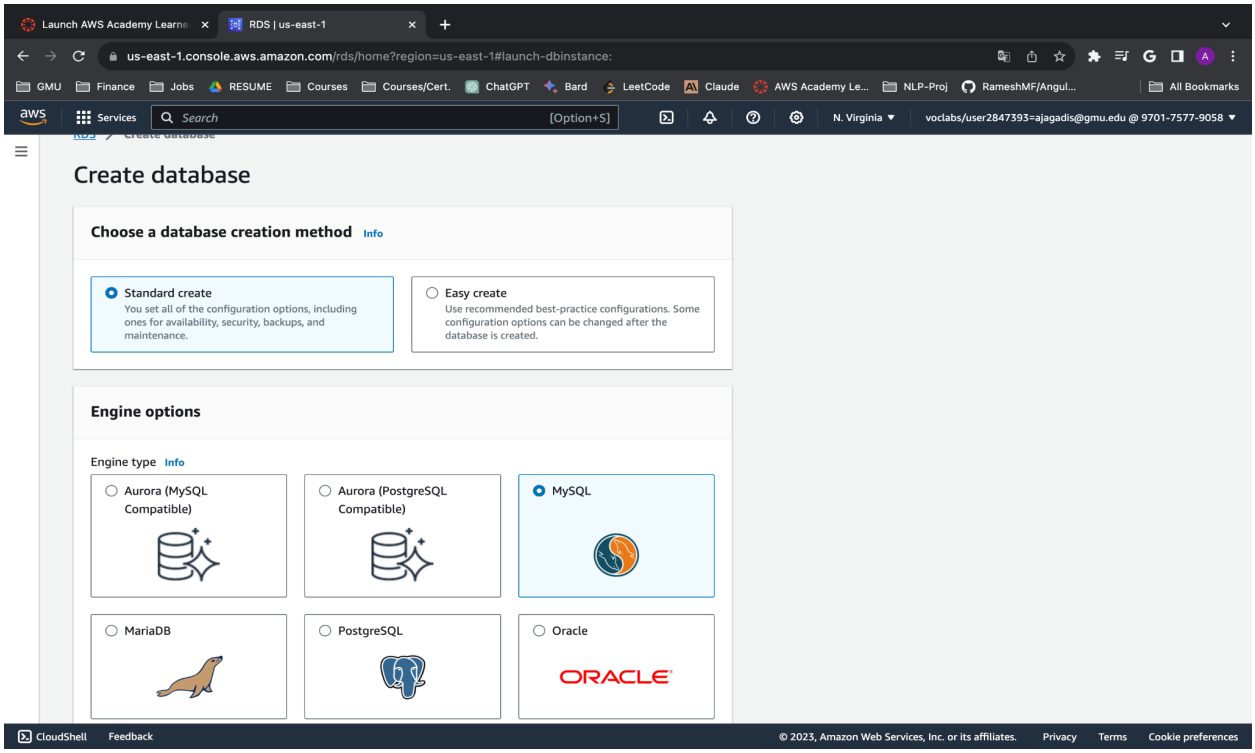
SWE 645 Assignment 3

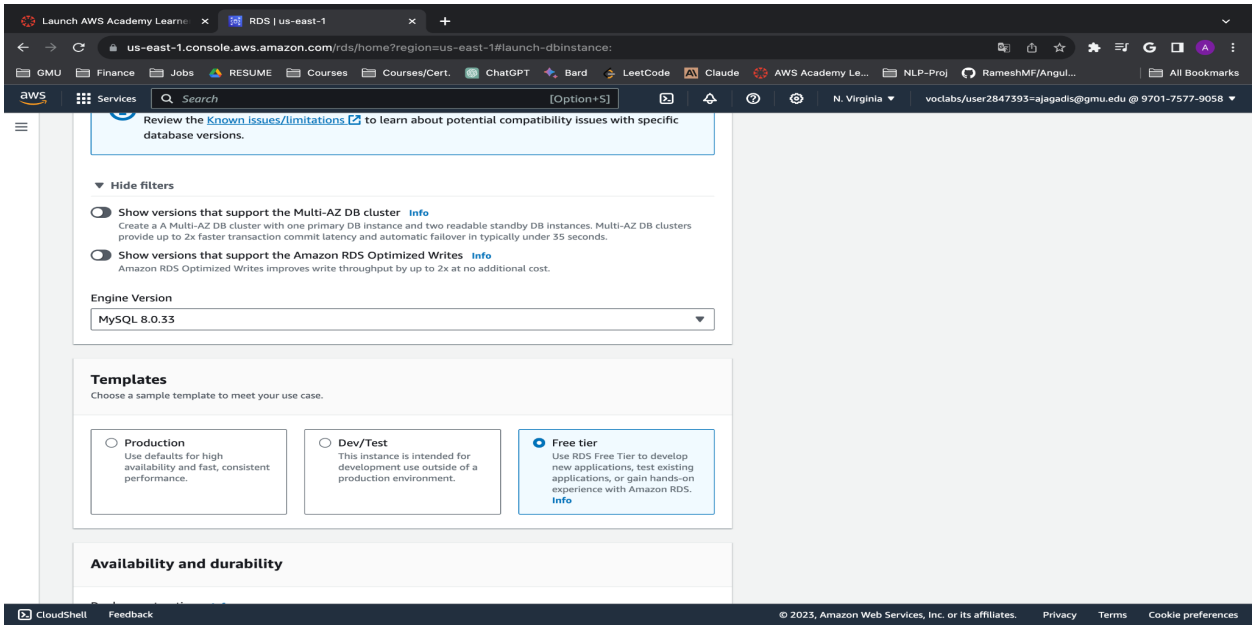
Team Members:
AKSHITA SRIKANTH - G01393896
ANUSHA JAGADISH - G01394233
AKASH PONNAM - G01378189
SHISHWAN GANDE - G01383809

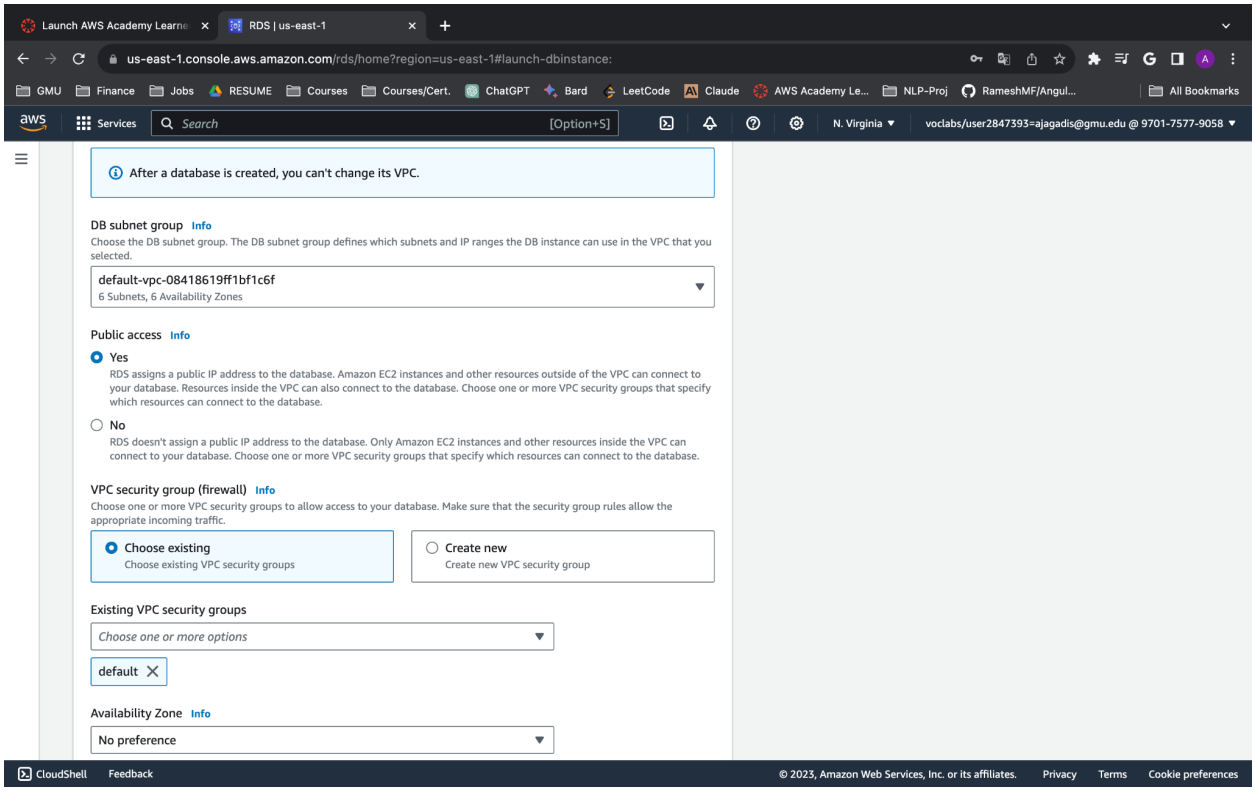
CREATING AWS RDS CONNECTION

- Sign in to the AWS console and go to the RDS service page.
- Select the option to create a new Database.
- Choose the Standard Create method to build the database.
- Select MySQL as the database engine.

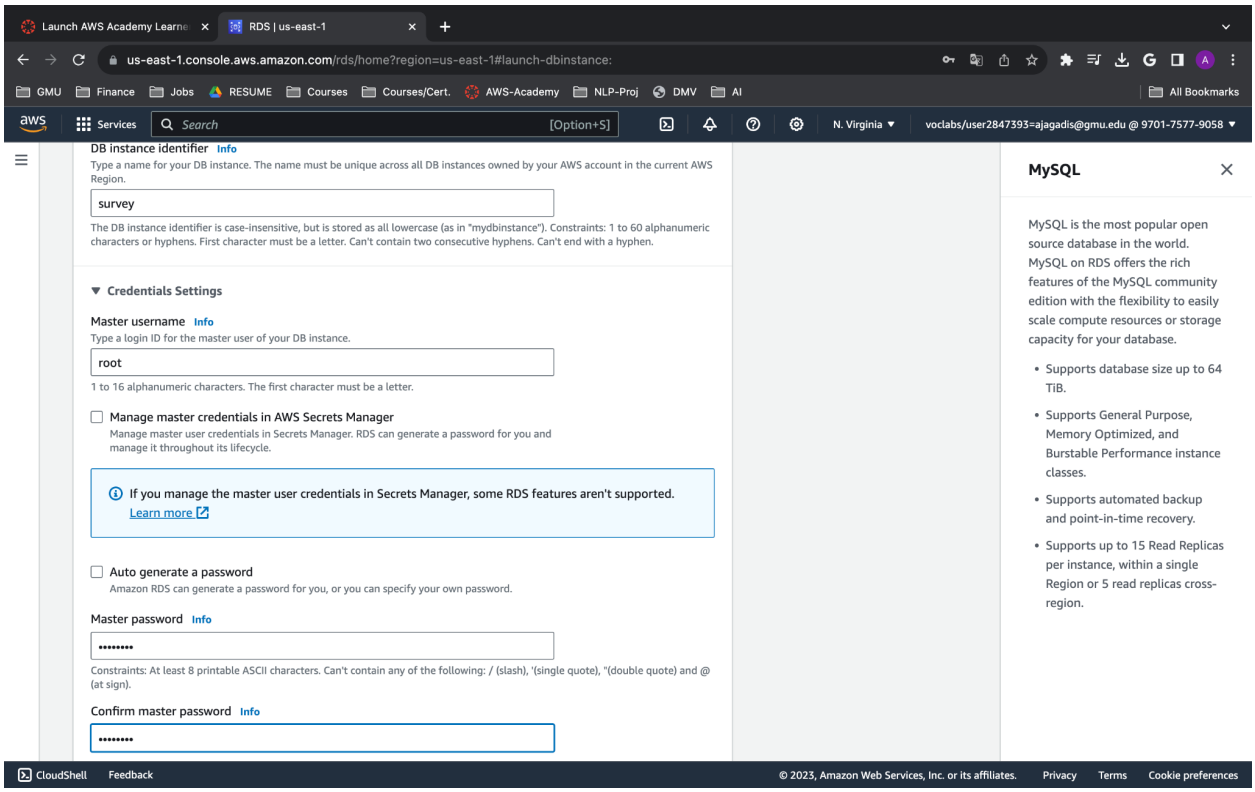


- Pick the db.t2.micro tier for the database instance computing capacity and storage allocation.
- Default storage configurations.
- Click the free tier as templates.

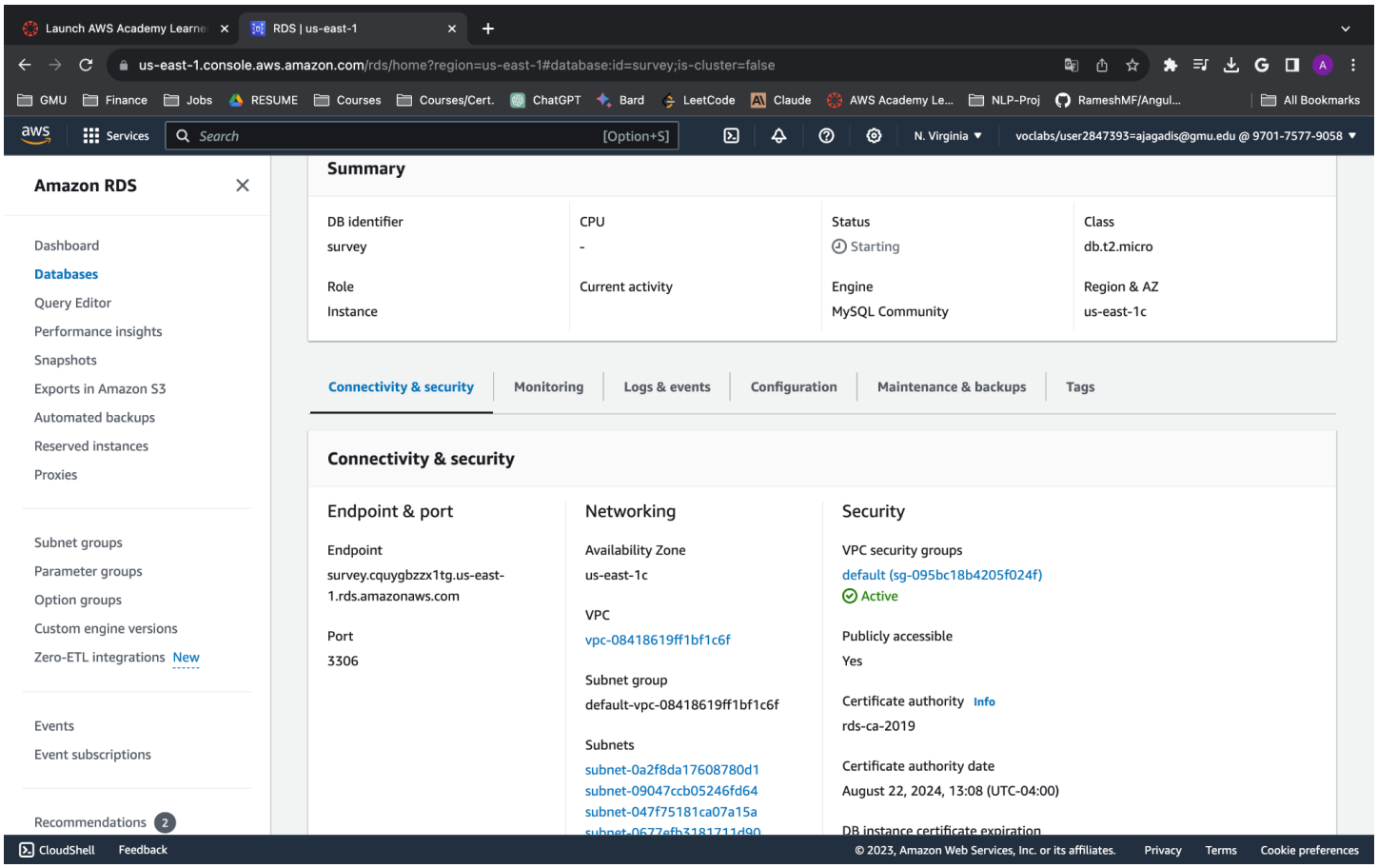




- Name the database, create a username and password to access it.

