Kubernetes Project 3 - DEPLOY A MULTI-TIER WEB APPLICATION ON KUBERNETES

Step 1: First check the minikube is there or not, if not there install

Step 2: Check docker install and node is connected or not

Step 3: Make directory and add all the files with code

Step 4: Build the docker image and push to the docker hub


```
master@master-vm:~/multi-tier-application/flask$ docker build -t juhichoudhary/flaskapp
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
            Install the buildx component to build images with BuildKit:
            https://docs.docker.com/go/buildx/

"docker build" requires exactly 1 argument.
See 'docker build --help'.

Usage:  docker build [OPTIONS] PATH | URL | -

Build an image from a Dockerfile
master@master-vm:~/multi-tier-application/flask$ docker push juhichoudhary/flaskapp
Using default tag: latest
The push refers to repository [docker.io/juhichoudhary/flaskapp]
An image does not exist locally with the tag: juhichoudhary/flaskapp
master@master-vm:~/multi-tier-application/flask$ docker build -t juhichoudhary/flaskapp .
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
            Install the buildx component to build images with BuildKit:
            https://docs.docker.com/go/buildx/

Sending build context to Docker daemon  8.192kB
Step 1/6 : FROM python:3.8
3.8: Pulling from library/python
cdd62bf39133: Pull complete
a47cff7f31e9: Pull complete
a173f2aee8e9: Pull complete
01272fe8adba: Extracting [=================================================>  ]  210.6MB/211.3MB
01272fe8adba: Pull complete
cddc73e4e6c7: Pull complete
cc48f13b5f0f: Pull complete
5a98c896c047: Pull complete
Digest: sha256:d411270700143fa2683cc8264d9fa5d3279fd3b6afff62ae81ea2f9d070e390c
```

Step 5: login to the Docker and push the image


```
Successfully built 3439a3e04d5e
Successfully tagged juhichoudhary/flaskapp:latest
master@master-vm:~/multi-tier-application/flask$ docker login
Authenticating with existing credentials...
WARNING! Your password will be stored unencrypted in /home/master/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
master@master-vm:~/multi-tier-application/flask$ docker push juhichoudhary/flaskapp
Using default tag: latest
The push refers to repository [docker.io/juhichoudhary/flaskapp]
b005f8114e16: Pushing [>                                                  ]  1.083MB/73.82MB
3c41ae1085d1: Pushing [==================================================>]  2.56kB
5245d34565af: Pushing [==================================================>]  4.096kB
a4bcd1c9ec6a: Pushing  1.536kB
32ee710ca3c7: Preparing
1767e4d52b5a: Waiting
45b98afd69b3: Waiting
b005f8114e16: Pushed
3c41ae1085d1: Pushed
5245d34565af: Pushed
a4bcd1c9ec6a: Pushed
32ee710ca3c7: Mounted from library/python
1767e4d52b5a: Mounted from library/python
45b98afd69b3: Mounted from library/python
2bce433c3a29: Mounted from library/python
f91dc7a486d9: Mounted from library/python
3e14a6961052: Mounted from library/python
d50132f2fe78: Mounted from library/python
latest: digest: sha256:eb87883880e4c8522d4d1a94aad94cf906212b50be6aeb000781bffcf6c03c78 size: 2628
```

Step 6: Apply the configuration

```
master@master-vm:~/multi-tier-application/nginx$ kubectl apply -f ~/multi-tier-application/flask/flask-deployment.yaml
deployment.apps/flask-app configured
master@master-vm:~/multi-tier-application/nginx$ kubectl apply -f ~/multi-tier-application/flask/flask-service.yaml
service/flask-service unchanged
master@master-vm:~/multi-tier-application/nginx$ kubectl apply -f ~/multi-tier-application/mysql/mysql-deployment.yaml
deployment.apps/mysql unchanged
service/mysql unchanged
master@master-vm:~/multi-tier-application/nginx$ kubectl apply -f ~/multi-tier-application/mysql/mysql-pv.yaml
persistentvolume/mysql-pv unchanged
persistentvolumeclaim/mysql-pvc unchanged
master@master-vm:~/multi-tier-application/nginx$ kubectl apply -f ~/multi-tier-application/mysql/mysql-secret.yaml
secret/mysql-secret unchanged
master@master-vm:~/multi-tier-application/nginx$ kubectl apply -f ~/multi-tier-application/nginx/nginx-configmap.yaml
configmap/nginx-config unchanged
master@master-vm:~/multi-tier-application/nginx$ kubectl apply -f ~/multi-tier-application/nginx/nginx-deployment.yaml
deployment.apps/nginx unchanged
master@master-vm:~/multi-tier-application/nginx$ kubectl apply -f ~/multi-tier-application/nginx/nginx-service.yaml
service/nginx-service unchanged
```

