

## **ASSIGNMENT-4**

**PROJECT NAME :NEWS TRACKER APPLICATION**

**KUBERNETES**

### **Deployment.html**

```
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: webpage  
      image: Assignment/flask  
      imagePullPolicy: Never  
      ports:  
        - containerPort: 5000  
          protocol: TCP
```

### **service.yaml**

```
apiVersion: v1  
kind: Service  
metadata:  
  name: flask-app-service  
spec:  
  type: ClusterIP  
  ports:  
    - port: 5000  
  selector:  
    app: flask-app
```

### **base.html**

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <meta charset="UTF-8">  
  <title> FlaskApp</title>
```

```
<style>

.message {
    padding: 10px;
    margin: 5px;
    background-color: #656362
}

h1 {
    color: #1d78cd;
}

nav a {
    color: #1d78cd;
    font-size: 25px;
    margin-left: 50px;
    text-decoration: none;
    text-transform: uppercase;
}

.alert {
    padding: 20px;
    margin: 5px;
    color: #977c30;
    background-color: #ddaa1ae;
}

.button {
background-color:#977c30; /* red */
border: none;
color: white;
padding: 16px 32px;
text-align: center;
```

```
text-decoration: none;  
display: inline-block;  
font-size: 16px;  
margin: 4px 2px;  
transition-duration: 0.4s;  
cursor: pointer;  
}
```

```
.button1 {  
background-color: white;  
color: black;  
border-radius: 10px;  
border: 2px solid #036523;  
}
```

```
.button1:hover {  
background-color: #00971e;  
color: rgb(47, 41, 51);  
}  
input[type=text],textarea {  
width: 50%;  
padding: 12px 20px;  
margin: 8px 0;  
box-sizing: border-box;  
border: 2px solid;  
border-radius: 4px;  
}
```

```
</style>
```

```
</head>
```

```
<body>
  <nav>
    <a href="#">Home</a>
    <a href="{{ url_for('index') }}">FlaskApp</a>
    <a href="{{ url_for('create') }}">Create</a>

  </nav>
  <hr>
  <div class="content">
    { % for message in get_flashed_messages() % }
      <div class="alert">{{ message }}</div>
    { % endfor %}
    { % block content % } { % endblock %}
  </div>

</body>
</html>
```

### **create.html**

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <title> FlaskApp</title>
    <style>
      .message {
        padding: 10px;
        margin: 5px;
        background-color: #656362
```

```
}

h1 {
    color: #1d78cd;

}

nav a {
    color: #1d78cd;
    font-size: 25px;
    margin-left: 50px;
    text-decoration: none;
    text-transform: uppercase;
}
```

```
.alert {
    padding: 20px;
    margin: 5px;
    color: #977c30;
    background-color: #ddaa1ae;
}

.button {
background-color:#977c30; /* red */
border: none;
color: white;
padding: 16px 32px;
text-align: center;
text-decoration: none;
display: inline-block;
font-size: 16px;
margin: 4px 2px;
transition-duration: 0.4s;
```

```
        cursor: pointer;
    }

.button1 {
    background-color: white;
    color: black;
    border-radius: 10px;
    border: 2px solid #036523;
}

.button1:hover {
    background-color: #00971e;
    color: rgb(47, 41, 51);
}

input[type=text],textarea {
    width: 50%;
    padding: 12px 20px;
    margin: 8px 0;
    box-sizing: border-box;
    border: 2px solid;
    border-radius: 4px;
}

</style>
</head>
<body>
<nav>
    <a href="#">Home</a>
    <a href="{{ url_for('index') }}">FlaskApp</a>
    <a href="{{ url_for('create') }}">Create</a>
```

```

</nav>
<hr>
<div class="content">
    {% for message in get_flashed_messages() %}
        <div class="alert">{{ message }}</div>
    {% endfor %}
    {% block content %} {% endblock %}
</div>

</body>
</html>

```



## index.html

```

{% extends 'base.html' %}

{% block content %}
    <h1 style="text-align: center;">WELCOME!</h1>
    <h1> {% block title %} <span style="color: rgb(29, 11, 166);">WRITE
    YOUR MESSAGES</span> {% endblock %}</h1>
    {% for message in messages %}

```

```
<div class='message'>
    <h3><span style="color: rgb(255, 255, 255);">TITLE : </span>
    { { message['title'] } }</h3>
    <p><span style="color: rgb(255, 255, 255);">MESSAGE :
    </span>{ { message['content'] } }</p>
</div>
{ % endfor %}
{ % endblock %}
```

```
PS C:\Windows\system32> docker pull hello-world
Using default tag: latest
latest: Pulling from library/hello-world
Digest: sha256:e18f0a777aefabe047a671ab3ec3eed05414477c951ab1a6f352a06974245fe7
Status: Image is up to date for hello-world:latest
docker.io/library/hello-world:latest
PS C:\Windows\system32>
```

## Dockerfile

```
FROM python:3.10-buster
WORKDIR /app
COPY ..
RUN pip install --no-cache-dir -r requirements.txt
CMD ["ibm_db2", "--bind", "0.0.0.0:5000", "app:app"]
```

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

16. Welcome to Project Delightful - IBM Cloud - mycluster-free - IBM Cloud

mycluster-free - IBM Cloud

IBM Cloud Search resources and products Catalog Manage + Project Delightful Account

Cluster / mycluster-free Preparing master, workers... Deploy in 10 steps Add tags Actions...

Overview Worker nodes Worker pools DevOps Help

Worker nodes

| Name     | Status            | Worker pool | Zone | Private IP | Public IP   | Version |
|----------|-------------------|-------------|------|------------|-------------|---------|
| 00000000 | Pending (pending) | default     | None | 0.0.0.0    | 124.6.154.1 | 1.0.0   |

Items per page: 25 1-1 of 1 items 1 of 1 page

Help Log in to your cluster Deploy your app Expose your app Add storage to your app Connect integrations Install add-ons Troubleshoot

25°C Cloudy

The screenshot shows the IBM Cloud Kubernetes dashboard for a cluster named 'mycluster-free'. The 'Worker nodes' section displays one node: '00000000' (Status: Pending). The dashboard also includes sections for Overview, Worker pools, and DevOps. A sidebar on the right provides links for help and various cluster management tasks.