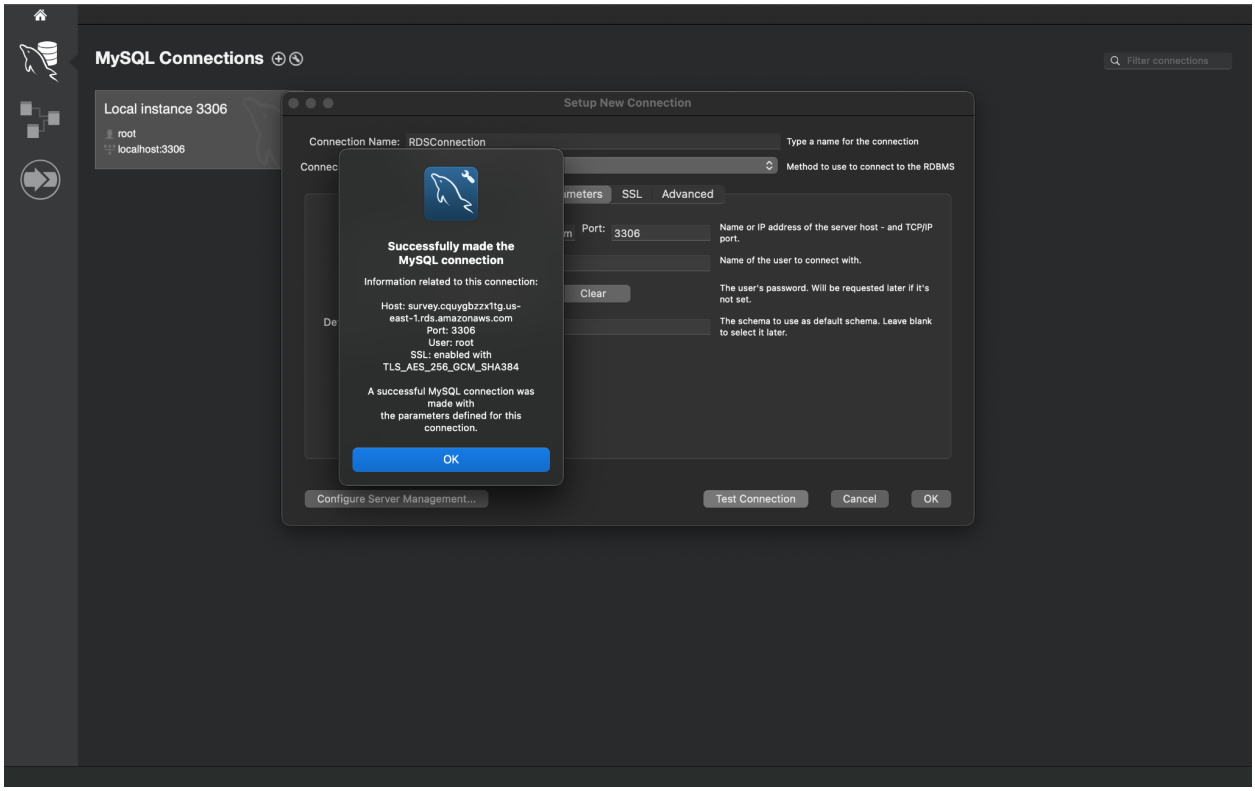


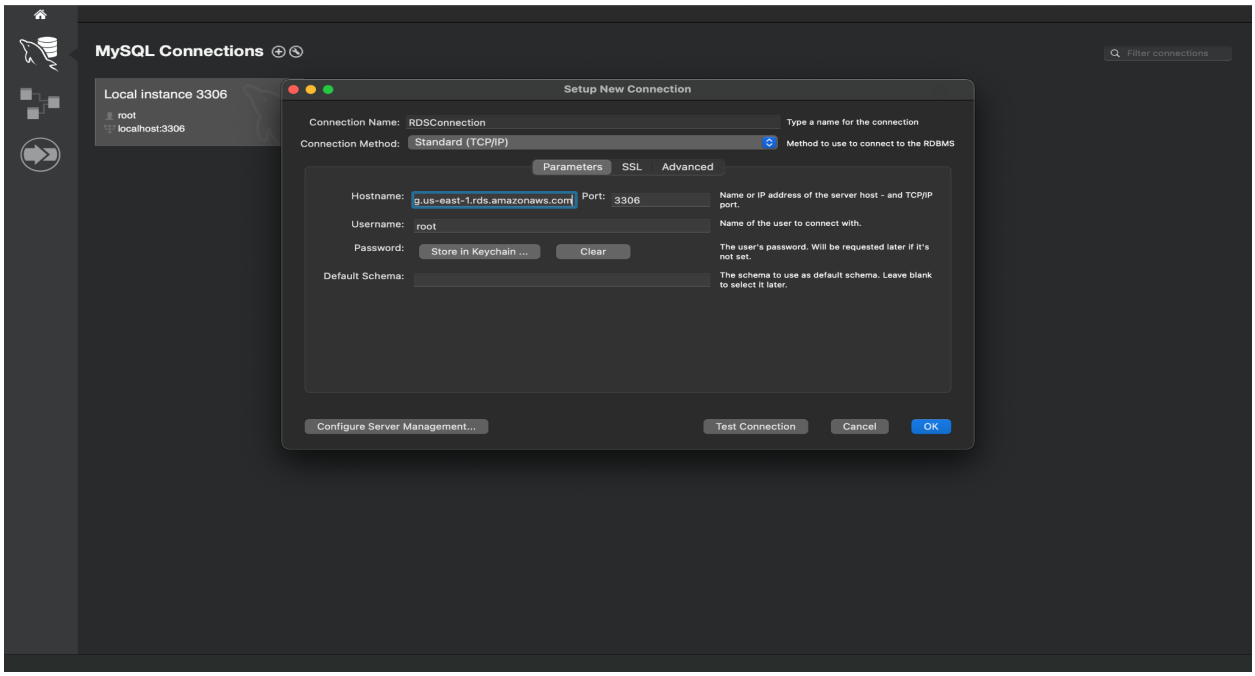
- Database will be created
- Database Endpoint : survey.cquygbzzx1tg.us-east-1.rds.amazonaws.com



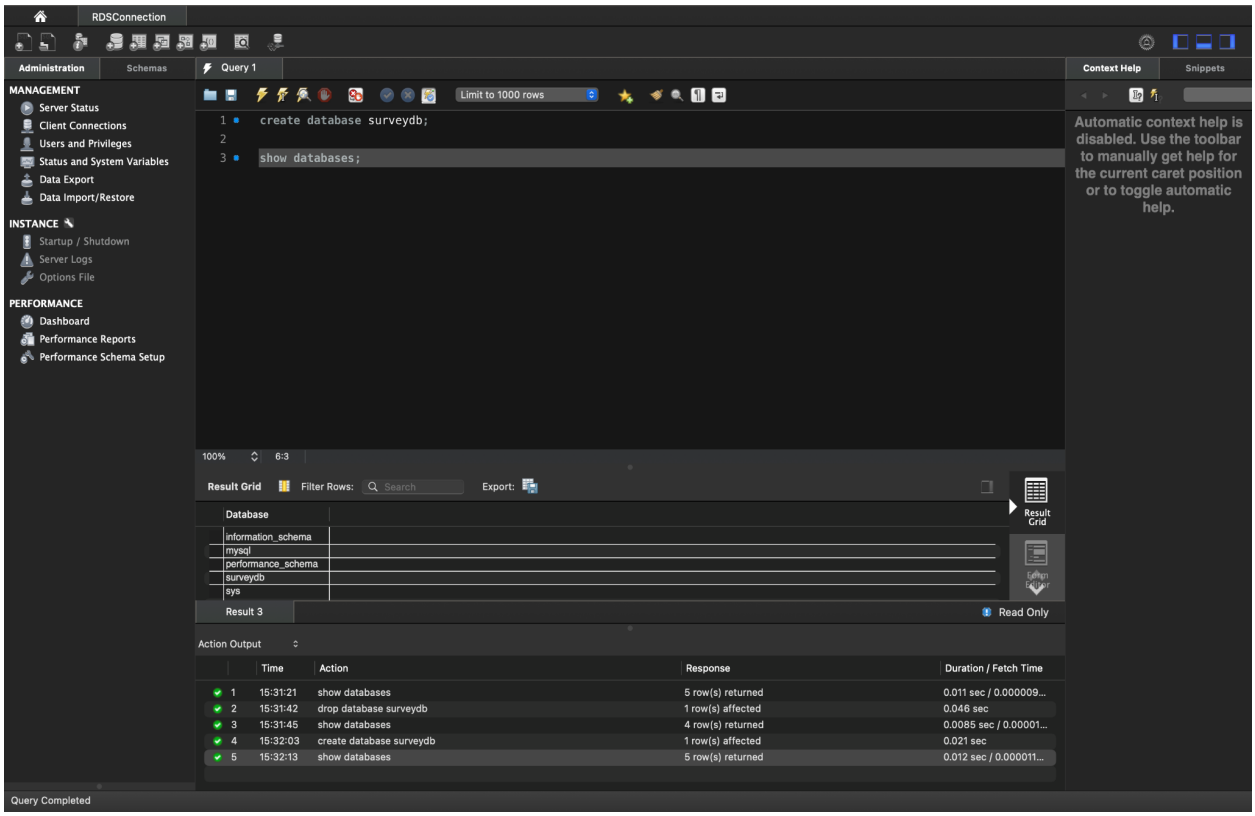
CONNECTING AWS MYSQL IN MYSQL WORKBENCH

- Setup a New Connection by clicking + icon.
- Give a name in the connection name tab.
- Give the AWS MySQL endpoint in the Hostname tab.
- Give the username and password which was given while creating the database in AWS.
- Click on Test connection and apply OK.



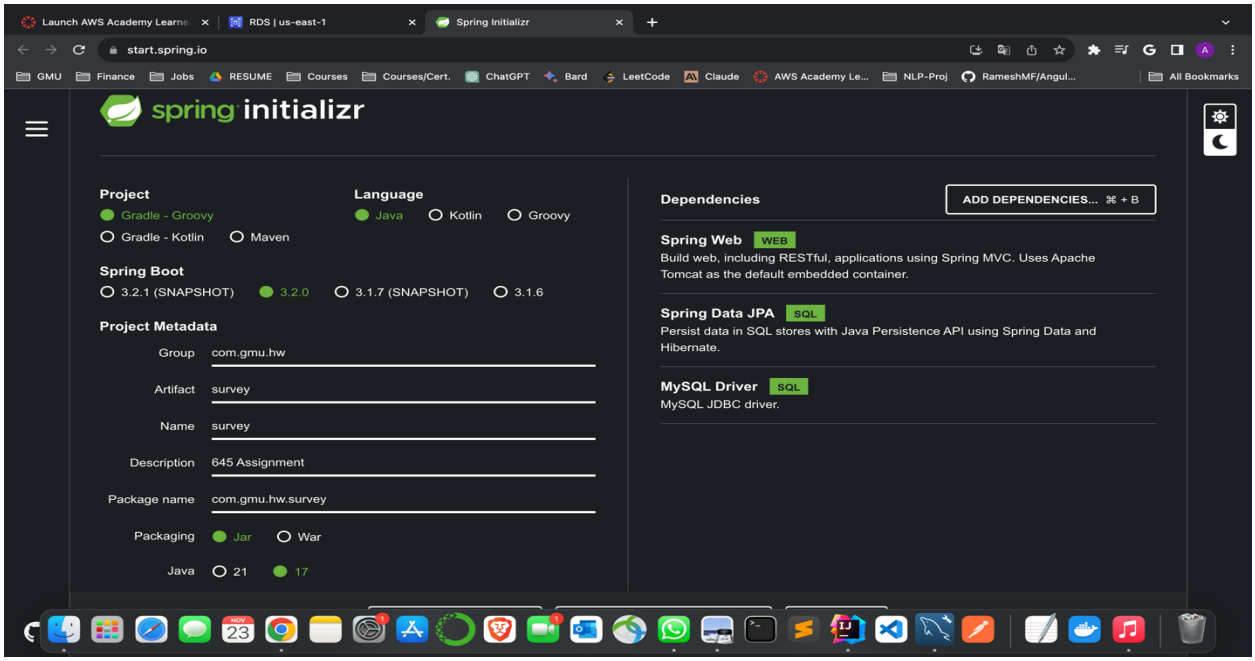


- Connection will be established to MySQL.
- Create a Database with name surveydb with the command
  CREATE DATABASE surveydb;

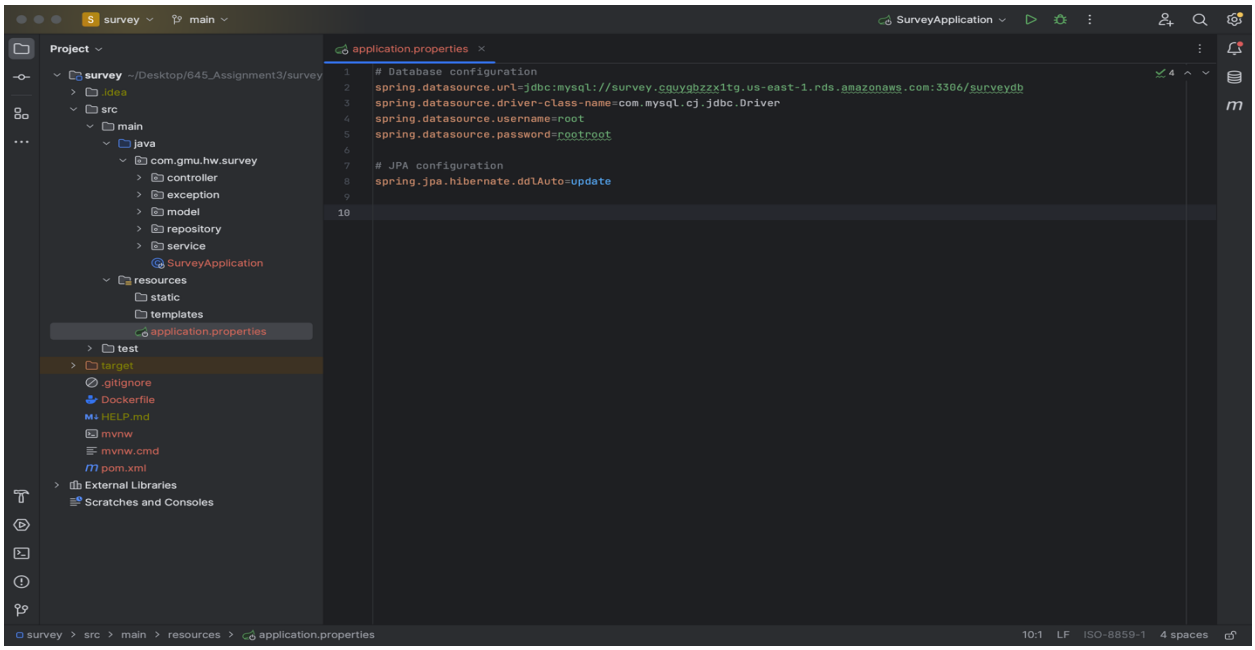


CREATING SPRING BOOT APPLICATION

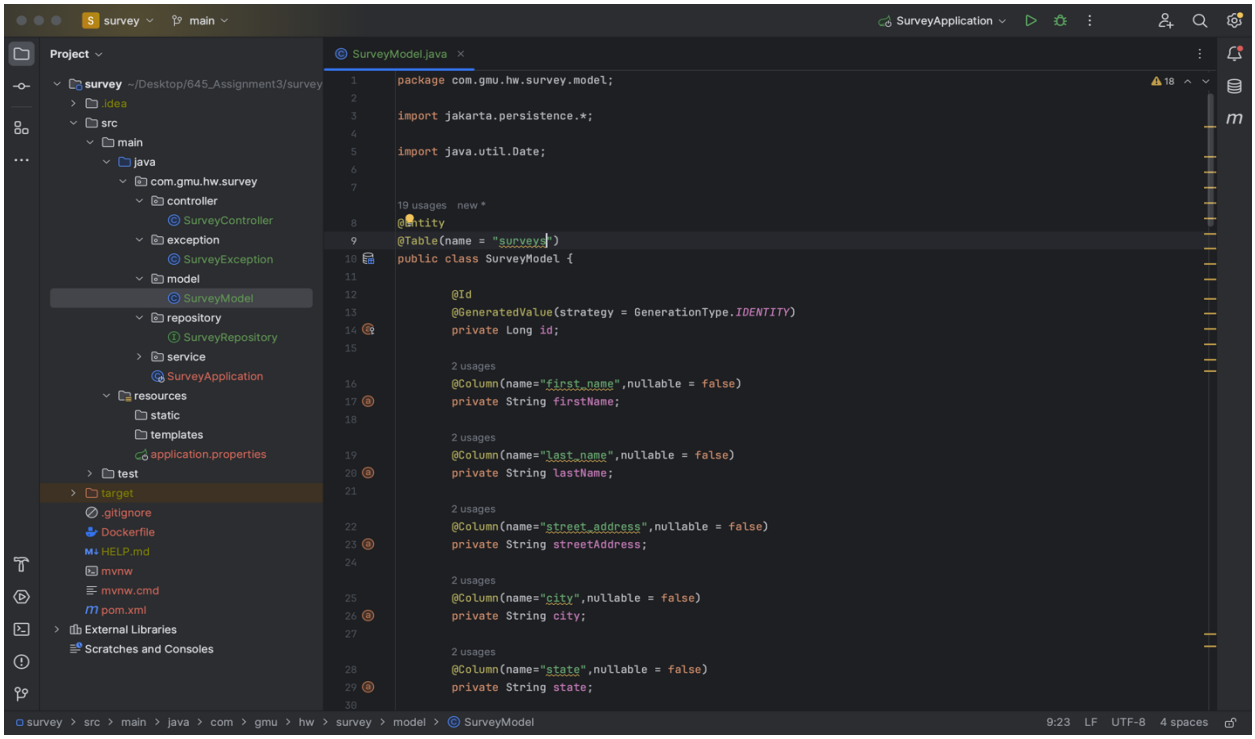
- Navigate to https://start.spring.io/ and download spring project with the below specs and click generate.



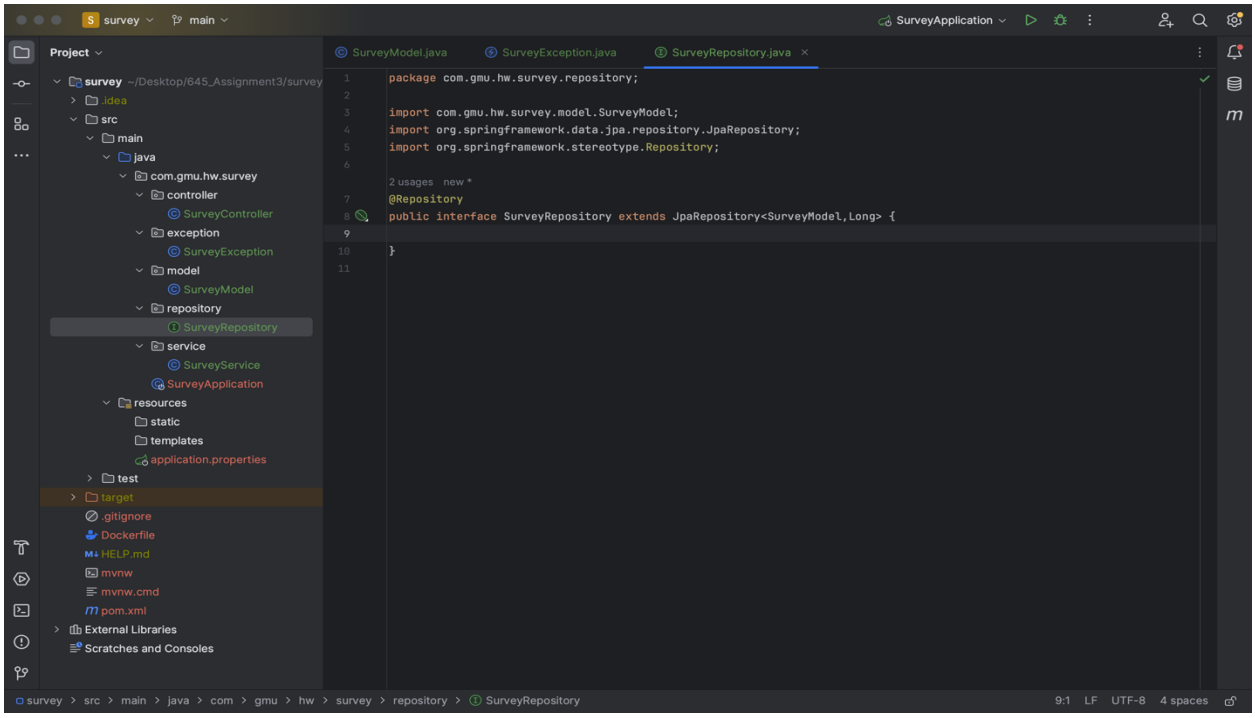
- Navigate to the application.properties file and add the below details.



