Hosting Tomcat Website in a Docker Container Manually

- First we need an instance.
- Allow All TCP and MySQL in the security group.
- Install docker using command "yum install docker -y".
- Now we will first create a database.
- For database we need MySQL in a container.
- We will use a MySQL image to create a container.
- Hit command "**docker run -d -p 3306:3306 -e MYSQL_ROOT_PASSWORD="1234 mysql**" to pull and run the mysql image.
- The container will be created after running the image.
- Hit command "**docker ps**" to view all the running containers.
- Hit command "**docker exec -it container_name /bin/bash**" to enter in the container.

```
[root@ip-172-31-23-34 ec2-user]# docker run -d -p 3306:3306 -e MYSQL_ROOT_PASSWORD="1234" mysql
Unable to find image 'mysql:latest' locally
latest: Pulling from library/mysql
2ba873cb070a: Pull complete
dd1a4da808dd: Pull complete
3292fb4adf41: Pull complete
3811c45068cc: Pull complete
e13320244c05: Pull complete
6a34d702f281: Pull complete
de90f4481477: Pull complete
d575200ae375: Pull complete
aea400be5707: Pull complete
38c930606a4f: Pull complete
Digest: sha256:0f2e15fb8b47db2518b1428239ed3e3fe6a6693401b2cf19552063562cfc2fc4
Status: Downloaded newer image for mysql:latest
d524d9f87937adb6aeed068b13f7d3ef7387398ab6e9d64395d008bb8c863e27
[root@ip-172-31-23-34 ec2-user]# docker ps
CONTAINER ID   IMAGE     COMMAND                CREATED        STATUS         PORTS                                                              NAMES
d524d9f87937   mysql     "docker-entrypoint.s…"  6 seconds ago  Up 4 seconds   0.0.0.0:3306->3306/tcp, :::3306->3306/tcp, 33060/tcp   sleepy_diffie
[root@ip-172-31-23-34 ec2-user]# docker exec -it sleepy_diffie /bin/bash
bash-4.4#
```

- Now we need to access the database.
- Hit command "**mysql -u root -p1234**"

```
bash-4.4# mysql -u root -p1234
mysql: [Warning] Using a password on the command line interface can be insecure.
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.3.0 MySQL Community Server - GPL

Copyright (c) 2000, 2024, 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>
```

- Now we need to create a database for our application.
- Hit command "**create database studentapp**".
- Hit command "**use studentapp**".

- Now we need to create a table structure.
- Use the command below to create a table structure.
- **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)
  );

```
mysql> create database studentapp;
Query OK, 1 row affected (0.01 sec)

mysql> use studentapp;
Database changed
mysql> 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)
    -> );
Query OK, 0 rows affected (0.03 sec)
```

- Table created.
- Hit command "**desc students;**" to view the table structure.

```
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)
```

- Now the database is created.
- Exit from mysql and the container.
- Now we need the IP of the mysql container.
- Hit command "**docker inspect container_name**".
- Copy the IP of the mysql container.
- Now we have to host the application in the tomcat.
- First we need **centos 7** image.

- Hit command "**docker run -it -p 8080:8080 centos:7**".
- This command will pull the image if not present locally and run the image.
- Also we will enter the container.
- Now we have entered the container.
- Change the directory to /opt.

```
[root@ip-172-31-23-34 ec2-user]# docker run -it -p 8080:8080 centos:7
Unable to find image 'centos:7' locally
7: Pulling from library/centos
2d473b07cdd5: Pull complete
Digest: sha256:be65f488b7764ad3638f236b7b515b3678369a5124c47b8d32916d6487418ea4
Status: Downloaded newer image for centos:7
[root@50f5ae38b995 /]# ls
anaconda-post.log  bin  dev  etc  home  lib  lib64  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
[root@50f5ae38b995 /]#
```

i-0a73aa02826e0c4f5 (3)

PublicIPs: 54.87.230.133   PrivateIPs: 172.31.23.34

- Now we have to download the tomcat here.
- Hit command "**curl -O https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.87/bin/apache-tomcat-9.0.87.tar.gz**" to download tomcat.

```
[root@50f5ae38b995 /]# cd /opt
[root@50f5ae38b995 opt]# curl -O https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.87/bin/apache-tomcat-9.0.87.tar.gz
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 11.1M  100 11.1M    0     0  23.4M      0 --:--:-- --:--:-- --:--:-- 23.4M
[root@50f5ae38b995 opt]# ls
apache-tomcat-9.0.87.tar.gz
[root@50f5ae38b995 opt]#
```

i-0a73aa02826e0c4f5 (3)

