>

<p>We're here to land you to your dream job!</p>

</div>

<form>

<h2>Enter Your Credentials</h2>

<label for="fname">First name:</label>

<input type="text" id="fname"
name="fname">

<label for="lname">Last name:</label>

<input type="text" id="lname"
name="lname">

<label for="email">Email i'd:</label>

<input type="text" id="email"
name="email">

<label for="phone">Phone Number:</label>


```
<input type="text" id="phone"
name="phone"><br><br>
```

```
<label for="quali">Educational
Qualification:</label><br>
```

```
<input type="text" id="quali"
name="quali"><br><br>
```

```
<label for="Percentage">Percentage:</label><br>
```

```
<input type="text" id="Percentage"
name="Percentage">
```

```
<p>Choose your Web language:</p>
```

```
<input type="radio" id="html"
name="web_language" value="HTML">
```

```
<label for="html">HTML</label><br>
```

```
<input type="radio" id="css" name="web_language"
value="CSS">
```

```
<label for="css">CSS</label><br>
```

```
<input type="radio" id="javascript"
name="web_languageweb_language"
value="JavaScript">
```



```
<label for="javascript">JavaScript</label>
<p>How many language do you know?</p>
<input type="checkbox" id="vehicle1"
name="vehicle1" value="Bike">
<label for="vehicle1"> 1</label><br>
<input type="checkbox" id="vehicle2"
name="vehicle2" value="Car">
<label for="vehicle2"> 2</label><br>
<input type="checkbox" id="vehicle3"
name="vehicle3" value="Boat">
<label for="vehicle3"> morethan 2</label>

<br><br><br><input type="submit"
value="Submit">
</form>

</body>
</html>
```

App.py code:

```
from flask import Flask, render_template
```

```
app = Flask(__name__)  
@app.route('/')  
def job():  
    return render_template('job.html')  
if __name__ == '__main__':  
app.run(debug=True,host ='0.0.0.0')
```

[Docker file :](#)

```
FROM python:3.10.7-alpine  
WORKDIR /Assignment4  
ADD . /Assignment4  
RUN pip install -r requirements.txt  
CMD ["python","app.py"]
```

[Requirements:](#)

Flask

Output:

JOB PORTAL

We're here to land you to your dream job!

Enter Your Credentials

First name:

Last name:

Email id:

Phone Number:

Educational Qualification:

Percentage:

Choose your Web language:

☐ HTML
☐ CSS
☐ JavaScript

How many language do you know?

☐ 1
☐ 2
☐ morethan 2