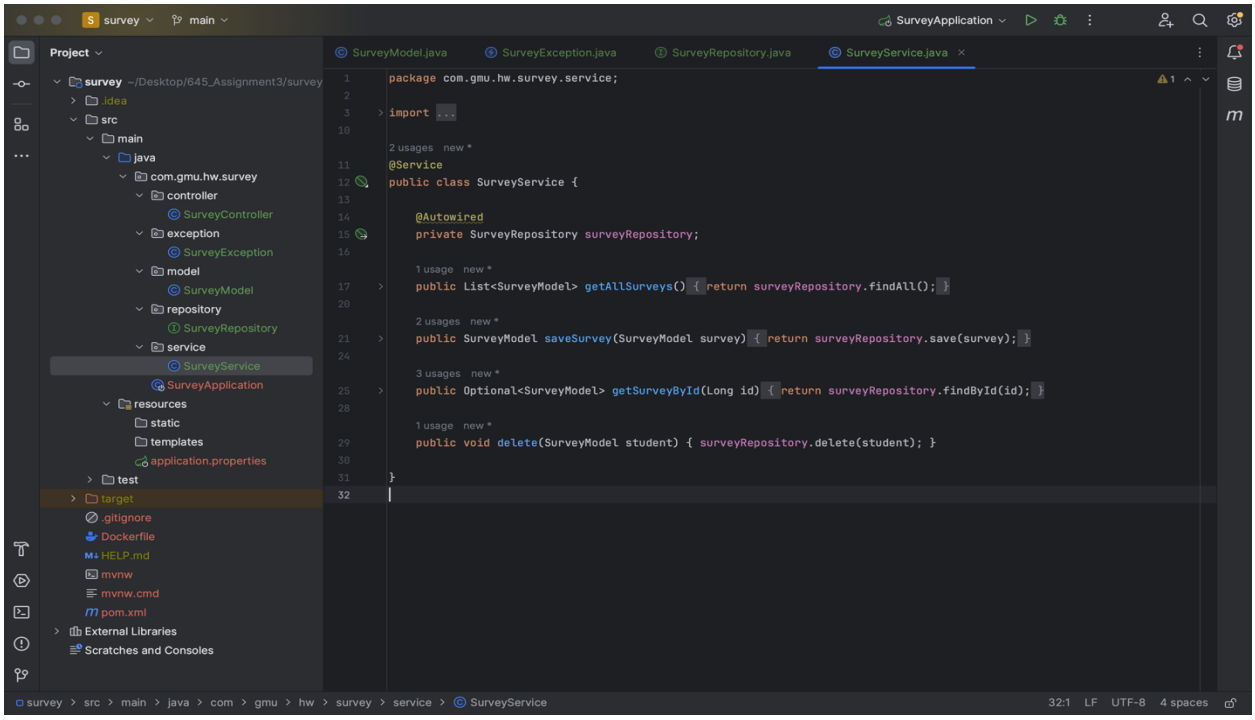
- Create SurveyModel which contains all the specifications of the table



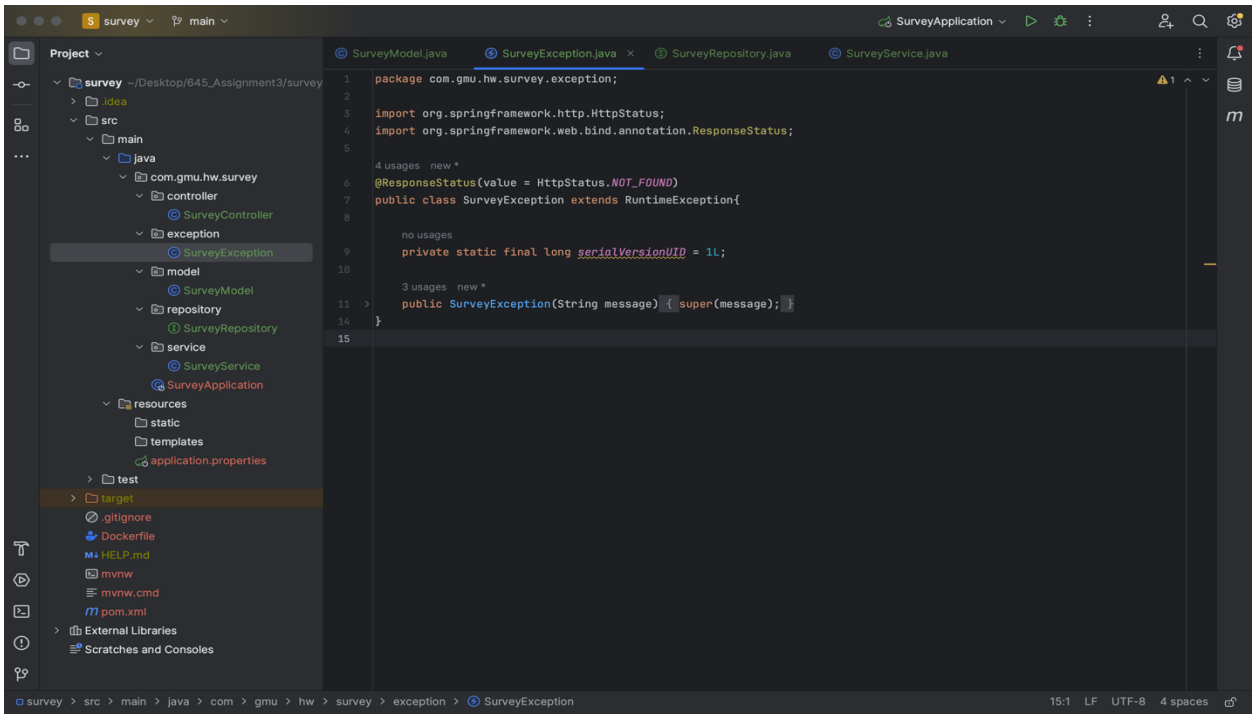
- Create JpaRepository



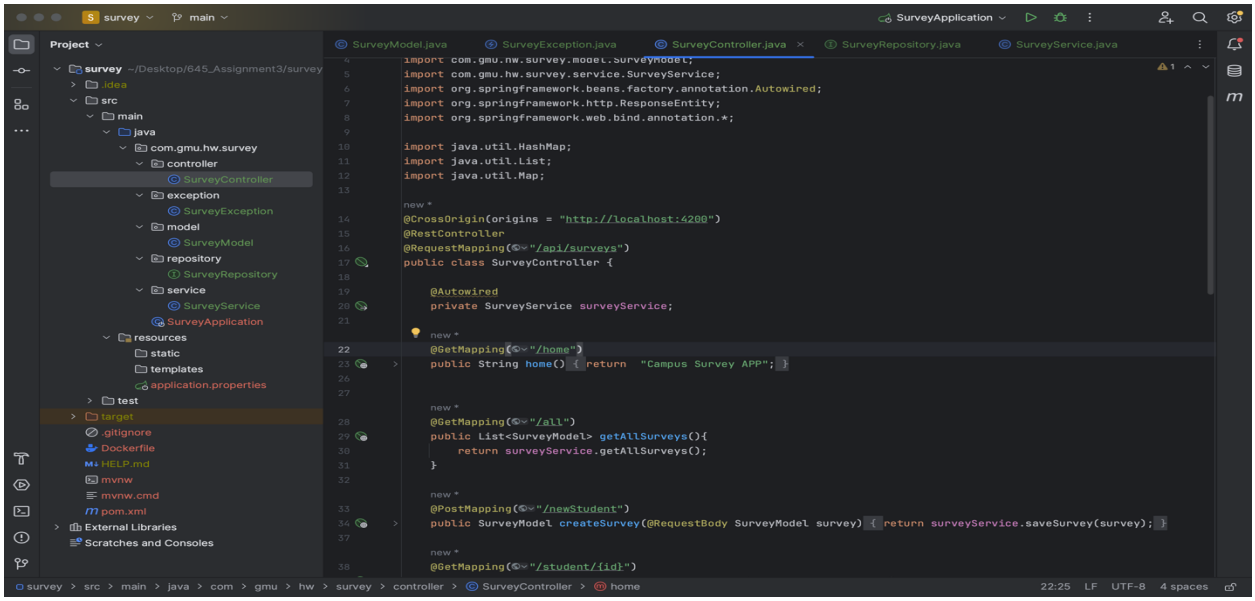
- Create Survey which extends the JpaRepository



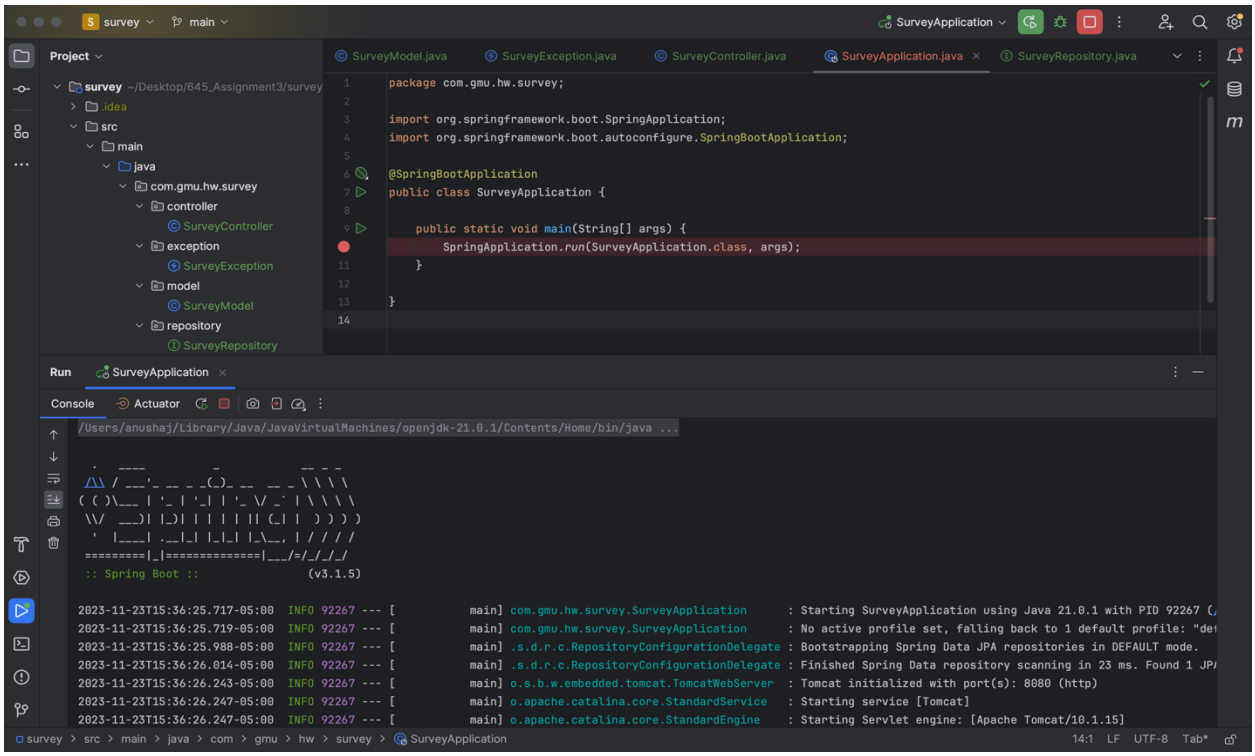
- Create an exception class to handle the exceptions



- Create a controller class to handle the REST api calls

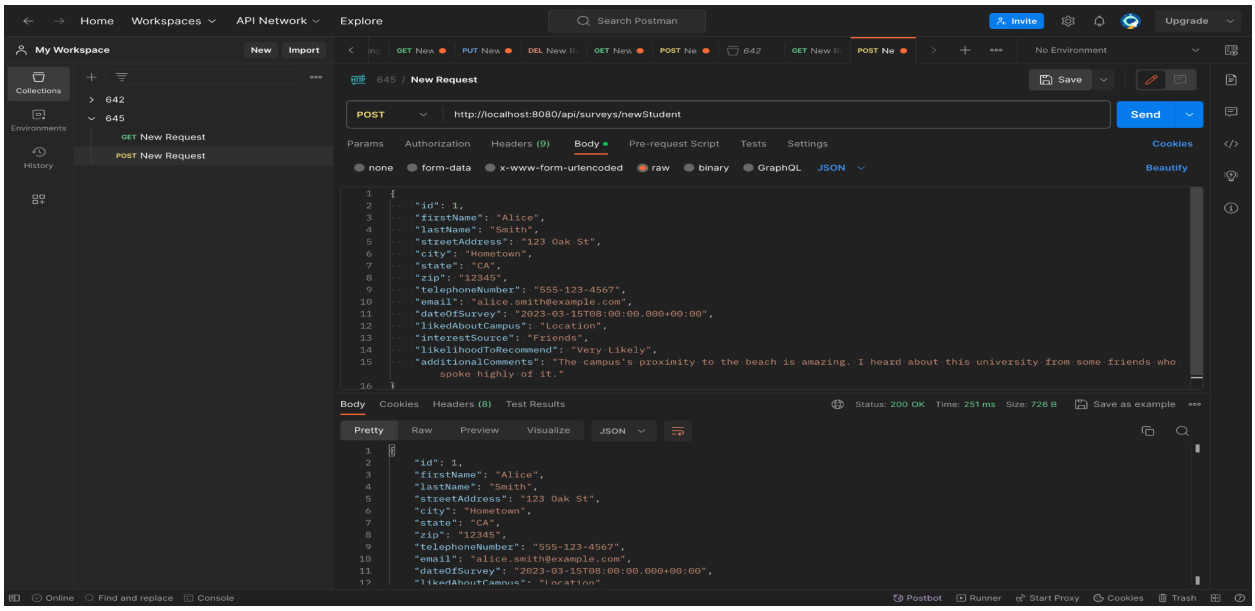


- Open the Survey Application and run the application

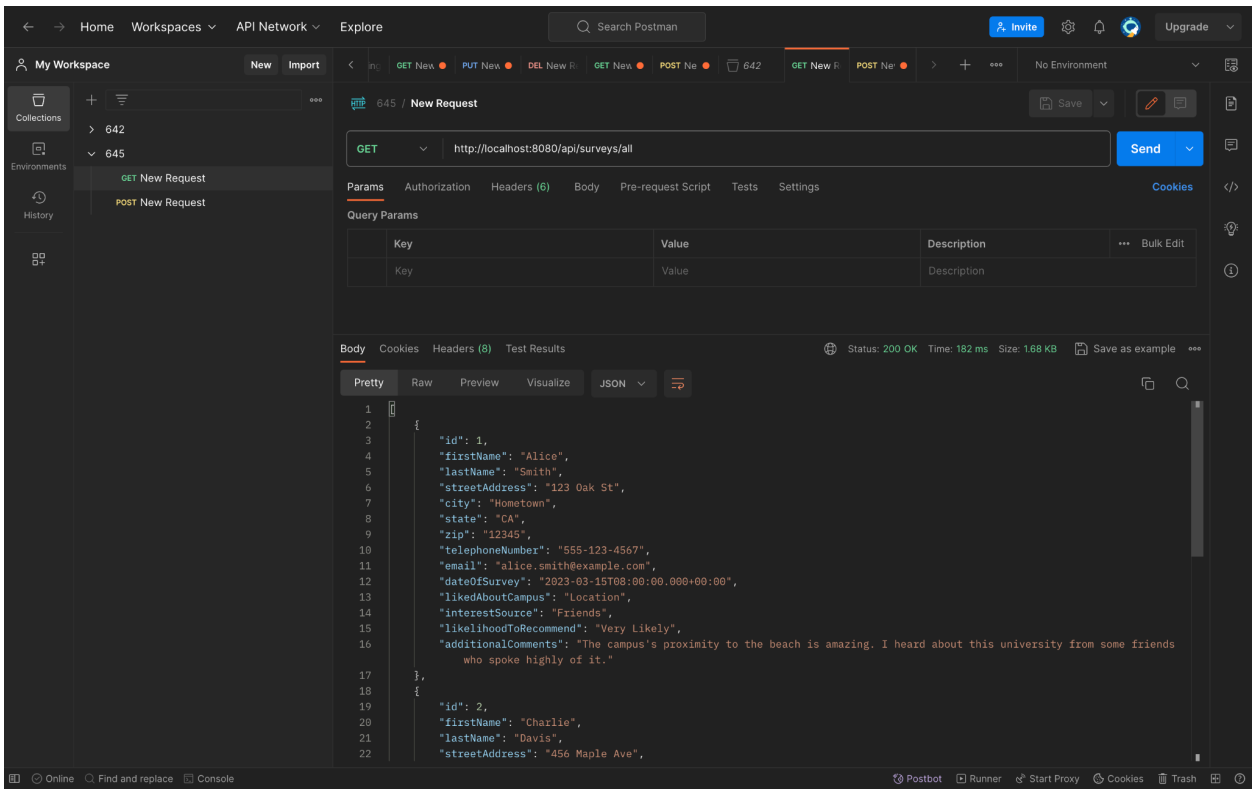


POSTMAN WITHOUT DOCKER

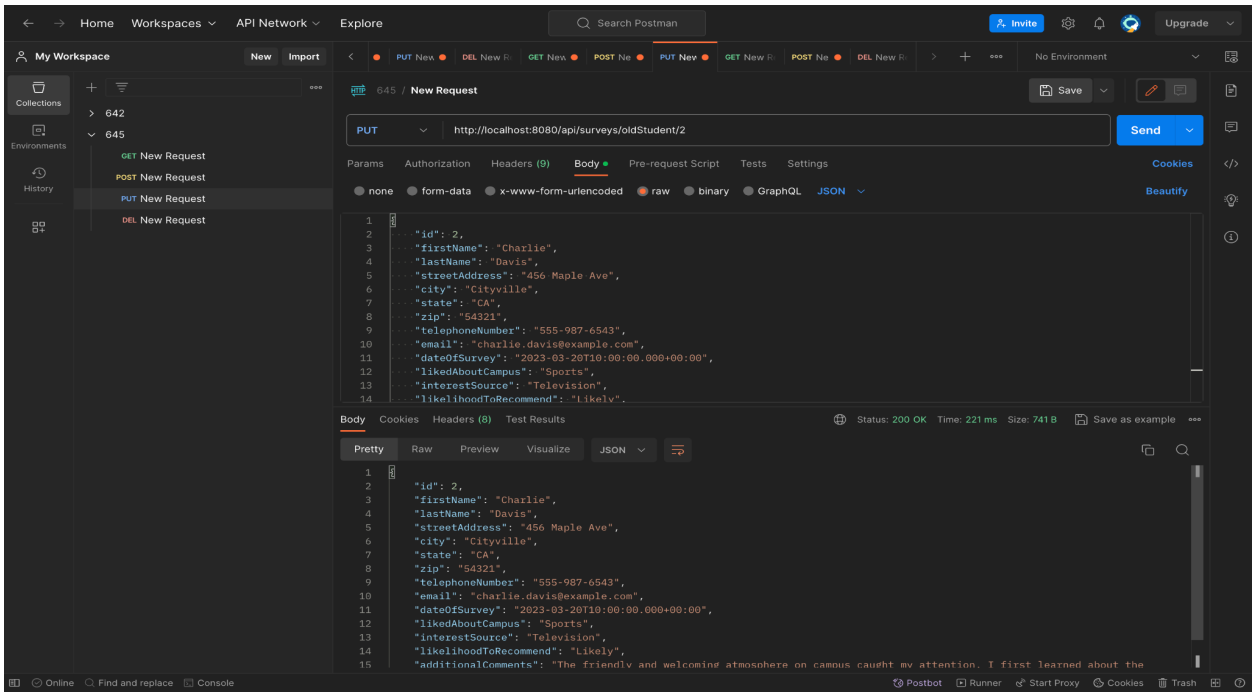
- Creating a new survey using the POST method.



