

Name: Ibrahim Reda elmsery

[GitHub](#)

## Task 2

Create a Dockerfile for a simple web app

(Try to use technologies rather than node and mongo)

Create a docker compose file that create the app from it's Docker images and use persistent storage.

### *Sol*

#### [Source code](#)

#### 1-Application Dockerfile

```
Docker-files > app > Dockerfile > ...
1  #this is multi stage docker file
2  #first stage is to build the project and second stage is to deploy the war file to tomcat
3  #this is a simple example of multi stage docker file
4  #this build the artifact and then deploy it to tomcat
5  FROM maven:3.9.9-eclipse-temurin-21-jammy AS build_image
6  RUN git clone https://github.com/hkhcoder/vprofile-project.git
7  RUN cd vprofile-project && git checkout containers && mvn clean install -DskipTests
8  #this will build the project and create the war file in the target folder
9  #second stage
10 FROM tomcat:10-jdk21
11 RUN rm -rf /usr/local/tomcat/webapps/*
12 COPY --from=build_image vprofile-project/target/vprofile-v2.war /usr/local/tomcat/webapps/ROOT.war
13 #/vprofile-project/vprofile-web/target/vprofile-web.war
14 EXPOSE 8080
15 CMD ["catalina.sh", "run"]
16
```

For this Dockerfile I use multistage docker file

**First stage** using image maven to build Java application and give me project.war .

**Second stage** using image Tomcat(application server) copy the artifacts from first stage to second stage

To reduce the size of image then expose application on port 8080 .

#### 2-DB Dockerfile

```
Docker-files > db > Dockerfile > FROM
1  FROM mysql:8.0.33
2  ENV MYSQL_ROOT_PASSWORD="vprodbpass"
3  ENV MYSQL_DATABASE="accounts"
4  ADD db_backup.sql docker-entrypoint-initdb.d/db_backup.sql
5  EXPOSE 3306
6  CMD ["mysqld"]
7
```

That's simple image of mysql you must give it Rootpassword to up that is mandatory

The add my db\_backup.sql to path docker-entrypoint-initdb.d

I get this path from docker hub docs

When a container is started for the first time, a new database with the specified name will be created and initialized with the provided configuration variables. Furthermore, it will execute files with extensions `.sh`, `.sql` and `.sql.gz` that are found in `/docker-entrypoint-initdb.d`

### 3-WebServer

```
1 FROM nginx
2 RUN rm -rf /etc/nginx/conf.d/default.conf
3 COPY nginxvproapp.conf /etc/nginx/conf.d/vproapp.conf
4
```

This dockerfile for nginx I remove default configuration then add my configuration to work as proxy it work on port 80 and will forward requests to port 8080 of tomcat

```
upstream vproapp {
    server vproapp:8080;
}
server {
    listen 80;
    location / {
        proxy_pass http://vproapp;
    }
}
```

## Dockercompose

```
services:
  > Run Service
  vprodb:
    build:
      context: ../Docker-files/db
    image: ibrahimmsery1/vprofiledb
    container_name: vprodb ## it must align with the service name db in -> vprofile-project/src/main/resources/application.properties
    ports:
      - "3306:3306"
    volumes:
      - vprodbdata:/var/lib/mysql
    environment:
      - MYSQL_ROOT_PASSWORD=vprodbpass
    networks:
      - vpro-network

  > Run Service
  vprocache01:
    image: memcached
    ports:
      - "11211:11211"
    networks:
      - vpro-network

  > Run Service
  vpromq01:
    image: rabbitmq
    ports:
      - "5672:5672"
    environment:
      - RABBITMQ_DEFAULT_USER=guest ## it must align with the service name db in -> vprofile-project/src/main/resources/application.properties
      - RABBITMQ_DEFAULT_PASS=guest
    networks:
      - vpro-network
```

```
2
3   > Run Service
4   vproapp:
5     build:
6       context: ../Docker-files/app
7     image: ibrahimmsery1/vprofileapp
8     container_name: vproapp
9     ports:
10       - "8080:8080"
11     volumes:
12       - vproappdata:/usr/local/tomcat/webapps
13     networks:
14       - vpro-network
15
16   > Run Service
17   vproweb:
18     build:
19       context: ../Docker-files/web
20     image: ibrahimmsery1/vprofileweb
21     container_name: vproweb
22     ports:
23       - "80:80"
24     networks:
25       - vpro-network
26
27 volumes:
28   vprodbdata: {}
29   vproappdata: {}
30
31 networks:
32   vpro-network:
33     driver: bridge
```