PublicIPs: 54.87.230.133   PrivateIPs: 172.31.23.34

- Extract the file using command "tar -xvf apache-tomcat-9.0.87.tar.gz".
- Now we need our application.
- Hit command "**curl -O https://webapp2-akashapp.s3.amazonaws.com/student.war**" to download the studentapp war file.
- We also need a mysql connector.
- Hit command "**curl -O https://webapp-akash.s3.amazonaws.com/mysql-connector-j-8.3.0.jar**" to download the connector file.
- Now move the student.war file to webapps and mysql-connector to lib.

```
[root@50f5ae38b995 opt]# ls
apache-tomcat-9.0.87  apache-tomcat-9.0.87.tar.gz  mysql-connector-j-8.3.0.jar  student.war
[root@50f5ae38b995 opt]# mv student.war apache-tomcat-9.0.87/webapps/
[root@50f5ae38b995 opt]# ls
apache-tomcat-9.0.87  apache-tomcat-9.0.87.tar.gz  mysql-connector-j-8.3.0.jar
[root@50f5ae38b995 opt]# mv mysql-connector-j-8.3.0.jar apache-tomcat-9.0.87/lib/
[root@50f5ae38b995 opt]# ls apache-tomcat-9.0.87/webapps/
ROOT  docs  examples  host-manager  manager  student.war
[root@50f5ae38b995 opt]# ls apache-tomcat-9.0.87/lib/
annotations-api.jar       catalina.jar      jsp-api.jar                 tomcat-i18n-cs.jar  tomcat-i18n-pt-BR.jar  tomcat-util.jar
catalina-ant.jar          ecj-4.20.jar      mysql-connector-j-8.3.0.jar  tomcat-i18n-de.jar  tomcat-i18n-ru.jar     tomcat-websocket.jar
catalina-ha.jar           el-api.jar        servlet-api.jar             tomcat-i18n-es.jar  tomcat-i18n-zh-CN.jar  websocket-api.jar
catalina-ssi.jar          jasper-el.jar     tomcat-api.jar              tomcat-i18n-fr.jar  tomcat-jdbc.jar
catalina-storeconfig.jar  jasper.jar        tomcat-coyote.jar           tomcat-i18n-ja.jar  tomcat-jni.jar
catalina-tribes.jar       jaspic-api.jar    tomcat-dbcp.jar             tomcat-i18n-ko.jar  tomcat-util-scan.jar
[root@50f5ae38b995 opt]#
```
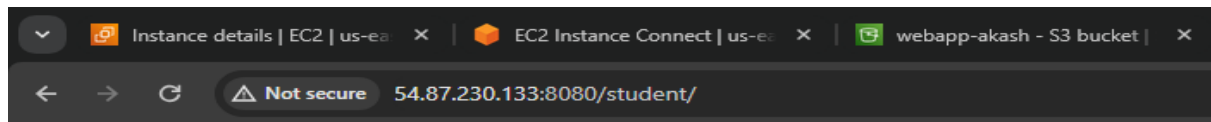
- Now we need a java environment for the tomcat service to run.
- Hit command "**yum install java -y**" to install java.
- Now we need to start the Catalina.sh
- Hit command "**./apache-tomcat-9.0.87/bin/catalina.sh start**" to start the service.

```
[root@50f5ae38b995 opt]# ./apache-tomcat-9.0.87/bin/catalina.sh start
Using CATALINA_BASE:   /opt/apache-tomcat-9.0.87
Using CATALINA_HOME:   /opt/apache-tomcat-9.0.87
Using CATALINA_TMPDIR: /opt/apache-tomcat-9.0.87/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /opt/apache-tomcat-9.0.87/bin/bootstrap.jar:/opt/apache-tomcat-9.0.87/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
[root@50f5ae38b995 opt]#
```

i-0a73aa02826e0c4f5 (3)

PublicIPs: 54.87.230.133   PrivateIPs: 172.31.23.34

- Now hit the IP of your instance and check if the page is visible.
- Our registration page is visible.

| ∨ | 🔲 Instance details \| EC2 \| us-ea ✕ | 🔶 EC2 Instance Connect \| us-ea ✕ | 🟩 webapp-akash - S3 bucket \| ✕ |