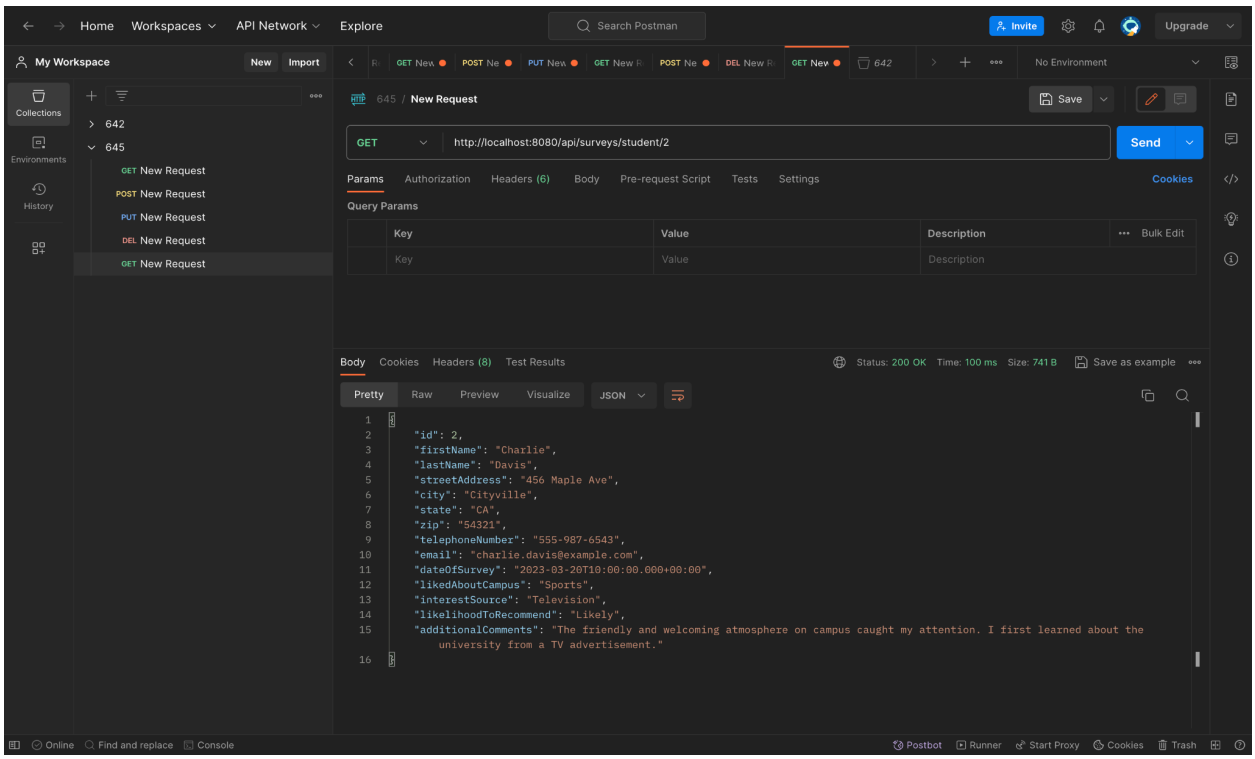
- Retrieving all the surveys from the GET method.



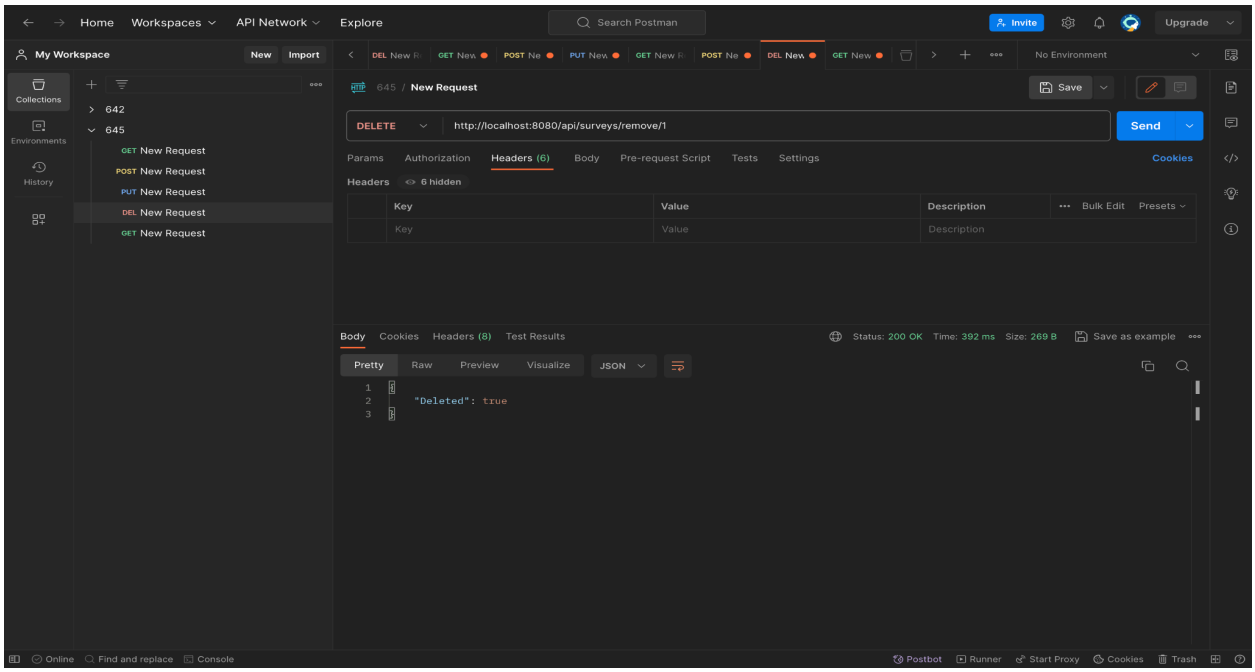
- Updating a particular survey details using the UPDATE method.



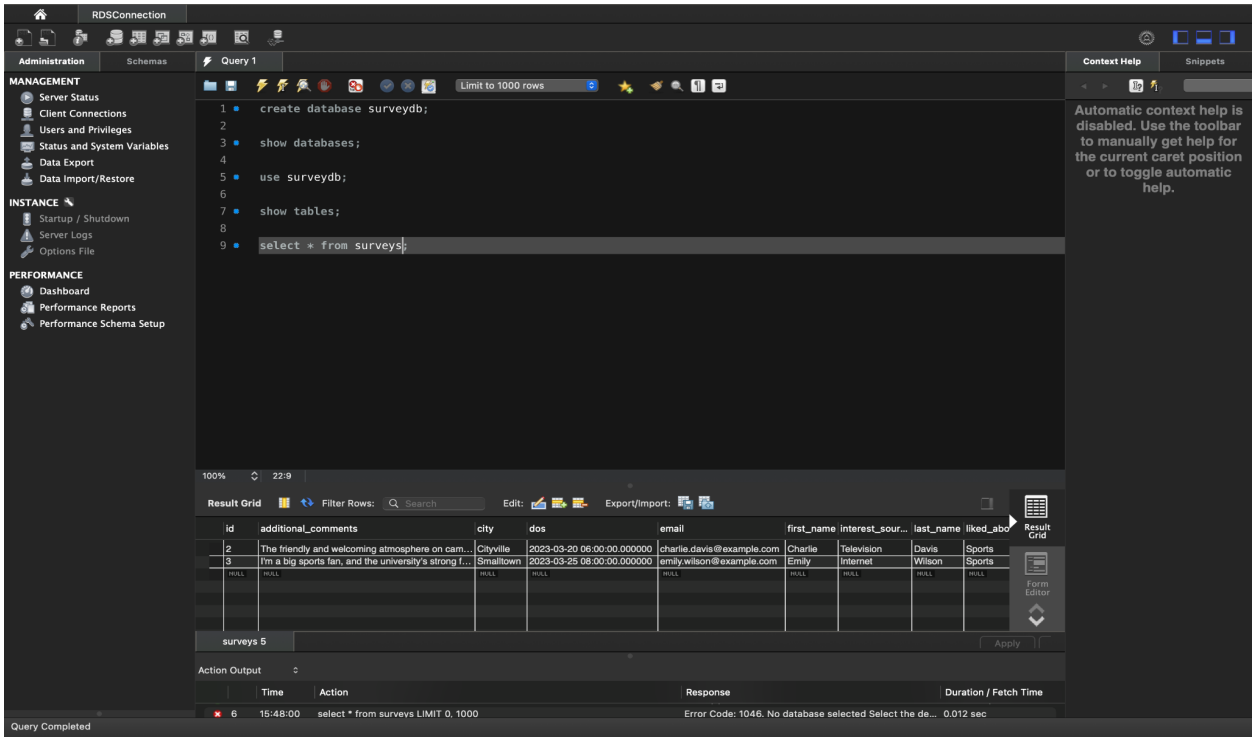
- Getting a particular survey using the survey id through the GET method.



- Deleting a survey based on the survey id with DELETE method.

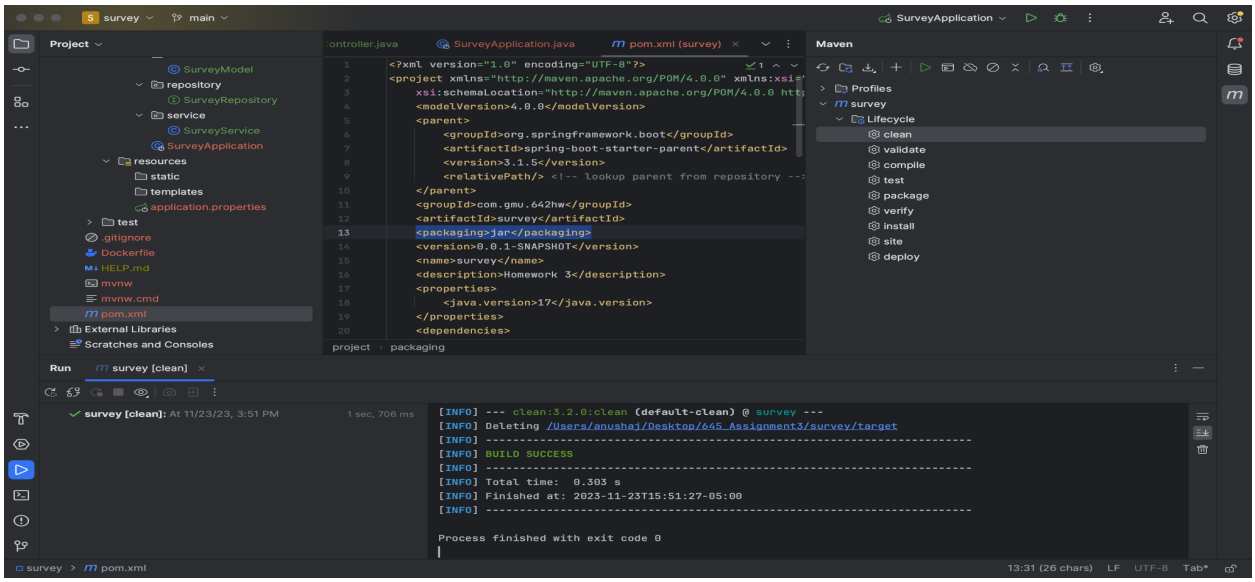


- Checking if data is inserted into the database.

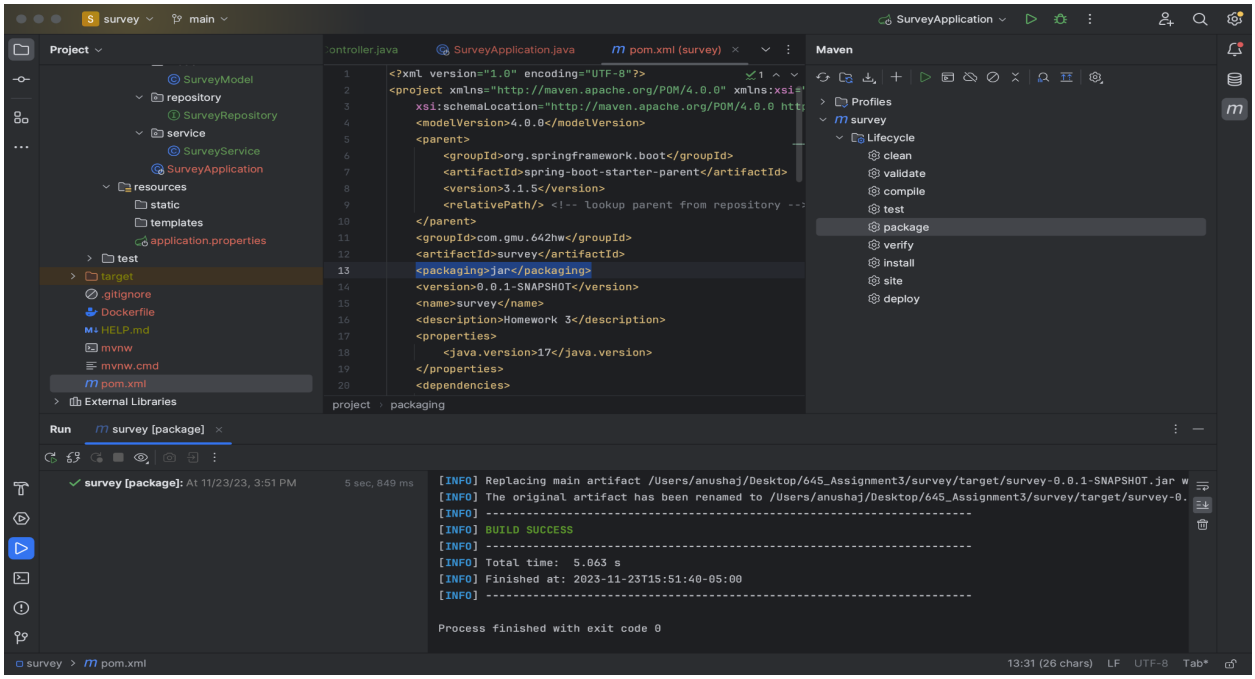


CREATING JAR

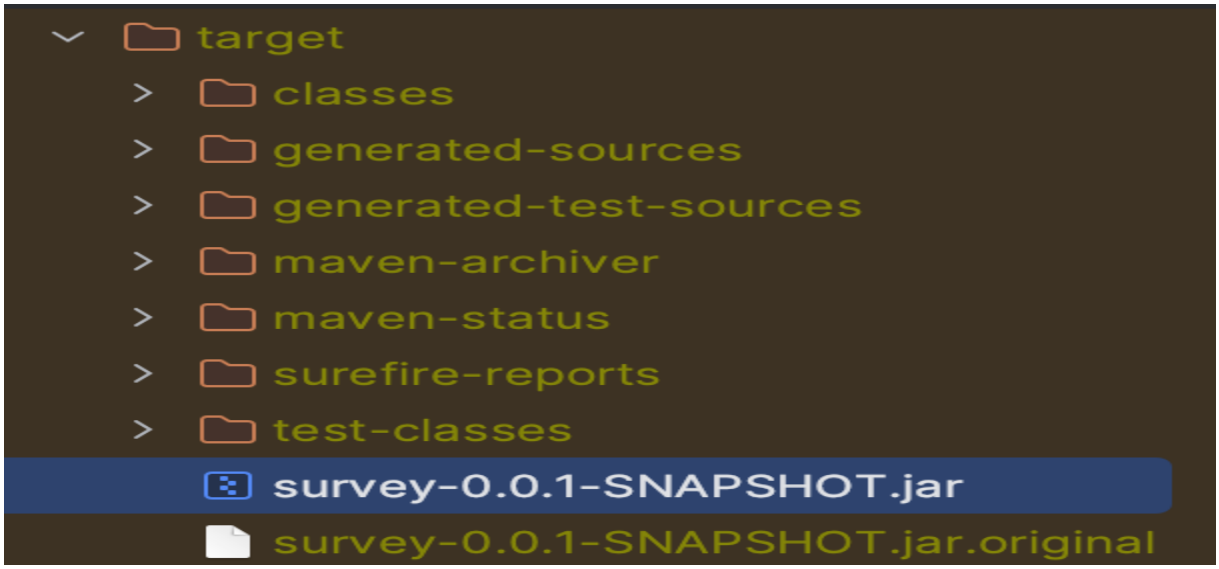
- Add <packaging> tag with in the pom.xml



- Then run mvn clean and mvn package

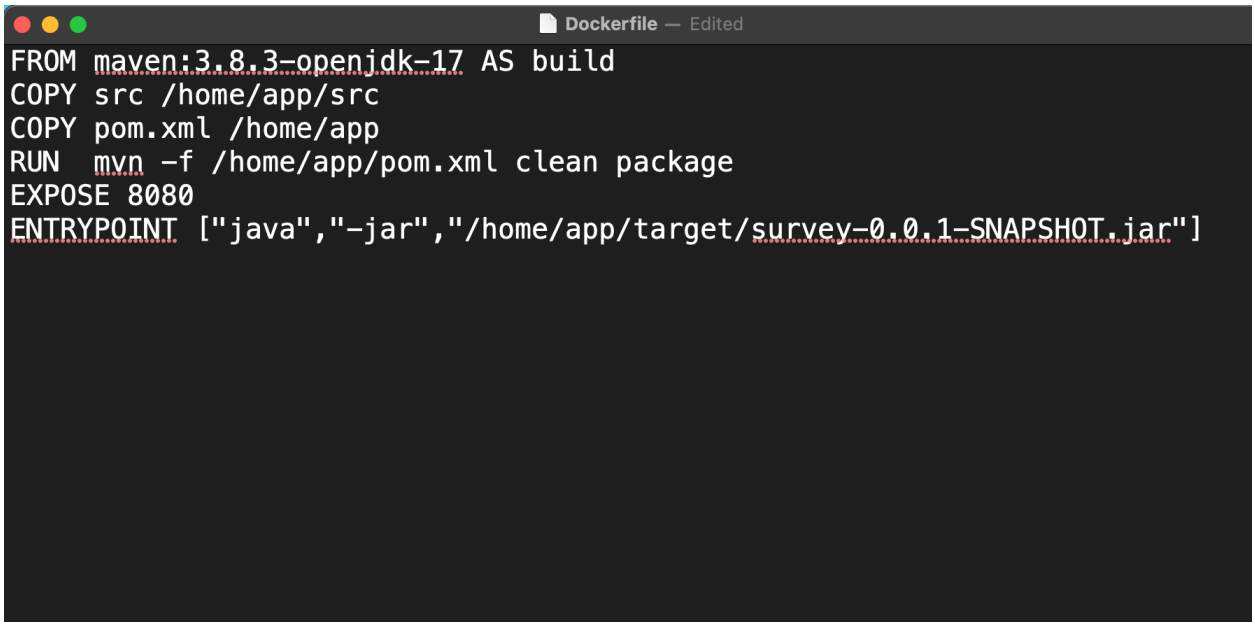


- A jar file will be created in the target folder

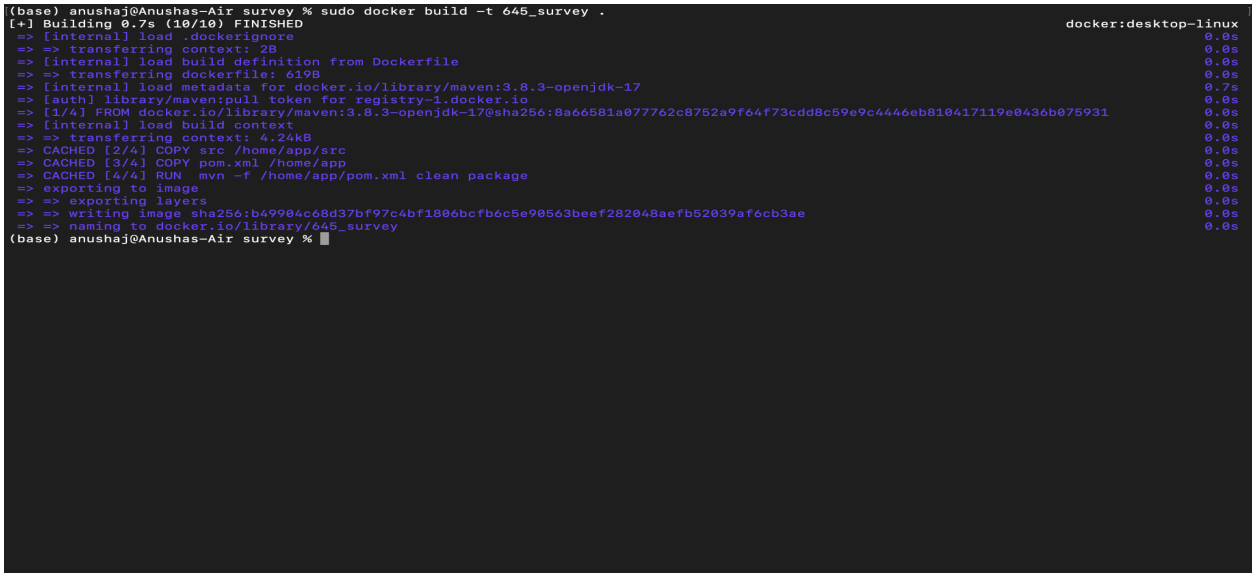


DOCKER

- Create a file named Dockerfile



- In the terminal in survey path, enter the commands
  sudo docker build –t 645_survey .