The name must same in `vprofile-project/src/main/resources/application.properties`

That's application.properties that developer give us

## docker-compose build

```
ubuntu@prometheus:~/ZeroSploit-Training/day-2/code$ docker-compose build
Compose now can delegate build to bake for better performances
Just set COMPOSE_BAKE=true
[+] Building 0.75 (3/23) FINISHED
=> [vproapp internal] load build definition from Dockerfile
=> => transferring dockerfile: 810B
=> [vproweb internal] load build definition from Dockerfile
=> => transferring dockerfile: 149B
=> [vprodb internal] load build definition from Dockerfile
=> => transferring dockerfile: 211B
=> [vproapp internal] load metadata for docker.io/library/mysql:8.0.33
=> [vproweb internal] load metadata for docker.io/library/nginx:latest
=> [vproapp internal] load metadata for docker.io/library/tomcat:10-jdk21
=> [vproapp internal] load metadata for docker.io/library/maven:3.9.0-eclipse-temurin-21-jammy
=> [vprodb auth] library/mysql:pull token for registry-1.docker.io
=> [vproweb auth] library/nginx:pull token for registry-1.docker.io
=> [vproapp auth] library/tomcat:pull token for registry-1.docker.io
=> [vproapp auth] library/maven:pull token for registry-1.docker.io
=> [vprodb internal] load .dockerignore
=> => transferring context: 2B
=> [vproweb internal] load .dockerignore
=> => transferring context: 2B
=> [vprodb internal] load build context
=> => transferring context: 6.22kB
=> [vprodb 1/2] FROM docker.io/library/mysql:8.0.33sha256:ea68e51ffe9b96fe6076f1218af11301aaef13c6701e0ec9aaef5791d5dc5d
=> [vproapp internal] load .dockerignore
=> => transferring context: 2B
=> [vproweb 1/2] FROM docker.io/library/nginx:latestsha256:9d6b58feeb2dbd3c56ab585333d627cc6e281011cfd8050fa4bcf2072c9496
=> [vproweb internal] load build context
=> => transferring context: 153B
=> CACHED [vprodb 2/2] ADD db_backup.sql docker-entrypoint-initdb.d/db_backup.sql
=> [vprodb] exporting to image
=> => exporting layers
=> => writing image sha256:g1f241bd09a67d67dbcd91955d813ec4cf6f8caef6e2671b5693aaa90d1319
=> => naming to docker.io/ibrahimelmsery1/vprofiledb
=> [vproapp build_image 1/3] FROM docker.io/library/maven:3.9.0-eclipse-temurin-21-jammysha256:164799975dfa005f4c42e64f07ceff2fb2b0992f8f74a44fbb8fd95c3457b2a8
=> [vproapp stage-1 1/3] FROM docker.io/library/tomcat:10-jdk21sha256:d7a144546ff686803354d6d75445137ab6b064de5d76006fffb408aa70603e0b
=> CACHED [vproapp stage-1 2/3] RUN rm -rf /usr/local/tomcat/webapps/*
=> CACHED [vproapp build_image 2/3] RUN git clone https://github.com/hkhcoder/vprofile-project.git
=> CACHED [vproapp build_image 3/3] RUN cd vprofile-project && git checkout containers && mvn clean install -DskipTests
=> CACHED [vproapp stage-1 3/3] COPY --from=build_image vprofile-project/target/vprofile-v2.war /usr/local/tomcat/webapps/ROOT.war
=> [vproapp] exporting to image
=> => exporting layers
=> => writing image sha256:8b9738559e4d8d4daa42ee8296856faa40a895dfc4c8e1b705b5a5e0b71f5
=> => naming to docker.io/ibrahimelmsery1/vprofileapp
=> CACHED [vproweb 2/3] RUN rm -rf /etc/nginx/conf.d/default.conf
=> CACHED [vproweb 3/3] COPY nginxvproapp.conf /etc/nginx/conf.d/vproapp.conf
=> [vproweb] exporting to image
=> => exporting layers
=> => writing image sha256:ed88887d787bb06bc4c67c84851025892c0b1e8b7f6b24d3dc04a6cccecbcc63
=> => naming to docker.io/ibrahimelmsery1/vprofileweb
=> [vprodb] resolving provenance for metadata file
=> [vproapp] resolving provenance for metadata file
=> [vproweb] resolving provenance for metadata file
[+] Building 3/3
✓ vproapp Built
```

