CDEC B24

Name – Vaibhav Navneet Jorvekar

Hosting three-tier student app via docker images

Step 1. Create Database in MYSQL for this use command

-docker run -d -p 306:3306 -e MYSQL_ROOT_PASSWORD=1234 mysql:latest

-docker ps

```
Last login: Tue Mar 19 08:47:11 2024 from 13.52.6.115
ubuntu@ip-172-31-26-198:~$ sudo -i
root@ip-172-31-26-198:~# docker run -d -p 306:3306 -e MYSQL_ROOT_PASSWORD=1234 mysql:latest
Unable to find image 'mysql:latest' locally
latest: Pulling from library/mysql
9a5c778f631f: Pull complete
9e77c3a95bf2: Pull complete
8b279a2086e0: Pull complete
c8bfbcde7882: Pull complete
d35b074b68ec: Pull complete
beea5014e6af: Pull complete
dc3791a61558: Pull complete
52f9323b9f0e: Pull complete
7f7391eab49b: Pull complete
8d2f04b287ee: Pull complete
Digest: sha256:9d1c923e5f66a89607285ee2641f8a53430a1ccd5e4a62b35eb8a48b74b9ff48
Status: Downloaded newer image for mysql:latest
0e24fe2c75ab07a3d85a01c0153884ed06bffc8c47f5d2f01e68a11fbd2116a1
root@ip-172-31-26-198:~# docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED          STATUS          PORTS                                                  NA
MES
0e24fe2c75ab   mysql:latest   "docker-entrypoint.s…"   12 seconds ago   Up 11 seconds   33060/tcp, 0.0.0.0:306->3306/tcp, :::306->3306/tcp   pr
actical_hugle
root@ip-172-31-26-198:~#
```

Step 2. After intering this command you enter into the mysql use command to create dadabase.

- docker exec -it <container id>mysql -u root -p1234

- create database studentapp;

-use studentapp;

- CREATE TABLE if not exists students(student_id INT NOT NULL AUTO_INCREMENT,

student_name VARCHAR(100) NOT NULL,

student_addr VARCHAR(100) NOT NULL,

student_age VARCHAR(3) NOT NULL,

student_qual VARCHAR(20) NOT NULL,

student_percent VARCHAR(10) NOT NULL,

student_year_passed VARCHAR(10) NOT NULL,

PRIMARY KEY (student_id)

);

Show the database using this command

-desc students;

```
mysql> desc students;
+--------------------+--------------+------+-----+---------+----------------+
| Field              | Type         | Null | Key | Default | Extra          |
+--------------------+--------------+------+-----+---------+----------------+
| student_id         | int          | NO   | PRI | NULL    | auto_increment |
| student_name       | varchar(100) | NO   |     | NULL    |                |
| student_addr       | varchar(100) | NO   |     | NULL    |                |
| student_age        | varchar(3)   | NO   |     | NULL    |                |
| student_qual       | varchar(20)  | NO   |     | NULL    |                |
| student_percent    | varchar(10)  | NO   |     | NULL    |                |
| student_year_passed | varchar(10) | NO   |     | NULL    |                |
+--------------------+--------------+------+-----+---------+----------------+
7 rows in set (0.00 sec)
```

and exit the mysql using command

-exit

And run this command to see image

-docker inspect <container id> | grep “IP”

```
root@ip-172-31-26-198:~# docker inspect 0e2 | grep "IP"
            "LinkLocalIPv6Address": "",
            "LinkLocalIPv6PrefixLen": 0,
            "SecondaryIPAddresses": null,
            "SecondaryIPv6Addresses": null,
            "GlobalIPv6Address": "",
            "GlobalIPv6PrefixLen": 0,
            "IPAddress": "172.17.0.2",
            "IPPrefixLen": 16,
            "IPv6Gateway": "",
                    "IPAMConfig": null,
                    "IPAddress": "172.17.0.2",
                    "IPPrefixLen": 16,
                    "IPv6Gateway": "",
                    "GlobalIPv6Address": "",
                    "GlobalIPv6PrefixLen": 0,
root@ip-172-31-26-198:~#
```

- Backend
  1. Create repo in github use existing and make two folder un the repo
     a) Frontend
     b) Backend

  2. In backend create 3 files.
     a) Dockerfile – your image
     b) Context.xml – add mysql ip
     c) Student.war

  3. Make git clone and build docker image.

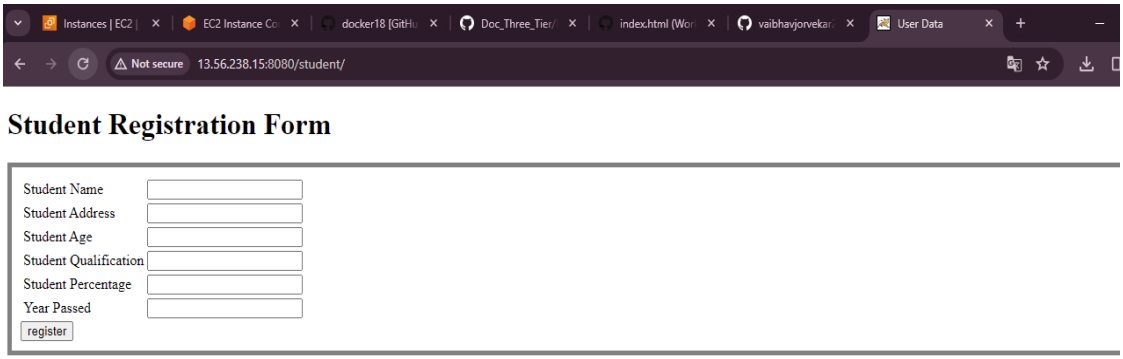
     -docker build .

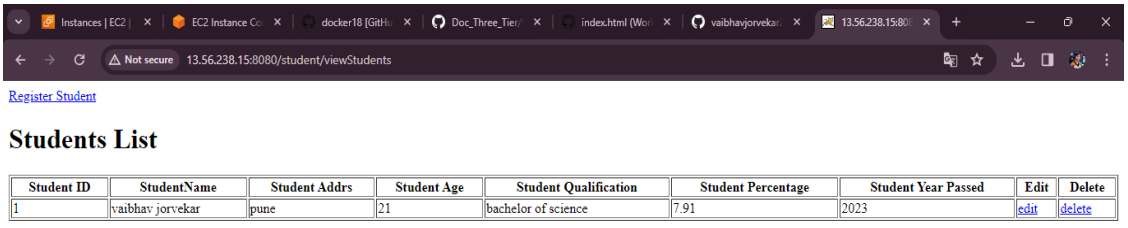
     -docker ps

     -docker images

     -docker run -d -p 8080:8080 <image id>

     Hit ip with 8080 port number and /student





- Frontend
  1. Create 2 files in frontend folder
     a) Dockerfile – your image
     b) Index.html – pest your instance ip

  2. Git push or pull after use commands
     -docker build .
     -docker ps
     -docker run -d -p 80:80 <image id>
     Hit ip with 80 port number

Welcome to Student Application on AWS.

Enter to Student Application