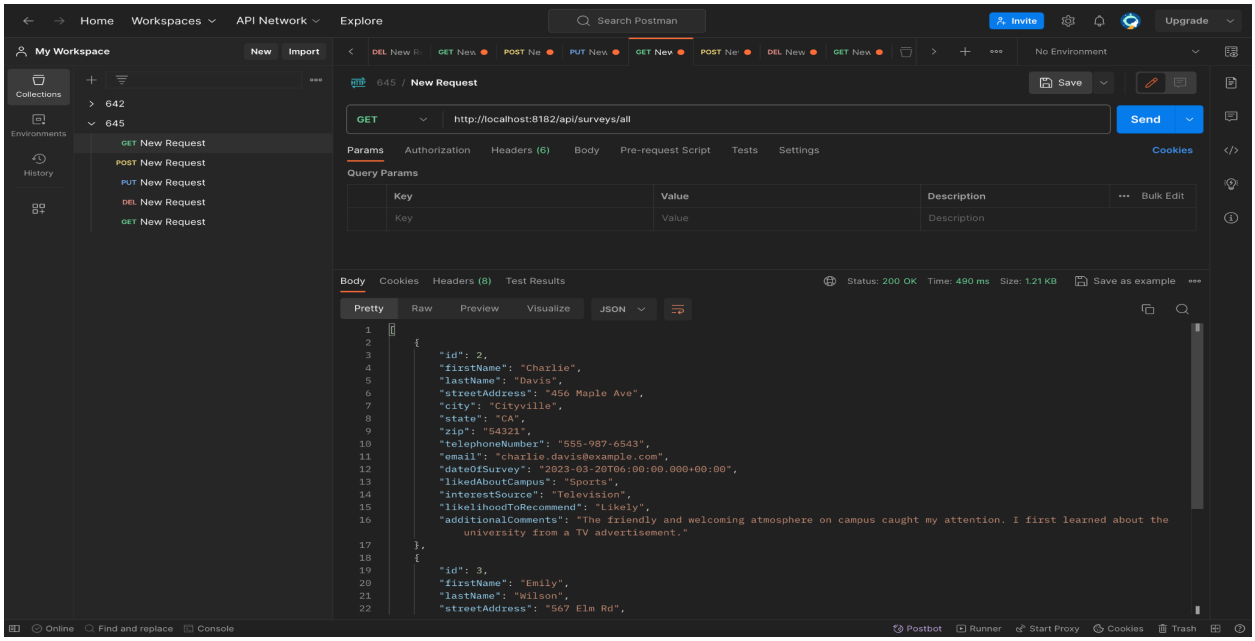


- docker run -it -p 8182:8080 645_survey.

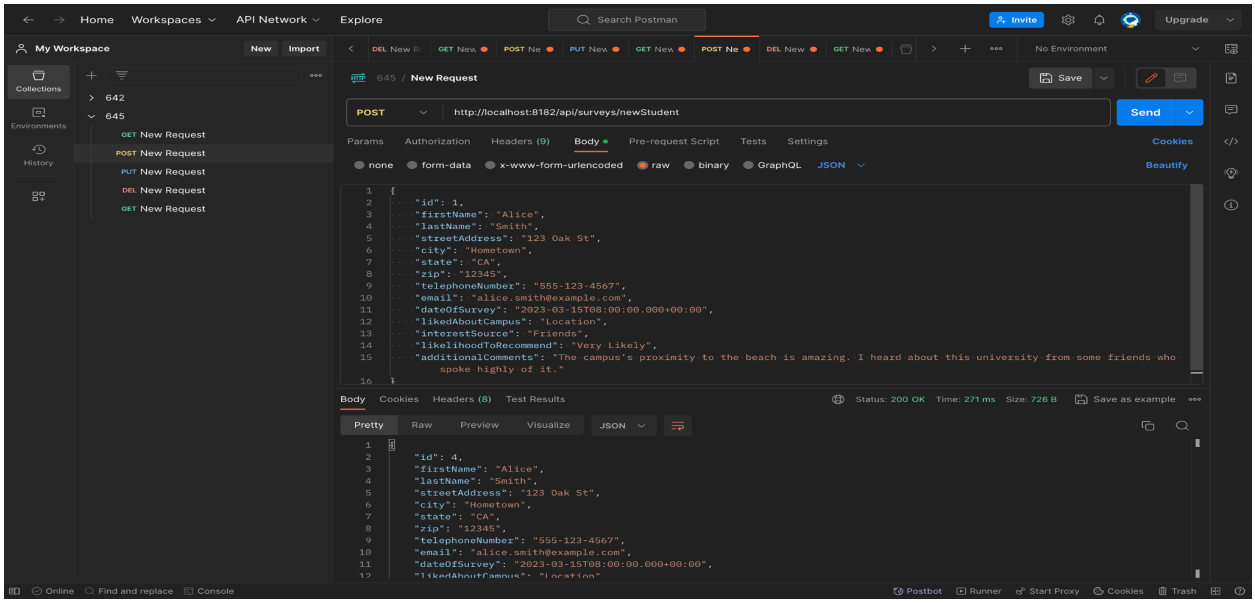
```
 => => exporting layers                                                                                                                                0.0s
 => => writing image sha256:b49904c68d37bf97c4bf1806bcfb6c5e90563beef282048aefb52039af6cb3ae                                                           0.0s
 => => naming to docker.io/library/645_survey                                                                                                          0.0s
(base) anushaj@Anushas-Air survey % docker run -it -p 8182:8080 645_survey

  .   ____          _            __ _ _
 /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
 \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
  '  |____| .__|_| |_|_| |_\__, | / / / /
 =========|_|==============|___/=/_/_/_/
 :: Spring Boot ::                (v3.1.5)

2023-11-23T20:59:42.417Z  INFO 1 --- [           main] com.gmu.hw.survey.SurveyApplication      : Starting SurveyApplication v0.0.1-SNAPSHOT us
ing Java 17.0.1 with PID 1 (/home/app/target/survey-0.0.1-SNAPSHOT.jar started by root in /)
2023-11-23T20:59:42.419Z  INFO 1 --- [           main] com.gmu.hw.survey.SurveyApplication      : No active profile set, falling back to 1 defa
ult profile: "default"
2023-11-23T20:59:43.201Z  INFO 1 --- [           main] .s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JPA repositories in
 DEFAULT mode.
2023-11-23T20:59:43.241Z  INFO 1 --- [           main] .s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning in 3
2 ms. Found 1 JPA repository interfaces.
2023-11-23T20:59:43.578Z  INFO 1 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat initialized with port(s): 8080 (http)
2023-11-23T20:59:43.584Z  INFO 1 --- [           main] o.apache.catalina.core.StandardService   : Starting service [Tomcat]
2023-11-23T20:59:43.584Z  INFO 1 --- [           main] o.apache.catalina.core.StandardEngine    : Starting Servlet engine: [Apache Tomcat/10.1.
15]
2023-11-23T20:59:43.626Z  INFO 1 --- [           main] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring embedded WebApplicationCo
ntext
2023-11-23T20:59:43.626Z  INFO 1 --- [           main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization co
mpleted in 1163 ms
2023-11-23T20:59:43.729Z  INFO 1 --- [           main] o.hibernate.jpa.internal.util.LogHelper  : HHH000204: Processing PersistenceUnitInfo [na
me: default]
2023-11-23T20:59:43.760Z  INFO 1 --- [           main] org.hibernate.Version                    : HHH000412: Hibernate ORM core version 6.2.13.
Final
2023-11-23T20:59:43.762Z  INFO 1 --- [           main] org.hibernate.cfg.Environment            : HHH000406: Using bytecode reflection optimize
r
2023-11-23T20:59:43.898Z  INFO 1 --- [           main] o.s.o.j.p.SpringPersistenceUnitInfo      : No LoadTimeWeaver setup: ignoring JPA class t
ransformer
2023-11-23T20:59:43.928Z  INFO 1 --- [           main] com.zaxxer.hikari.HikariDataSource       : HikariPool-1 - Starting...
2023-11-23T20:59:44.320Z  INFO 1 --- [           main] com.zaxxer.hikari.pool.HikariPool        : HikariPool-1 - Added connection com.mysql.cj.
jdbc.ConnectionImpl@4e868ef5
2023-11-23T20:59:44.322Z  INFO 1 --- [           main] com.zaxxer.hikari.HikariDataSource       : HikariPool-1 - Start completed.
2023-11-23T20:59:44.837Z  INFO 1 --- [           main] o.h.e.t.j.p.i.JtaPlatformInitiator       : HHH000489: No JTA platform available (set 'hi
bernate.transaction.jta.platform' to enable JTA platform integration)
2023-11-23T20:59:44.886Z  INFO 1 --- [           main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for pers
```

POSTMAN WITH DOCKER

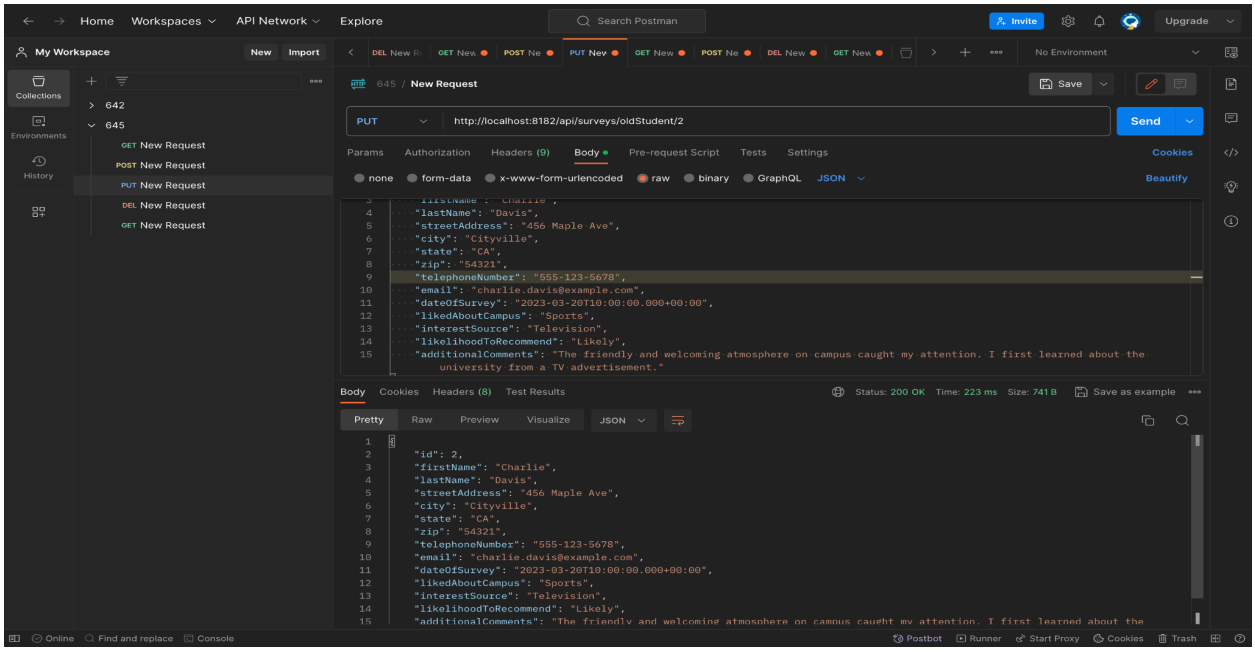
- Retrieving all the surveys from the GET method.



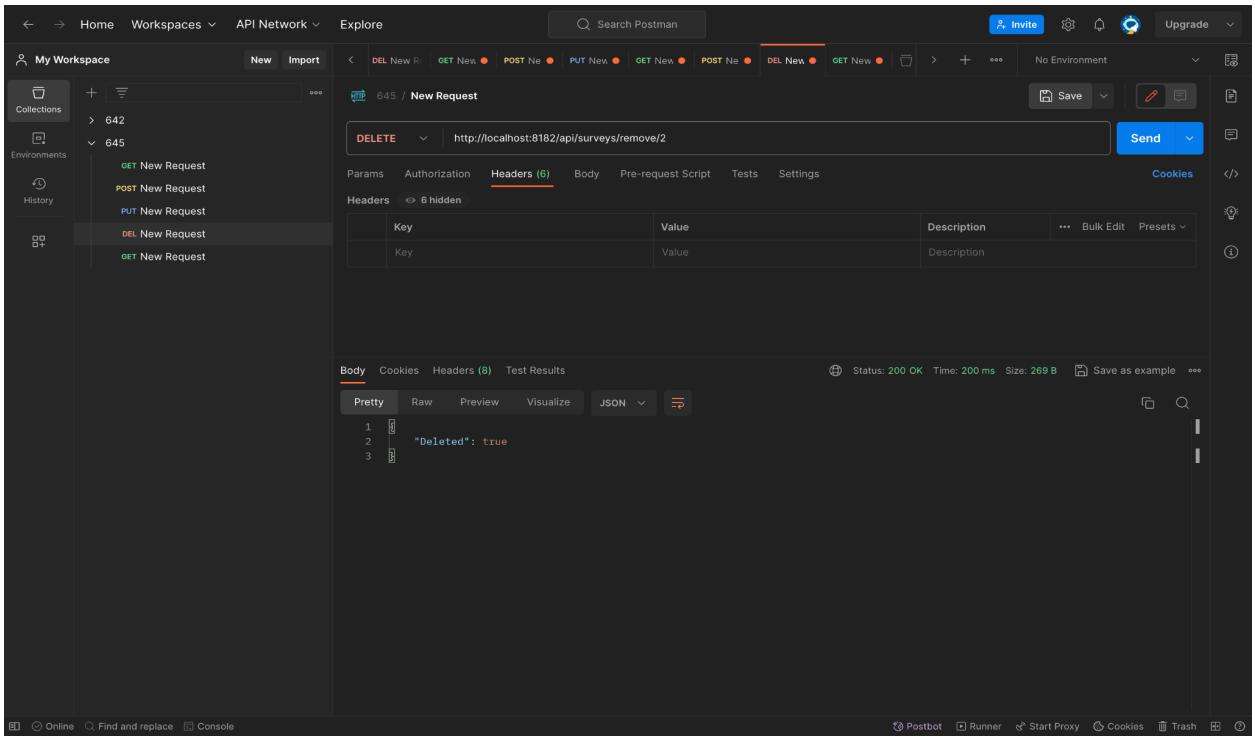
- Creating a new survey using the POST method.



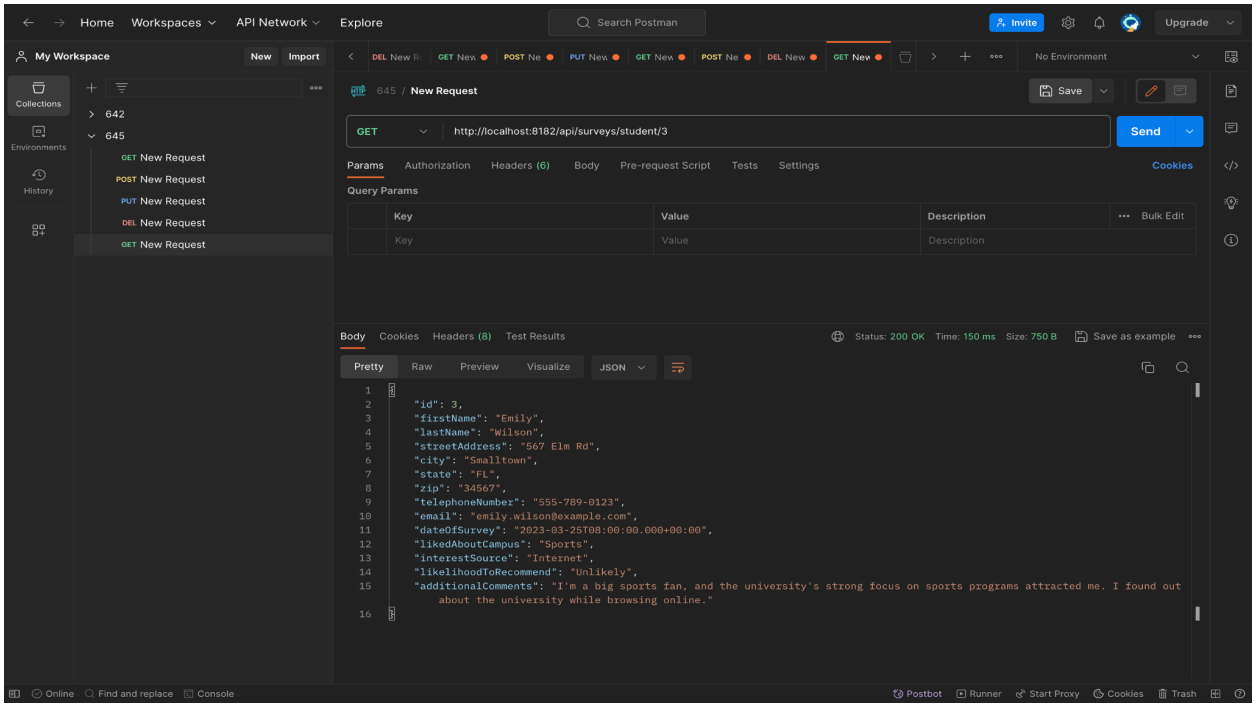
- Updating a particular survey details using the UPDATE method.



- Deleting a survey based on the survey id with DELETE method.



- Getting a particular survey using the survey id through the GET method.