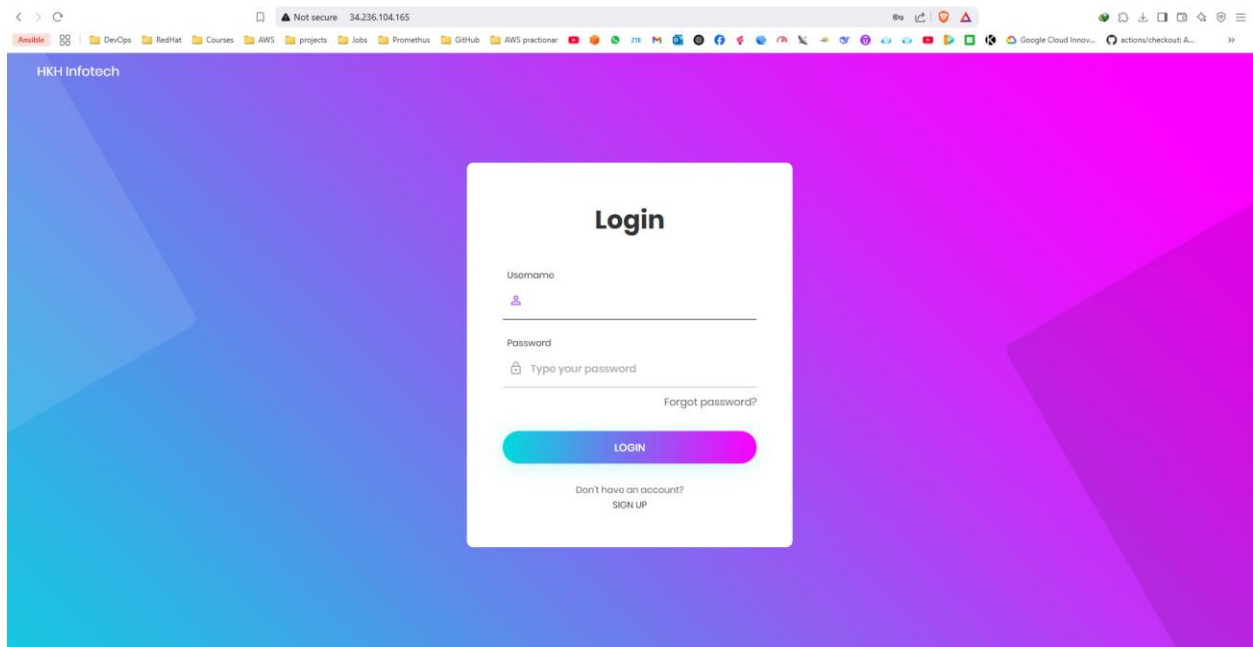
## docker-compose up -d

```
ubuntu@prometheus:~/ZeroSploit-Training/day-2/code$ docker-compose up -d
[+] Running 17/17
✓ vpromq01 Pulled
✓ 5a7813e071bf Already exists
✓ 85501ecaa030 Pull complete
✓ 93f0b4fc063e Pull complete
✓ e81c1fe1aabb3 Pull complete
✓ d28379621bee Pull complete
✓ 49713ded3555 Pull complete
✓ 6ead817cd7b9 Pull complete
✓ f547bfe987eb Pull complete
✓ 4682ab1b02bd Pull complete
✓ vprocache01 Pulled
✓ 7cf63256a31a Already exists
✓ aad02ec62adc Pull complete
✓ f3fd8efc49ea Pull complete
✓ 70987d4fe68d Pull complete
✓ 11cbf3d5f33d Pull complete
✓ e7394c9da6d6 Pull complete
[+] Running 5/5
✓ Container vproweb Started
✓ Container vprodb Started
✓ Container code-vprocache01-1 Started
✓ Container vproapp Started
✓ Container code-vpromq01-1 Started
ubuntu@prometheus:~/ZeroSploit-Training/day-2/code$
```