Step 7: get all pods

```
master@master-vm:~/multi-tier-application/nginx$ kubectl get pods -o wide
NAME                        READY   STATUS    RESTARTS   AGE   IP            NODE       NOMINATED NODE   READINESS GATES
flask-app-5b4dfc7c78-27mhf  1/1     Running   0          15h   10.244.0.19   minikube   <none>           <none>
flask-app-5b4dfc7c78-74fm4  1/1     Running   0          15h   10.244.0.17   minikube   <none>           <none>
flask-app-5b4dfc7c78-9scrk  1/1     Running   0          15h   10.244.0.16   minikube   <none>           <none>
flask-app-5b4dfc7c78-qdzkm  1/1     Running   0          15h   10.244.0.18   minikube   <none>           <none>
mysql-5f797c4f79-rzxrs      1/1     Running   0          15h   10.244.0.20   minikube   <none>           <none>
nginx-766c76446-6pf2g       1/1     Running   0          34s   10.244.0.22   minikube   <none>           <none>
master@master-vm:~/multi-tier-application/nginx$ minikube ip
192.168.49.2
master@master-vm:~/multi-tier-application/nginx$ kubectl get services nginx-service
NAME            TYPE       CLUSTER-IP     EXTERNAL-IP   PORT(S)        AGE
nginx-service   NodePort   10.97.49.102   <none>        80:30007/TCP   15h
```

Step 8: Database created and outputs

```
master@master-vm:~/multi-tier-application/mysql$ kubectl exec -it mysql-5f797c4f79-rzxrs -- mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 14
Server version: 5.7.44 MySQL Community Server (GPL)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE DATABASE mydb;
ERROR 1007 (HY000): Can't create database 'mydb'; database exists
mysql> USE mydb;
Database changed
mysql> SHOW TABLES;
Empty set (0.01 sec)

mysql> CREATE TABLE users (
    ->     id INT AUTO_INCREMENT PRIMARY KEY,
    ->     name VARCHAR(100),
    ->     email VARCHAR(100)
    -> );
Query OK, 0 rows affected (0.16 sec)

mysql> INSERT INTO users (name, email) VALUES ('Alice', 'alice@example.com');
Query OK, 1 row affected (0.05 sec)

mysql> INSERT INTO users (name, email) VALUES ('Bob', 'bob@example.com');
Query OK, 1 row affected (0.03 sec)

mysql> SELECT * FROM users;
+----+-------+-------------------+
| id | name  | email             |
+----+-------+-------------------+
|  1 | Alice | alice@example.com |
|  2 | Bob   | bob@example.com   |
+----+-------+-------------------+
```
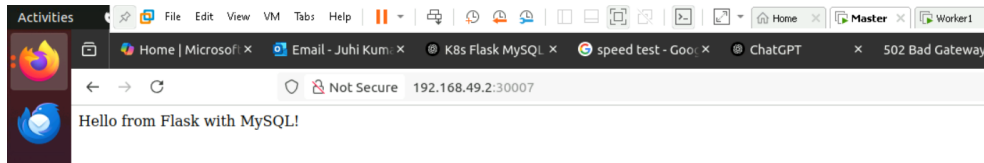
Step 9: open browser and run ip:30007

Hello from Flask with MySQL!

Step10: Database output table created