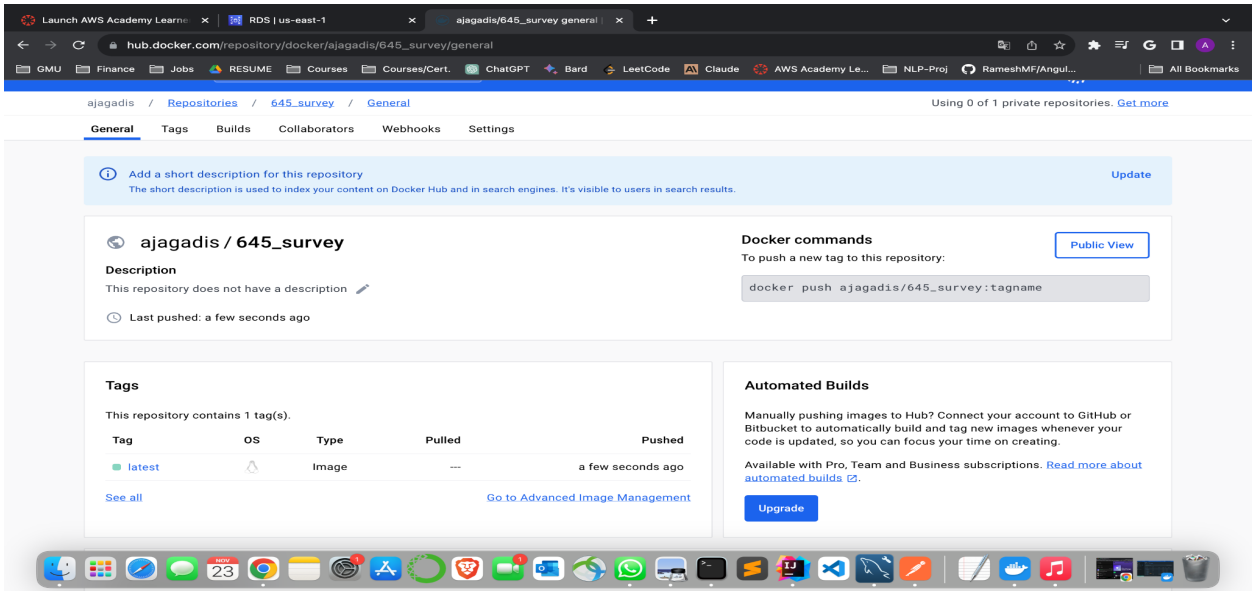


DOCKER IMAGE

Run the below commands to push the image to docker

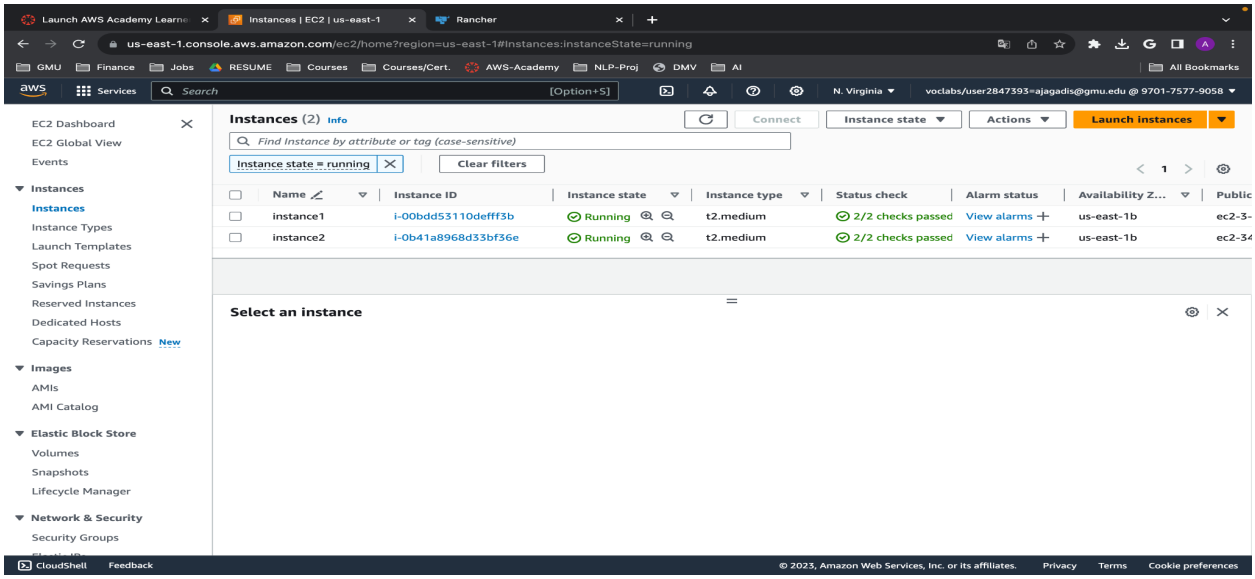
- docker tag 645_survey ajagadis/645_survey
- docker push ajagadish/645_survey

```
(base) anushaj@Anushas-Air survey % docker tag 645_survey ajagadis/645_survey
(base) anushaj@Anushas-Air survey % docker push ajagadis/645_survey
Using default tag: latest
The push refers to repository [docker.io/ajagadis/645_survey]
5539dac4206e: Pushed
88515675eec9: Pushed
0a9e5de54a13: Pushed
366e0ac3b92a: Mounted from library/maven
48f9189d20fc: Mounted from library/maven
eaab9d11cc5d: Mounted from library/maven
4da0bd95a2ee: Mounted from library/maven
3ea4fe4d8edd: Mounted from library/maven
6ba7dd301dbf: Mounted from library/maven
aac6d61109c4: Mounted from library/maven
latest: digest: sha256:9ff6b8a8b2e520a8bec8ef4f2e216f2aabc56618717b6391ebb1dfc8f7107407 size: 2420
(base) anushaj@Anushas-Air survey %
```

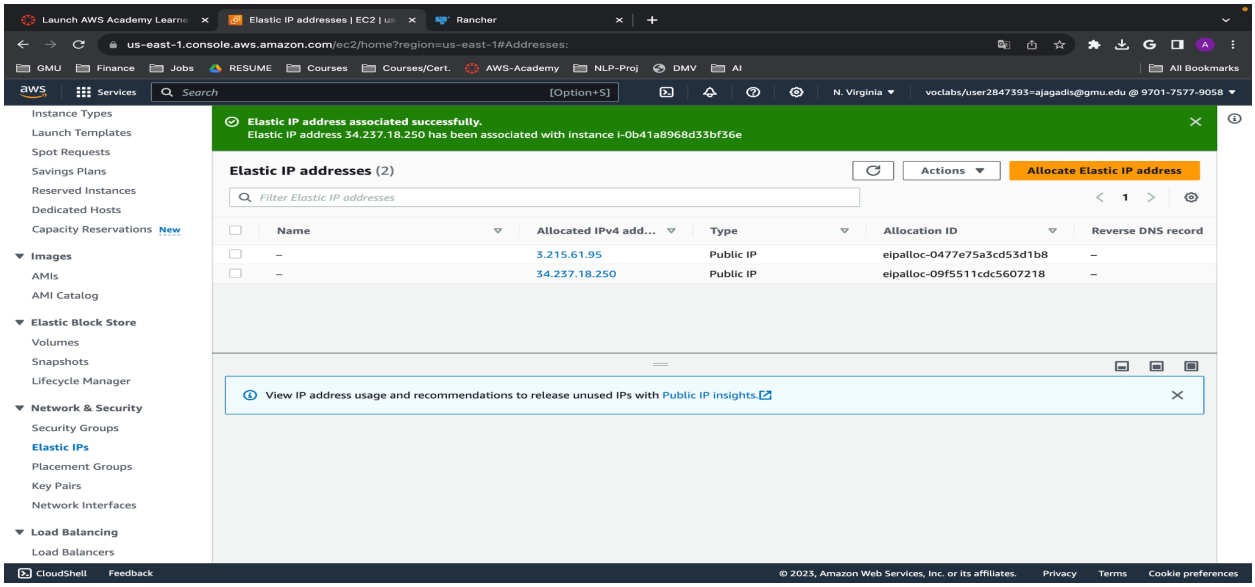


EC2 INSTANCE

- Create two EC2 instances by choosing Ubuntu from the AMI list by choosing t2.medium from the Instance types available.



- Now allocate the elastic ip address with the instance.



- Connect to EC2 instance enter the following commands
  a) sudo apt-get update

```
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-89-110:~$ sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 c-n-f Metadata [286 k
B]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [217 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse Translation-en [112 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 c-n-f Metadata [83
72 B]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1200 k
B]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [252 kB
]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [
16.1 kB]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [
1168 kB]
Get:15 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [985 kB]
Get:16 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [192 kB]
Get:17 http://security.ubuntu.com/ubuntu jammy-security/main amd64 c-n-f Metadata [11.4 kB]
Get:18 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [1134 kB]
Get:19 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [185 kB]
```

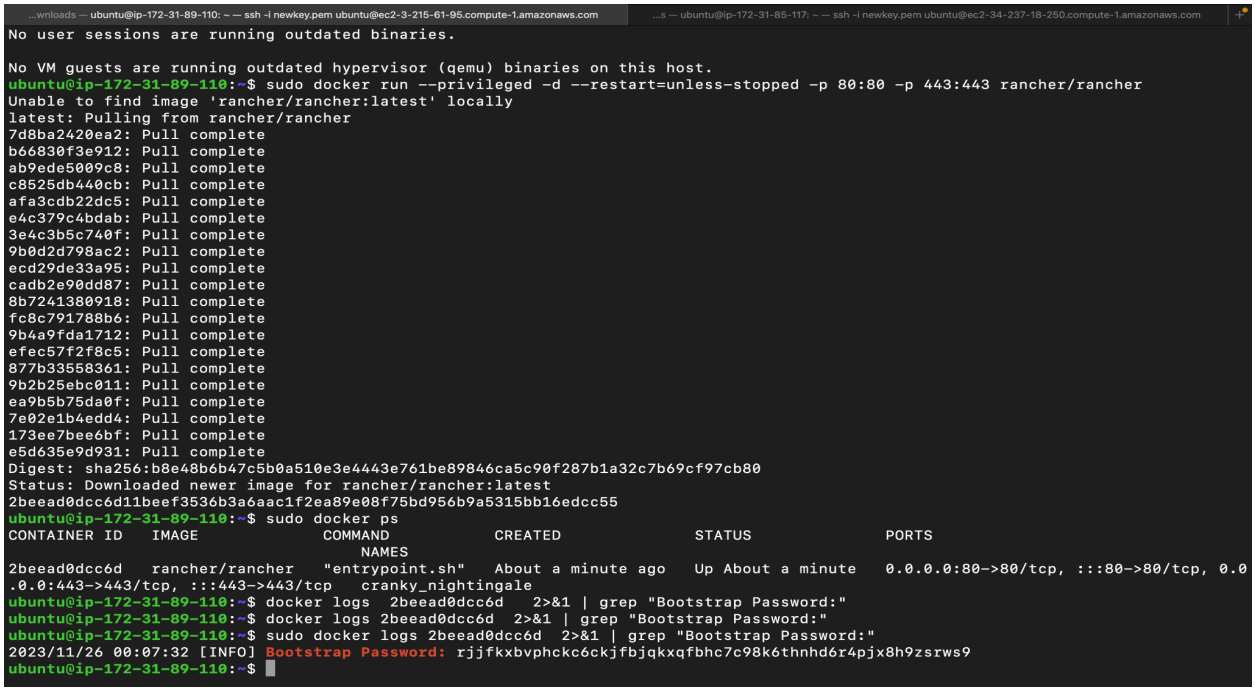
b) sudo apt install docker.io
c) docker -v

To deploy Rancher navigate to the rancher website and copy the command:
https://www.rancher.com/quick-start

```
Reading package lists... Done
ubuntu@ip-172-31-89-110:~$ sudo apt install docker.io
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  bridge-utils containerd dns-root-data dnsmasq-base pigz runc ubuntu-fan
Suggested packages:
  ifupdown aufs-tools cgroupfs-mount | cgroup-lite debootstrap docker-doc rinse zfs-fuse
  | zfsutils
The following NEW packages will be installed:
  bridge-utils containerd dns-root-data dnsmasq-base docker.io pigz runc ubuntu-fan
0 upgraded, 8 newly installed, 0 to remove and 62 not upgraded.
Need to get 69.7 MB of archives.
After this operation, 267 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 pigz amd64 2.6-1 [63.6 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 bridge-utils amd64 1.7-1ubuntu3 [34.4 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 runc amd64 1.1.7-0ubuntu1~22.04.1 [4249 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 containerd amd64 1.7.2-0ubuntu1~22.04.1 [36.0 MB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 dns-root-data all 2021011101 [5256 B]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 dnsmasq-base amd64 2.86-1.1ubuntu0.3 [354 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 docker.io amd64 24.0.5-0ubuntu1~22.04.1 [28.9 MB
]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 ubuntu-fan all 0.12.16 [35.2 kB]
Fetched 69.7 MB in 2s (28.9 MB/s)
Preconfiguring packages ...
Selecting previously unselected package pigz.
(Reading database ... 64726 files and directories currently installed.)
Preparing to unpack .../0-pigz_2.6-1_amd64.deb ...
Unpacking pigz (2.6-1) ...
Selecting previously unselected package bridge-utils.
Preparing to unpack .../1-bridge-utils_1.7-1ubuntu3_amd64.deb ...
Unpacking bridge-utils (1.7-1ubuntu3) ...
Selecting previously unselected package runc.
Preparing to unpack .../2-runc_1.1.7-0ubuntu1~22.04.1_amd64.deb ...
Unpacking runc (1.1.7-0ubuntu1~22.04.1) ...
Selecting previously unselected package containerd.
Preparing to unpack .../3-containerd_1.7.2-0ubuntu1~22.04.1_amd64.deb ...
```

d) To launch the rancher as a docker container run the command

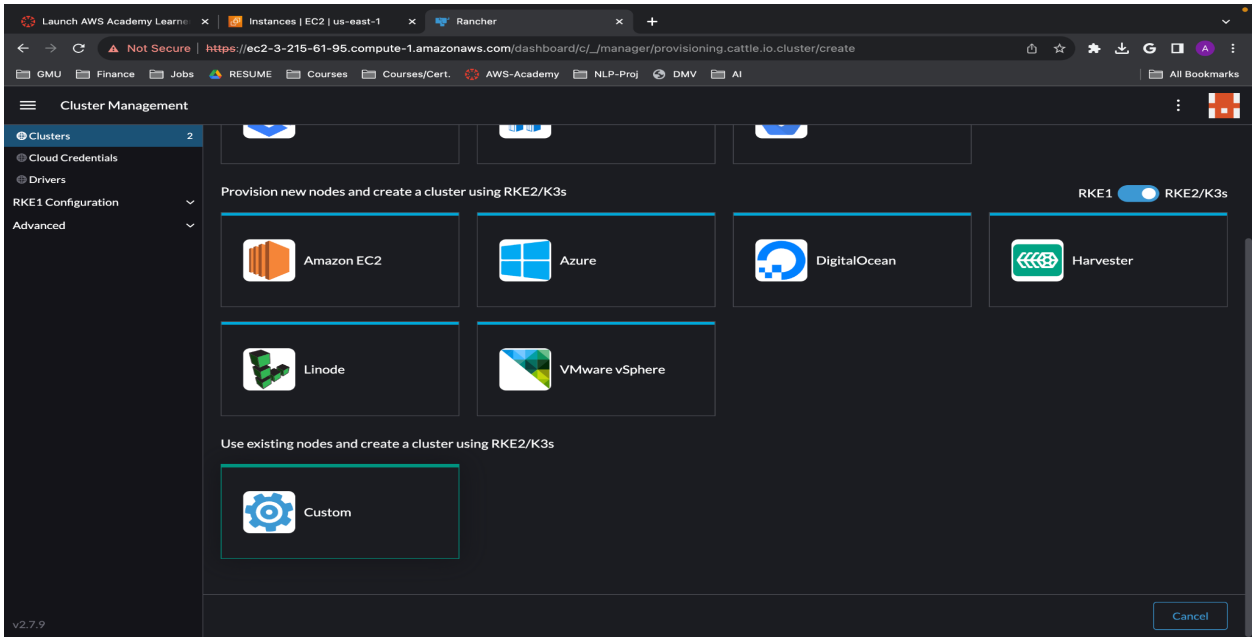
docker run --privileged -d --restart=unless-stopped -p 80:80 -p 443:443 rancher/rancher



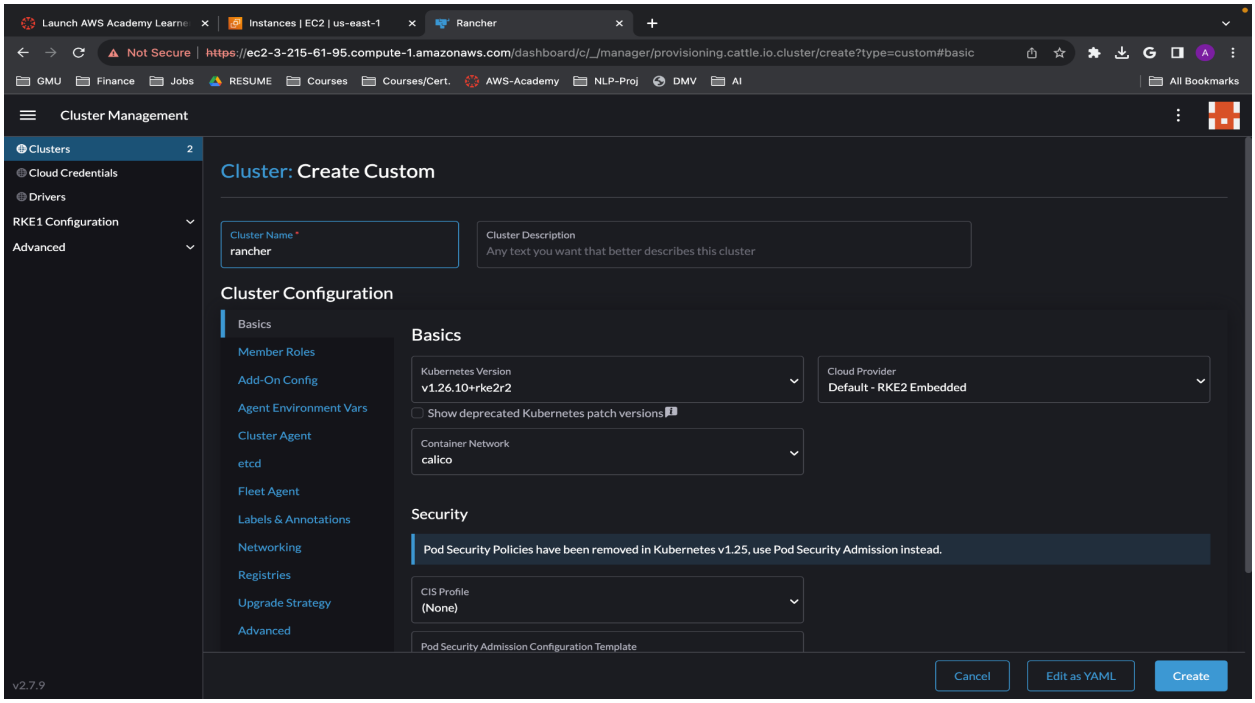
- Next step,to get the container id of the Docker run “ sudo docker ps ” command
  And then copy the container id to get the bootstrap password of the Rancher
  sudo docker logs d17e192c92da 2>&1 | grep "Bootstrap Password:"
- Now, copy the password obtained and login, now we can create a new password and launch the rancher.