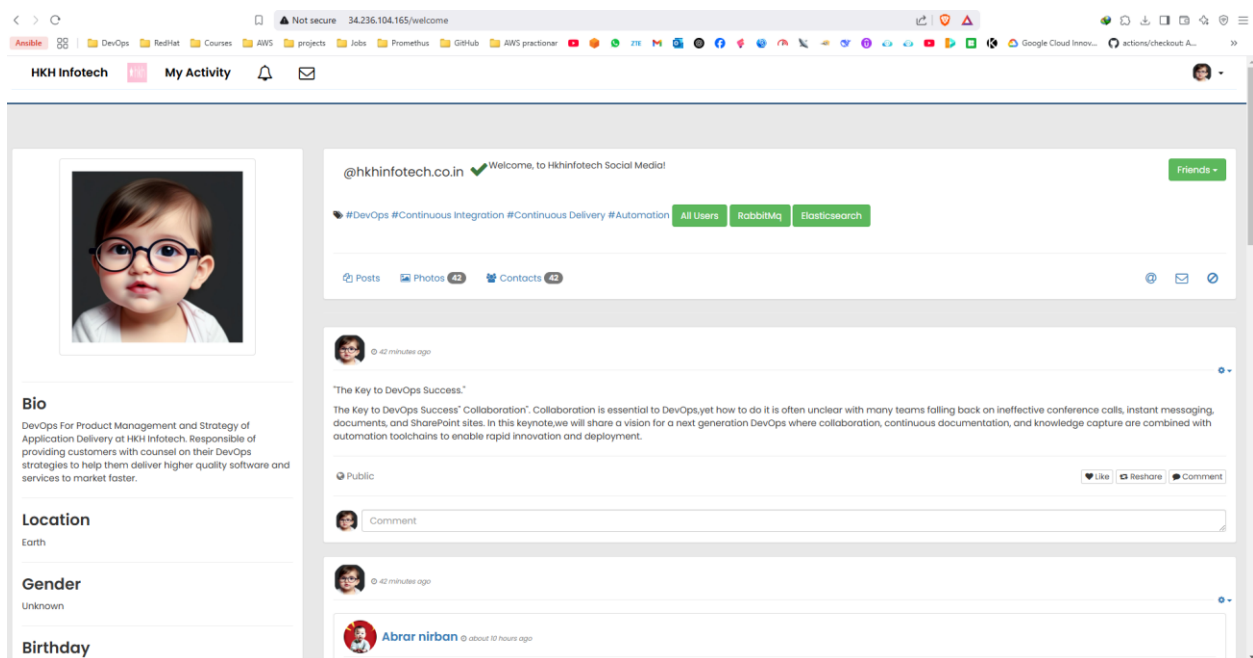
## docker-compose ps

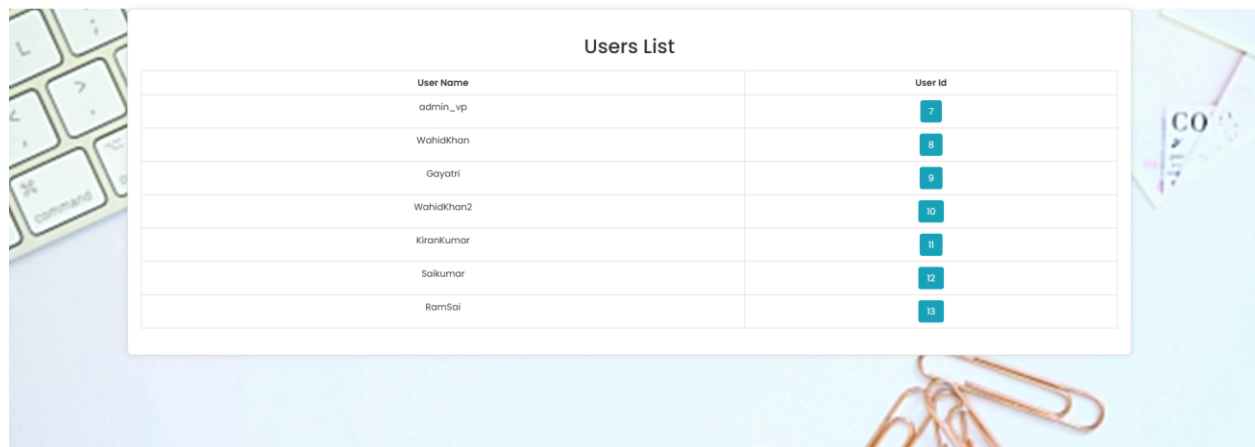
```
Container code-vpromq01-1 Started
ubuntu@prometheus:~/ZeroSploit-Training/day-2/code$ docker-compose ps
NAME                                IMAGE                                SERVICE    CREATED         STATUS         PORTS
code-vprocache01-1                 memcached                          "docker-entr... 10 minutes ago Up 10 minutes 0.0.0.0:11211->11211/tcp, [::]:11211->11211/tcp
code-vpromq01-1                    rabbitmq                            "docker-entr... 10 minutes ago Up 10 minutes 4369/tcp, 5671/tcp, 15691-15692/tcp, 25672/tcp, 0.0.0.0:5672->5672/tcp, [::]:5672->5672/tcp
vproapp                            ibrahimelmsery1/vprofileapp       "catalina.sh r... vproapp      10 minutes ago Up 10 minutes 0.0.0.0:8080->8080/tcp, [::]:8080->8080/tcp
vprodb                             ibrahimelmsery1/vprofiledb        "docker-entr... 10 minutes ago Up 10 minutes 0.0.0.0:3306->3306/tcp, [::]:3306->3306/tcp, 33060/tcp
vproweb                           ibrahimelmsery1/vprofileweb       "docker-entr... 10 minutes ago Up 10 minutes 0.0.0.0:80->80/tcp, [::]:80->80/tcp
ubuntu@prometheus:~/ZeroSploit-Training/day-2/code$
```

I install all of this on EC2 instance so I have public IP I will hit it .



Application up and running





User Name	User Id
admin_vp	7
Wahidkhan	8
Gayatri	9
Wahidkhan2	10
Kirankumar	11
Saikumar	12
RamSai	13

DB up and running

```
ubuntu@prometheus:~/ZeroSploit-Training/day-2/code$ docker-compose down
[+] Running 4/5
  Container vprodb          Stopping
  ✓ Container code-vpromq01-1 Removed
  ✓ Container vproapp       Removed
  ✓ Container code-vprocache01-1 Removed
  ✓ Container vproweb       Removed
```

Docker-compose down

Done

Thank you 😊