Assignments Containerization,

Docker, and Docker Hub:

Submitted by – Thamilselvan.D

Date of submission 09/07/2024

Submitted to – vikul

L2 - Create the Container using the same Application Image and run the application in a Web Browser using container port mapping

Step 1 installed docker in ec2 ubuntu instance using command

apt install Docker.io and verified the version using command
docker –-version

```
No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-20-51:/home/ubuntu# docker --version
Docker version 24.0.7, build 24.0.7-0ubuntu2~22.04.1
root@ip-172-31-20-51:/home/ubuntu# 
```

Step2 by using git repository

https://github.com/Tamil1011/Addressbook2.git fetched the address book in ec2 ubuntu instance

```
No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-20-51:/home/ubuntu# git clone https://github.com/Tamil1011/Addressbook2.git
Cloning into 'Addressbook2'...
remote: Enumerating objects: 356, done.
remote: Total 356 (delta 0), reused 0 (delta 0), pack-reused 356
Receiving objects: 100% (356/356), 31.12 MiB | 35.92 MiB/s, done.
Resolving deltas: 100% (127/127), done.
root@ip-172-31-20-51:/home/ubuntu# ls
Addressbook2
root@ip-172-31-20-51:/home/ubuntu# 
```

And installed maven to package the code and to create WAR file

apt install maven -y

man package

```
[INFO] ------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------
[INFO] Total time:  36.175 s
[INFO] Finished at: 2024-07-12T15:59:27Z
[INFO] ------------------------------------------------------------
root@ip-172-31-20-51:/home/ubuntu/Addressbook2#
root@ip-172-31-20-51:/home/ubuntu/Addressbook2# ls
Babufile    Jenkinsfile  addressbook_screenshot.png  build.xml       deploy-docker.yml  file1.txt  provider.tf                          src
Dockerfile  README.md    build.properties            create-instance.tf  dev.inv        pom.xml    sonar-project.properties  target
root@ip-172-31-20-51:/home/ubuntu/Addressbook2# cd target/
root@ip-172-31-20-51:/home/ubuntu/Addressbook2/target# ls
addressbook  addressbook.war  classes  generated-sources  generated-test-sources  maven-archiver  maven-status  surefire-reports  test-classes
root@ip-172-31-20-51:/home/ubuntu/Addressbook2/target#
```

Created the docker file

```
root@ip-172-31-20-51:/home/ubuntu/Addressbook2# cat Dockerfile
FROM tomcat:9
COPY /target/*.war /usr/local/tomcat/webapps/address.war
EXPOSE 8080
CMD ["catalina.sh","run"]
root@ip-172-31-20-51:/home/ubuntu/Addressbook2#
```

Created the image out of docker file

docker build -t address .

```
root@ip-172-31-20-51:/home/ubuntu/Addressbook2# docker images
REPOSITORY    TAG      IMAGE ID       CREATED          SIZE
address       latest   759089aa4a92   29 seconds ago   472MB
tomcat        9        11d72868fce2   4 days ago       456MB
root@ip-172-31-20-51:/home/ubuntu/Addressbook2#
```

Created the container out of image

```
root@ip-172-31-20-51:/home/ubuntu/Addressbook2# docker run -itd address:latest
bed4de2776e0bc4dd305291c14f78d4c0506321565eabc92d721cc1bc91d5340
root@ip-172-31-20-51:/home/ubuntu/Addressbook2# docker ps -a
CONTAINER ID   IMAGE            COMMAND             CREATED          STATUS         PORTS      NAMES
bed4de2776e0   address:latest   "catalina.sh run"   10 seconds ago   Up 8 seconds   8080/tcp   elegant_robinson
root@ip-172-31-20-51:/home/ubuntu/Addressbook2#
```

Changes the security group

EC2 > Security Groups > sg-08f164eb86cab2049 - default > Edit inbound rules

Edit inbound rules Info

Inbound rules control the incoming traffic that's allowed to reach the instance.

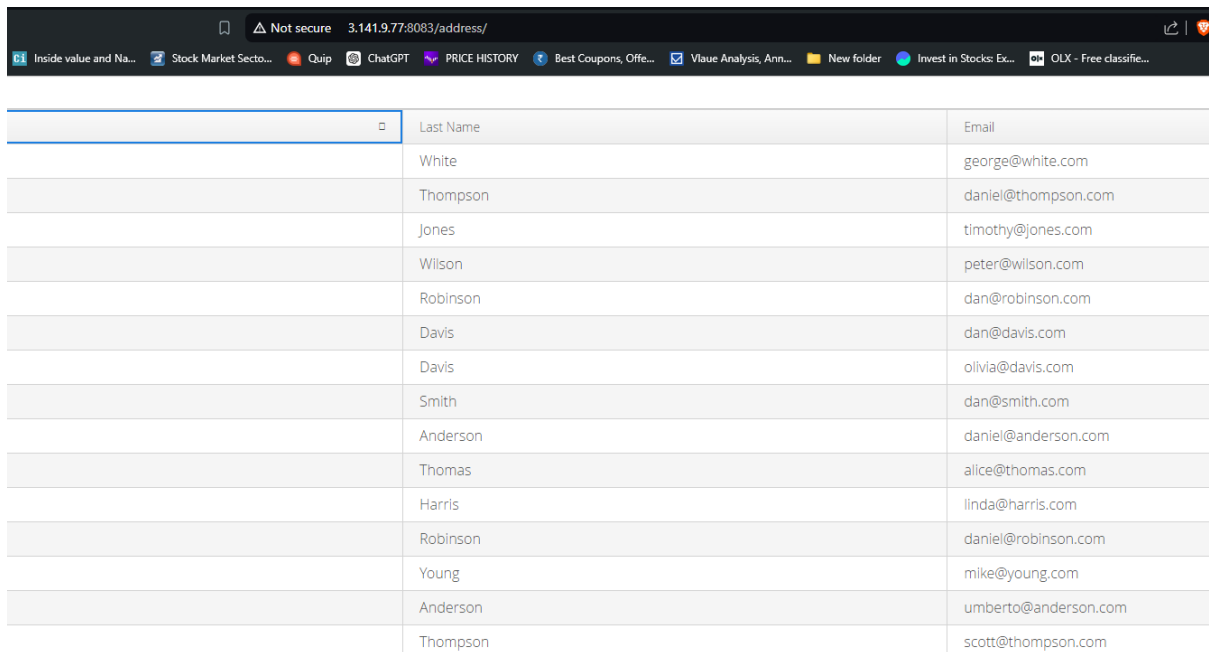**Inbound rules** Info

Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info		Description - op
sgr-0a03bd338e1b596f3	Custom TCP ▼	TCP	8080 - 8089	Custom ▼	Q	addresss app
					0.0.0.0/0 ✕	
sgr-0bd3b0cc6fab119f9	SSH ▼	TCP	22	Custom ▼	Q	

running the application in a Web Browser using container port mapping

step 3 exposing the docker port to the ubuntu instance so the end user can access the application from any ip address

docker run -itd -p 8083:8080 address:latest

OUTPUT

	Last Name	Email
	White	george@white.com
	Thompson	daniel@thompson.com
	Jones	timothy@jones.com
	Wilson	peter@wilson.com
	Robinson	dan@robinson.com
	Davis	dan@davis.com
	Davis	olivia@davis.com
	Smith	dan@smith.com
	Anderson	daniel@anderson.com
	Thomas	alice@thomas.com
	Harris	linda@harris.com
	Robinson	daniel@robinson.com
	Young	mike@young.com
	Anderson	umberto@anderson.com
	Thompson	scott@thompson.com