← → C ⚠ Not secure 54.87.230.133:8080/student/

Student Registration Form

Student Name	[ ]
Student Address	[ ]
Student Age	[ ]
Student Qualification	[ ]
Student Percentage	[ ]
Year Passed	[ ]

[register]

- Now we have to do the configuration so the data should be saved to database.
- Open the context.xml file in editor.
- The context.xml file is located in apache-tomcat/conf.
- Add the configuration to the context.xml file.
- Give the mysql containe IP address as the endpoint.

```
<!-- The contents of this file will be loaded for each web application -->
<Context>
        <Resource name="jdbc/TestDB" auth="Container" type="javax.sql.DataSource"
                maxTotal="100" maxIdle="30" maxWaitMillis="10000"
                username="root" password="1234" driverClassName="com.mysql.jdbc.Driver"
                url="jdbc:mysql://172.17.0.2:3306/studentapp"/>
```

- Now we need to stop the Catalina.sh service.
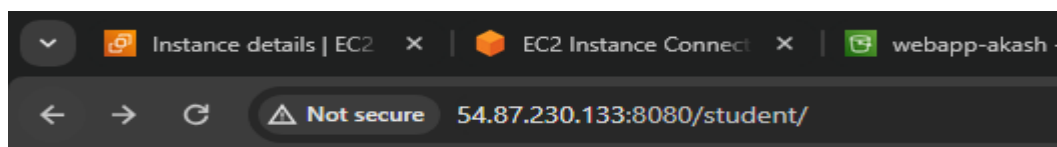- Hit command "**./apache-tomcat-9.0.87/bin/catalina.sh stop**" to stop the service.

- Hit command **"./apache-tomcat-9.0.87/bin/catalina.sh start"** to start the Catalina service.

```
[root@50f5ae38b995 opt]# ./apache-tomcat-9.0.87/bin/catalina.sh stop
Using CATALINA_BASE:   /opt/apache-tomcat-9.0.87
Using CATALINA_HOME:   /opt/apache-tomcat-9.0.87
Using CATALINA_TMPDIR: /opt/apache-tomcat-9.0.87/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /opt/apache-tomcat-9.0.87/bin/bootstrap.jar:/opt/apache-tomcat-9.0.87/bin/tomcat-juli.jar
Using CATALINA_OPTS:
```

```
[root@50f5ae38b995 opt]# ./apache-tomcat-9.0.87/bin/catalina.sh start
Using CATALINA_BASE:   /opt/apache-tomcat-9.0.87
Using CATALINA_HOME:   /opt/apache-tomcat-9.0.87
Using CATALINA_TMPDIR: /opt/apache-tomcat-9.0.87/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /opt/apache-tomcat-9.0.87/bin/bootstrap.jar:/opt/apache-tomcat-9.0.87/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
[root@50f5ae38b995 opt]#
```

i-0a73aa02826e0c4f5 (3)

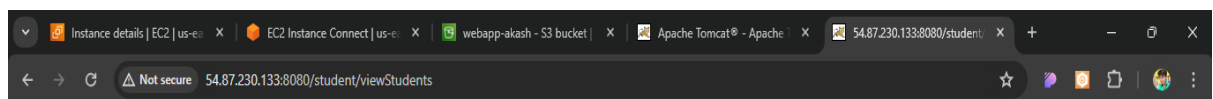PublicIPs: 54.87.230.133   PrivateIPs: 172.31.23.34

- Now hit the IP address.
- Fill the form and click register button.

Instance details | EC2   ×   EC2 Instance Connect   ×   webapp-akash -

← → C   ⚠ Not secure   54.87.230.133:8080/student/

Student Registration Form

Student Name	Akash Shinde
Student Address	PUNE
Student Age	24
Student Qualification	B.E. Mechanical
Student Percentage	88
Year Passed	2022

register

- Data saved successfully in the database.

Instance details | EC2 | us-ea   ×   EC2 Instance Connect | us-ea   ×   webapp-akash - S3 bucket | ×   Apache Tomcat® - Apache   ×   54.87.230.133:8080/student   ×   +

← → C   ⚠ Not secure   54.87.230.133:8080/student/viewStudents

Register Student

Students List

Student ID	StudentName	Student Addrs	Student Age	Student Qualification	Student Percentage	Student Year Passed	Edit	Delete
1	Akash Shinde	PUNE	24	B.E. Mechanical	88	2022	edit	delete