CLUSTER CREATION

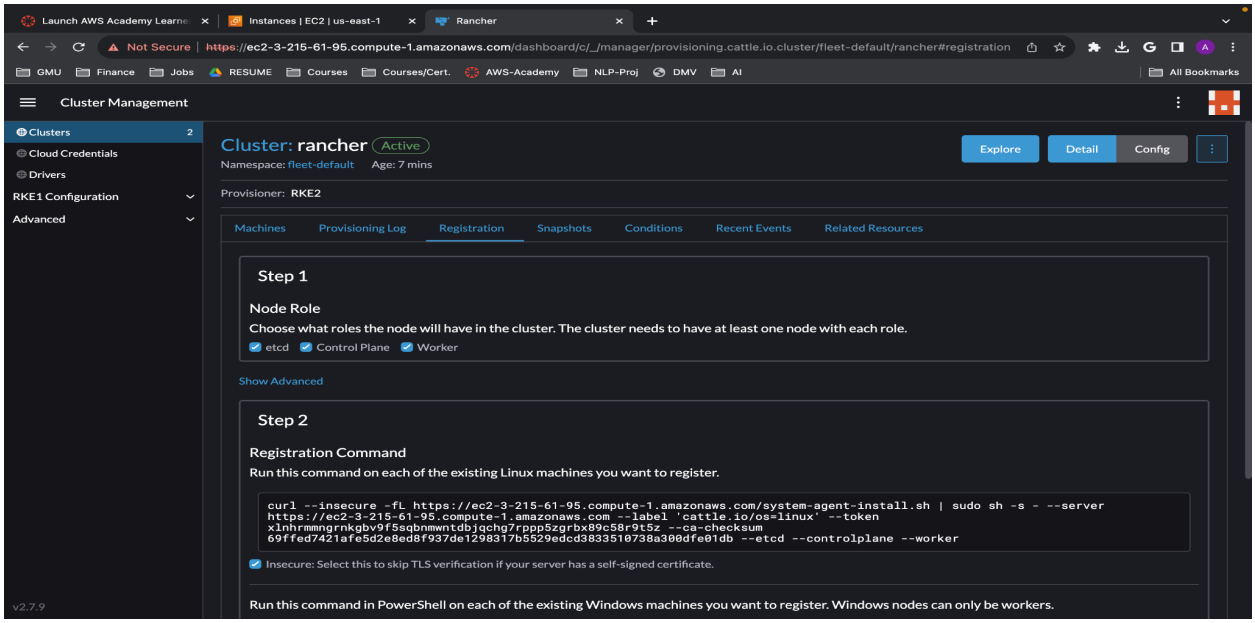
- In the Rancher dashboard click on create, select custom as the option



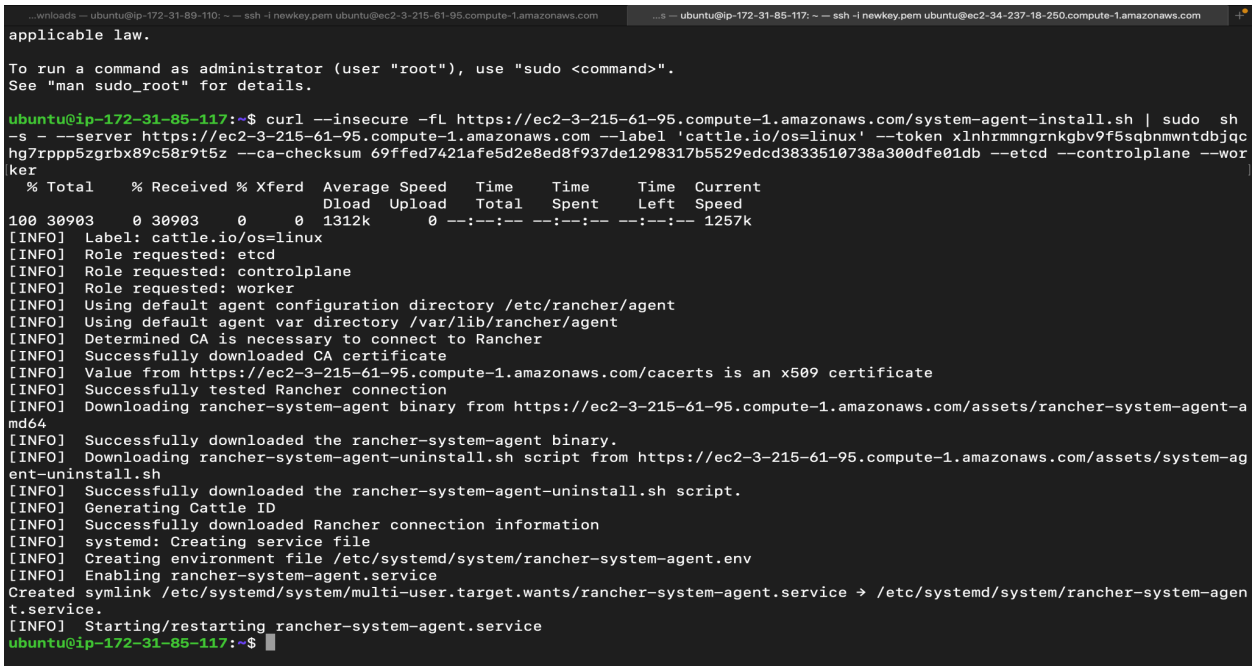
- Give a name to the cluster and then create



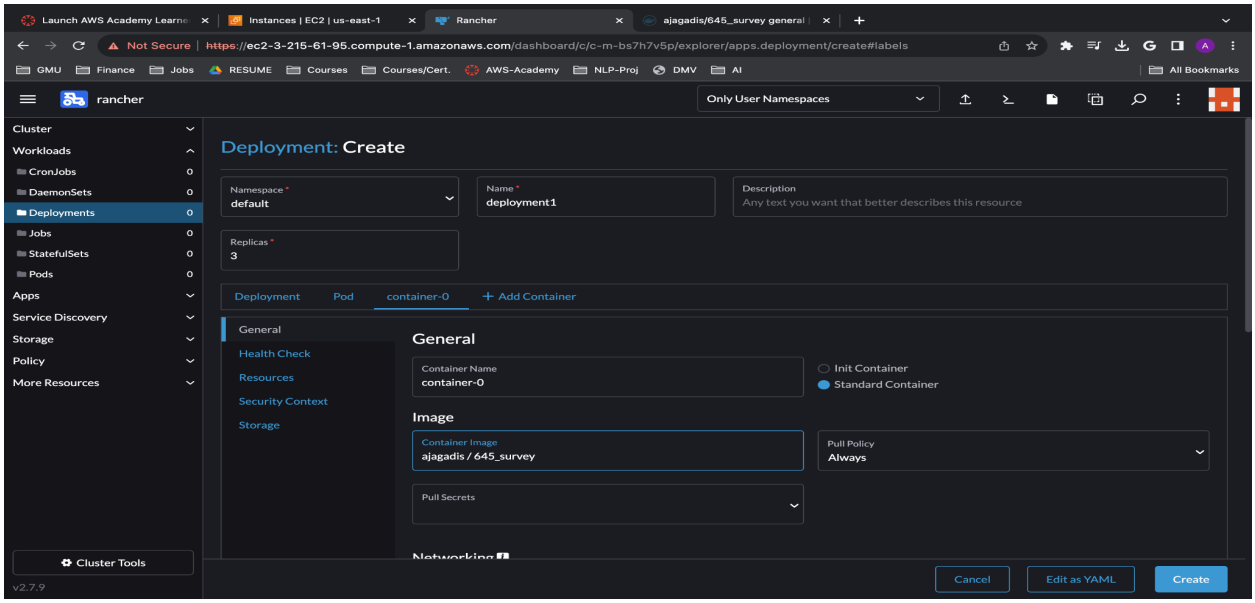
- After some time the clusters will be active.

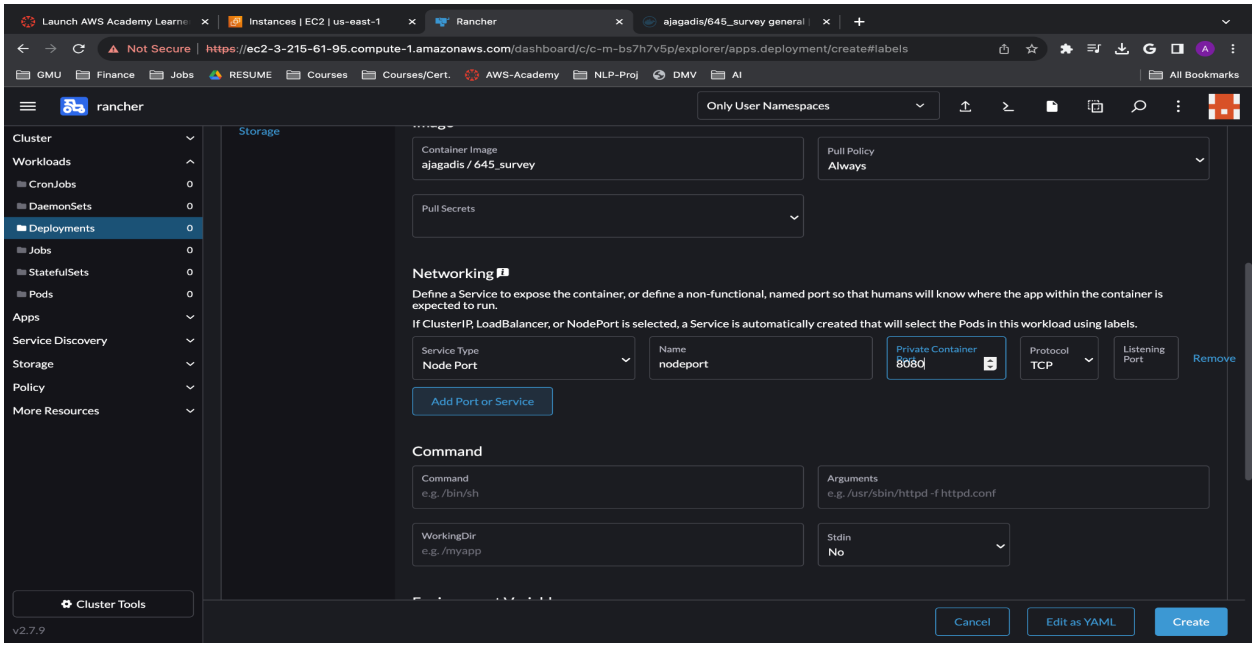


- After creation, copy the curl command and run it in the second EC2 instance where rancher has not been deployed.



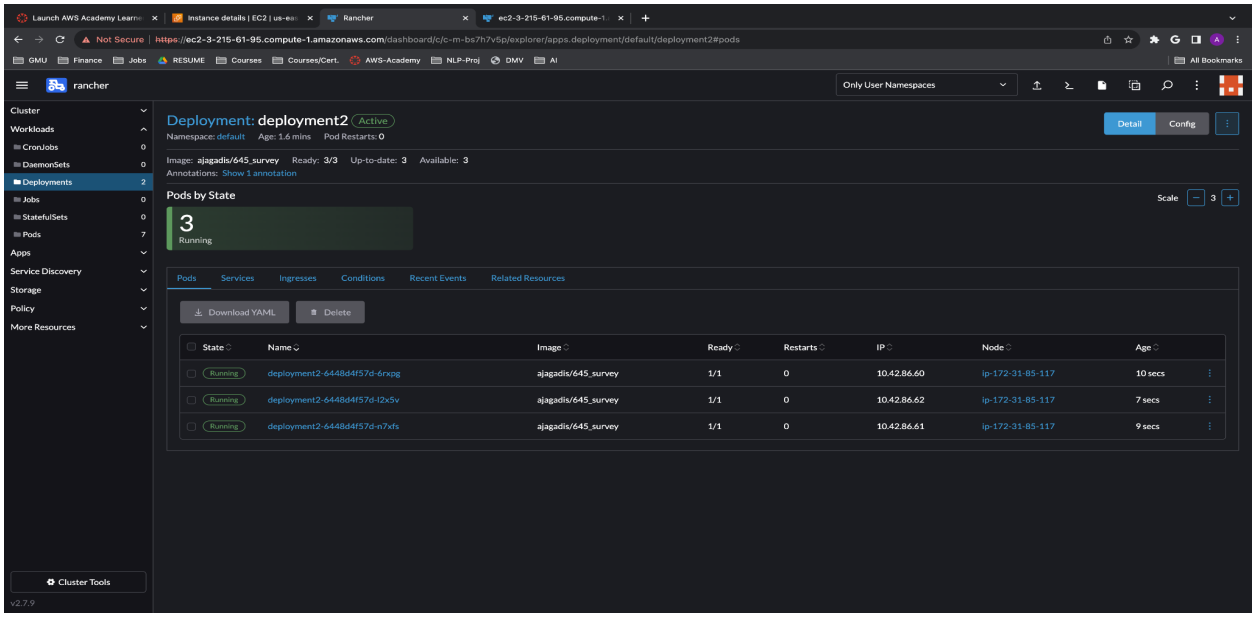
- Now open the cluster, and then open workload tab and open deployment tab, then create a deployment with three replicas and the docker image and add nodeport in networking.



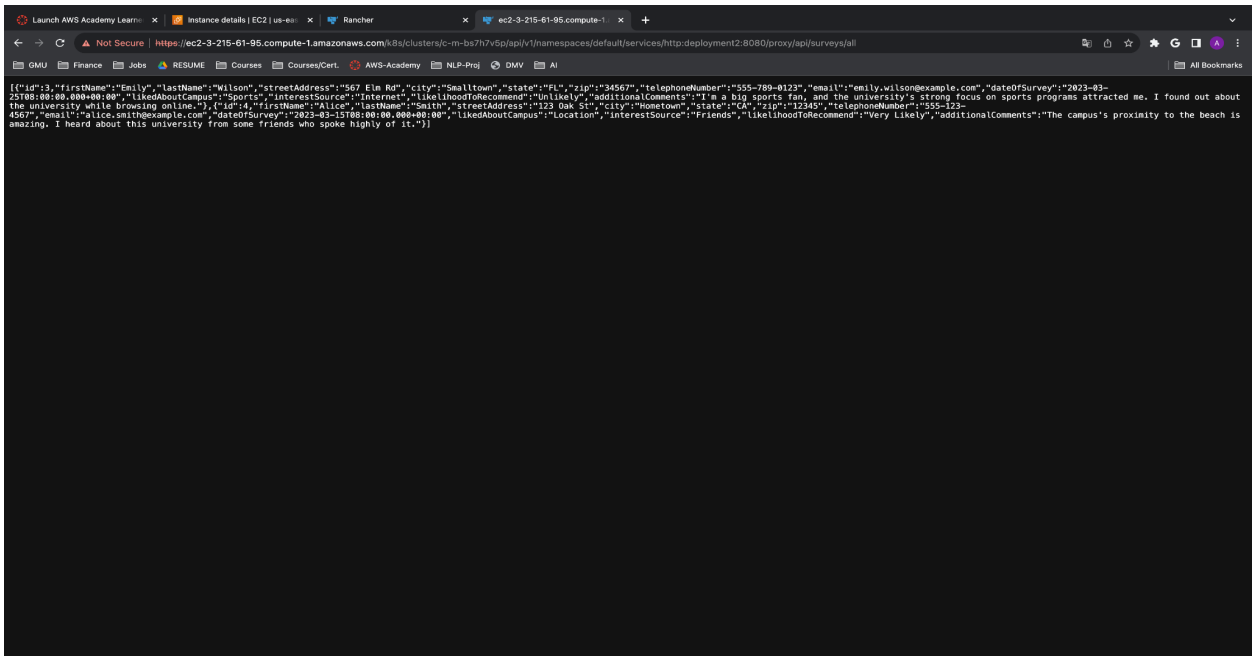


- The nodeport deployment can be accessed using the below url. https://ec2-23-20-134-74.compute-1.amazonaws.com/k8s/clusters/c-m-hrw8c9zn/api/v1/namespaces/default /services/http:deploy:8080/proxy/swe/

- After some time the deployment will be active.

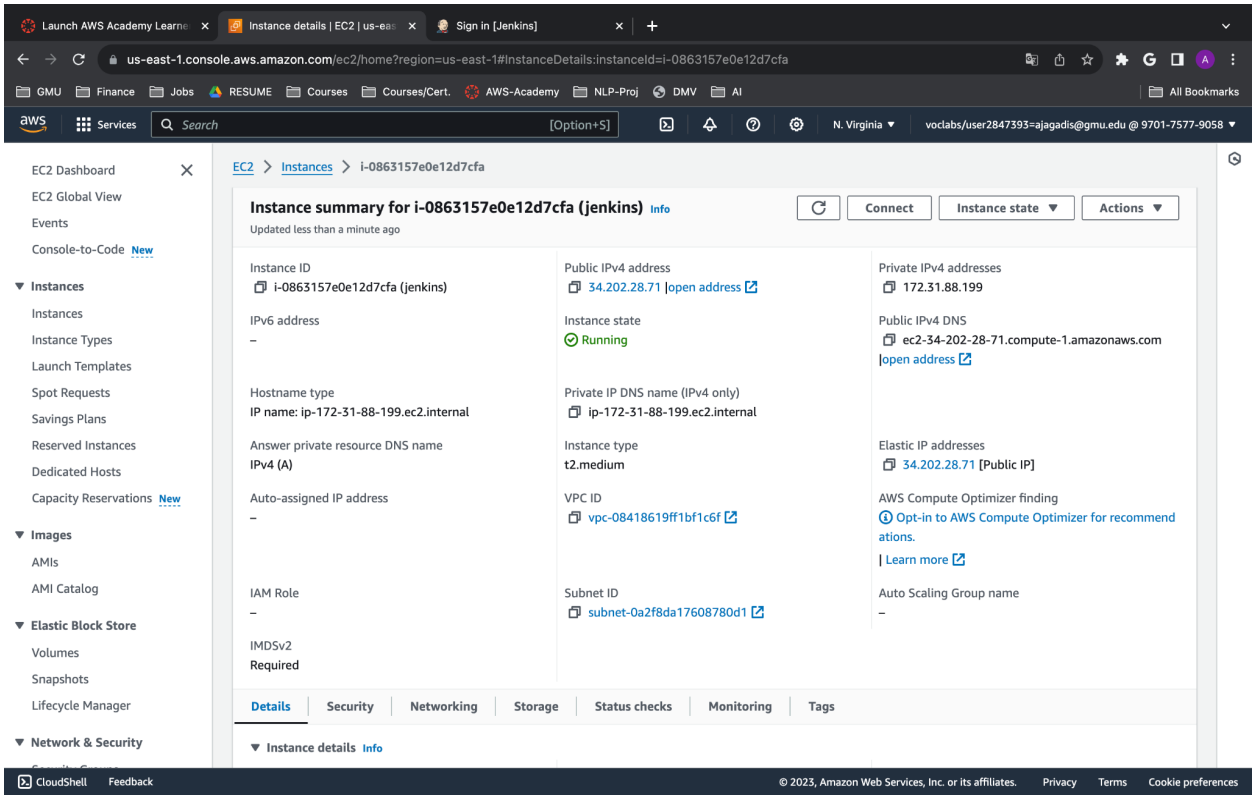


- By following the previous method create one more deployment for load balancer and after activation it can be accessed using the url. https://ec2-23-20-134-74.compute-1.amazonaws.com/k8s/clusters/c-m-hrw8c9zn/api/v1/namespaces/default /services/http:deploy-lb:8080/proxy/api/surveys/all



JENKINS

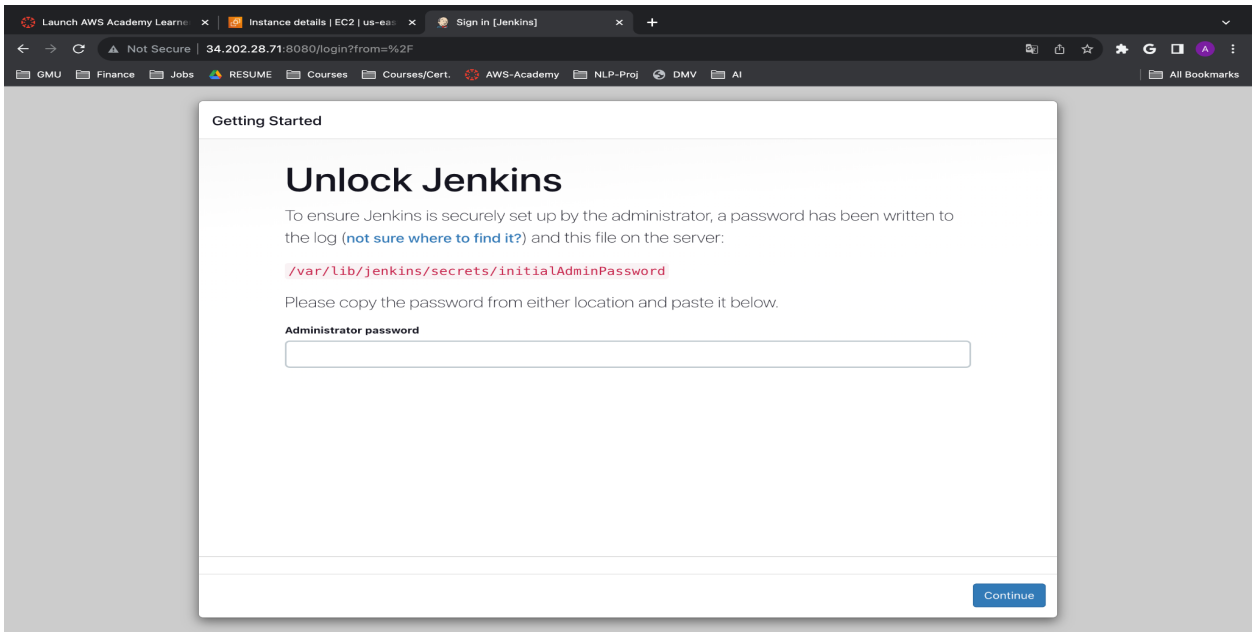
- Launch an EC2 instance and pick Ubuntu from the Amazon Machine Images (AMI) catalog
- Choose the t2.medium instance type for computing capacity and memory



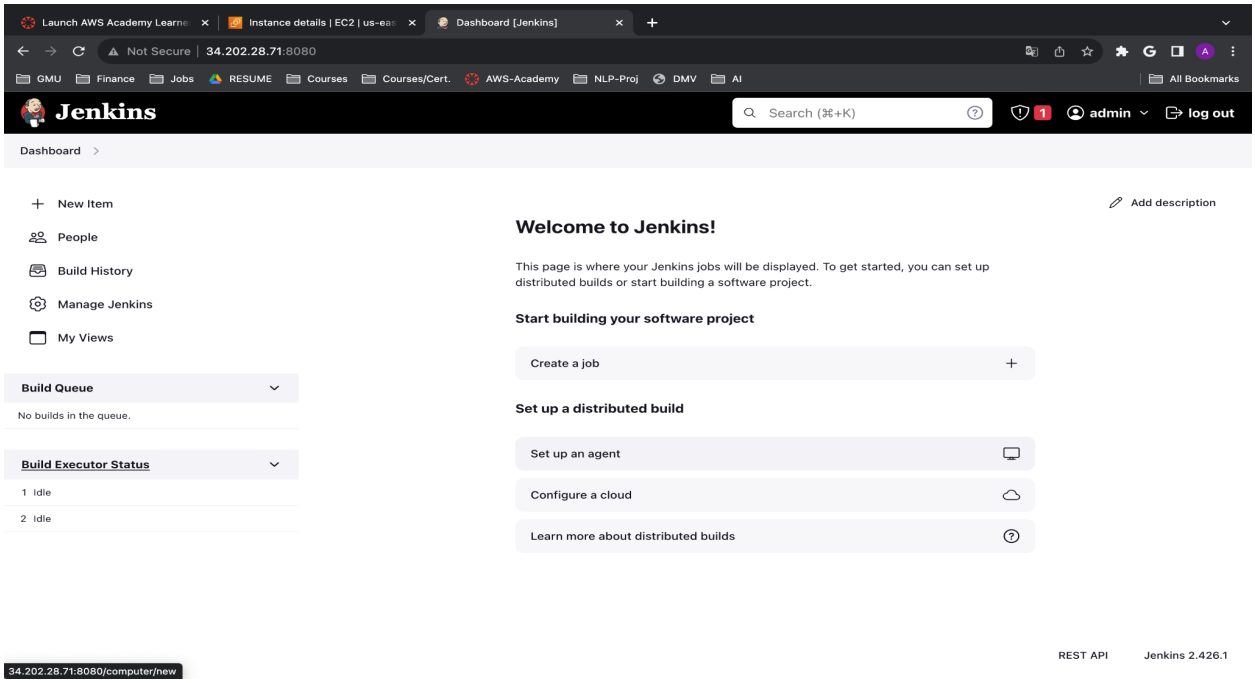
- Connect to the EC2 instance and enter the following commands given below:
  a) sudo apt-get update
  b) sudo apt install docker.io
  c) Verify that docker is installed by running the command **docker -v**
  d) curl -fsSL https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key | sudo tee\
  /usr/share/keyrings/jenkins-keyring.asc > /dev/null
  e) echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \
  https://pkg.jenkins.io/debian-stable binary/ | sudo tee \
  /etc/apt/sources.list.d/jenkins.list > /dev/null
  f) sudo apt-get update
  g) sudo apt install openjdk-11-jdk
  h) sudo apt-get update
  i) sudo apt-get install jenkins
  f) Enter the command systemctl status jenkins and check the status of Jenkins

```
ubuntu@ip-172-31-88-199:~$ systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
     Loaded: loaded (/lib/systemd/system/jenkins.service; enabled; vendor preset: enabled)
     Active: active (running) since Wed 2023-11-29 00:58:37 UTC; 11min ago
   Main PID: 5880 (java)
      Tasks: 48 (limit: 4667)
     Memory: 1.2G
        CPU: 46.807s
     CGroup: /system.slice/jenkins.service
             └─5880 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/war

Nov 29 00:58:21 ip-172-31-88-199 jenkins[5880]: 21c495183e1a47a5b358f8bfa5ca17b1
Nov 29 00:58:21 ip-172-31-88-199 jenkins[5880]: This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword
Nov 29 00:58:21 ip-172-31-88-199 jenkins[5880]: ************************************************************
Nov 29 00:58:21 ip-172-31-88-199 jenkins[5880]: ************************************************************
Nov 29 00:58:21 ip-172-31-88-199 jenkins[5880]: ************************************************************
Nov 29 00:58:37 ip-172-31-88-199 jenkins[5880]: 2023-11-29 00:58:37.226+0000 [id=33]        INFO        jenkins.InitReactorR
Nov 29 00:58:37 ip-172-31-88-199 jenkins[5880]: 2023-11-29 00:58:37.245+0000 [id=24]        INFO        hudson.lifecycle.Lif
Nov 29 00:58:37 ip-172-31-88-199 systemd[1]: Started Jenkins Continuous Integration Server.
Nov 29 00:58:37 ip-172-31-88-199 jenkins[5880]: 2023-11-29 00:58:37.290+0000 [id=49]        INFO        h.m.DownloadService$
Nov 29 00:58:37 ip-172-31-88-199 jenkins[5880]: 2023-11-29 00:58:37.290+0000 [id=49]        INFO        hudson.util.Retrier#
lines 1-20/20 (END)
```

- Once the Jenkins status is active go to the EC2 instance security and allow inbound traffic on port 8080
- Now, launch the Jenkins instance using the public IPv4 address: http://18.215.66.252:8080/.



- Login using the admin password and launch the Jenkins dashboard



- Open a terminal session on the EC2 server outside Jenkins.

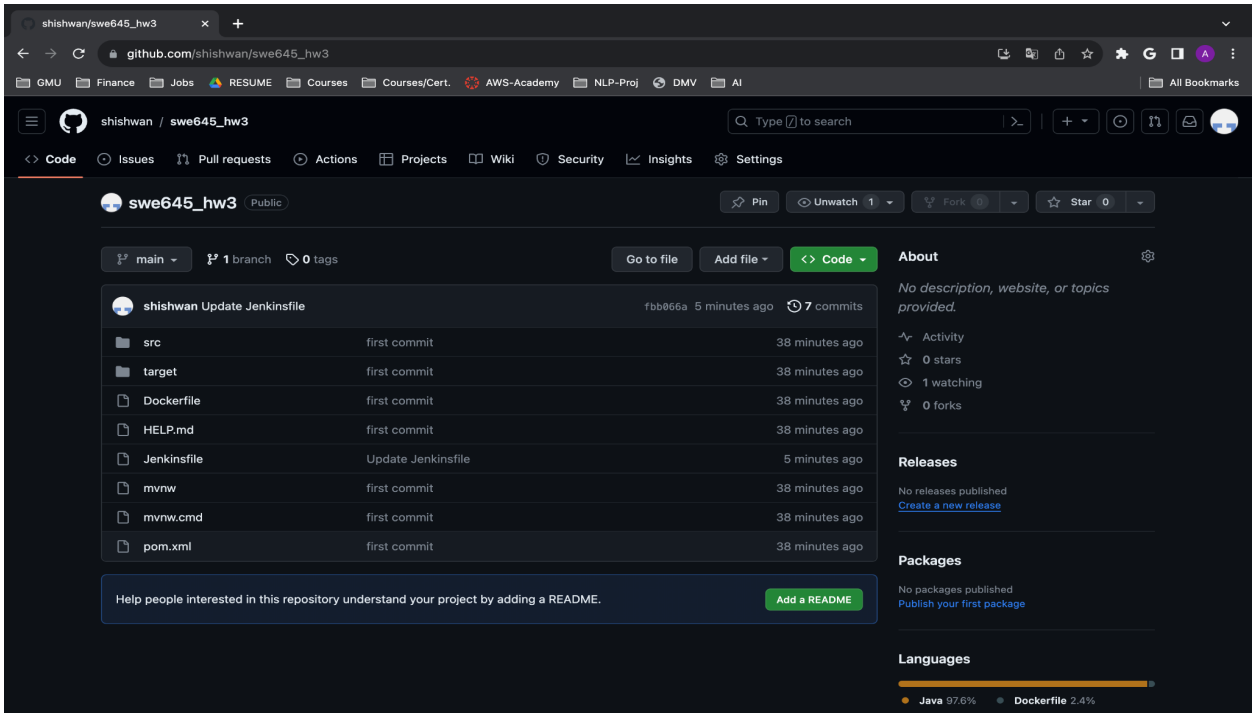
- Enter the commands to install kubectl for Kubernetes management.
  a) sudo apt install snapd
  b) sudo snap install kubectl –classi
  c) Login as Jenkins user : sudo su jenkin

- In Rancher, navigate to your cluster and download the KubeConfig file

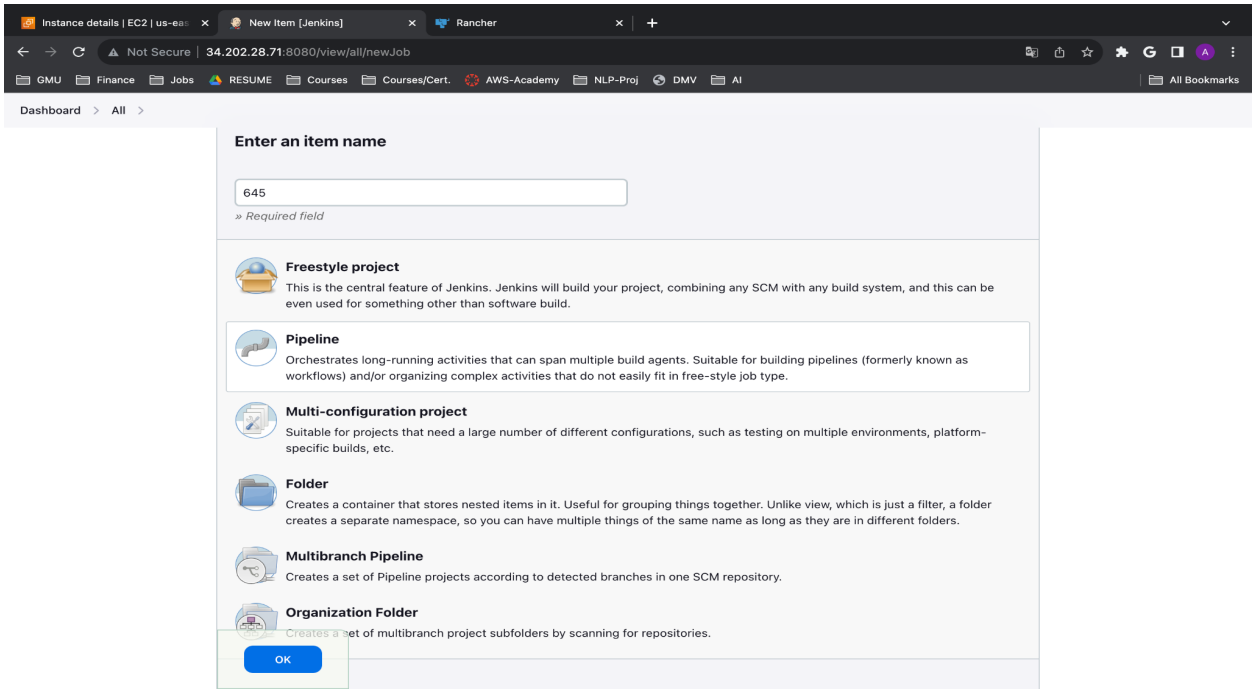
- Enter the following command to enable kubectl connection with our rancher cluster.
  a) Cd /vaar/lib/jenkins
  b) > .kube
  c) > config
  d) cat > config
  e) Copy the content from the KubeConfig file and paste it here and do ctrl + d.
  kubectl config current-context

```
apiVersion: v1
kind: Config
clusters:
- name: "rancher"
  cluster:
    server: "https://ec2-3-215-61-95.compute-1.amazonaws.com/k8s/clusters/c-m-bs7h7v5p"
    certificate-authority-data: "LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSUJ2VENDQ\
      VdPZ0F3SUJBZ0lCQURBS0JnZ3Foa2pPUFFRREFqQkdNUnd3R2dZRFZRUUtFeE5rZVc1aGJXbGoKY\
      kdsemRHVnVaWEl0YjNKbk1TWXdKQVlEVlFRRERCMWtlVzVoYldsamJHbHpkR1Z1WlhJdFkyRkFNV\
      GN3TURrMQpOekkxTkRBZUZ3MHlNekV4TWpZd01EQTNNelJhRncwek16RXhNak13TURBM016UmFNR\
      Vl4SERBYUJnTlZCQW9UCkUyUjVibUZ0YVdOc2FYTjBaVzVsY2kxdmNtY3hKakFrQmdOVkJBTU1IV\
      1I1Ym1GdGFXTnNhWE4wWlc1bGNpMWoKWVVBeE56QXdPVFUzTWpVME1Ga3dFd1lIS29aSXpqMENBU\
      VlJS29aSXpqMERBUWNEUWdBRTVtQnRMQ01ZSDhiWAorb0JjRG5nckI4MDJsaEJhOUFVN205NVF6d\
      GluZUh3Mzl6RWt2dzNYdnMwZ0h2aFhuMkFlREMvZVhaYktCeW95CnVBbmJWck8yWTZOQ01FQXdEZ\
      1lEVlIwUEFRSC9CQVFEQWdLa01BOEdBMVVkRXdFQi93UUZNQU1CQWY4d0hRWUQKVlIwT0JCWUVGT\
      0VYaHN6eGQwUTc0MlJmYkNEdmVKcnQzYW1xTUFvR0NDcUdTTTQ5QkFNQ0EwZ0FNRVVDSVFEVQowd\
      WZiVGJPS1JXU1NQc3VnQ2Mzd0tYMVQxeGtMSmh2WEE0cW02TjY1SUFJZ0lnWnpvMUlneDAxaWgxa\
      WJnQTM2CnZwRlBrWXlUQUhYbW1MUEJsRXNiZnRrPQotLS0tLUVORCBDRVJUSUZJQ0FURS0tLS0t"

users:
- name: "rancher"
  user:
    token: "kubeconfig-user-cg46f98tlt:xg96rzm2kztz6gfn57bpk5vlftzhbgwdt9kklqzd7gmpvb9jx8c7cr"


contexts:
- name: "rancher"
  context:
    user: "rancher"
    cluster: "rancher"

current-context: "rancher"

~
~
~
~
~
~
                                                                    1,14          All
```

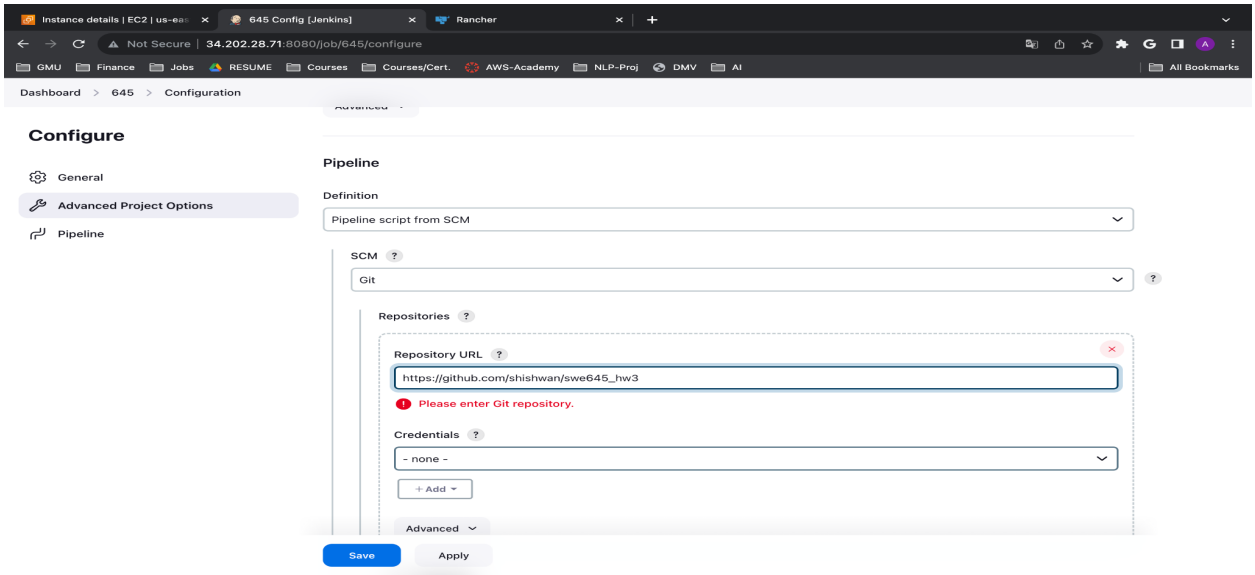
- Push the contents to github https://github.com/a-jagadish/swe_645



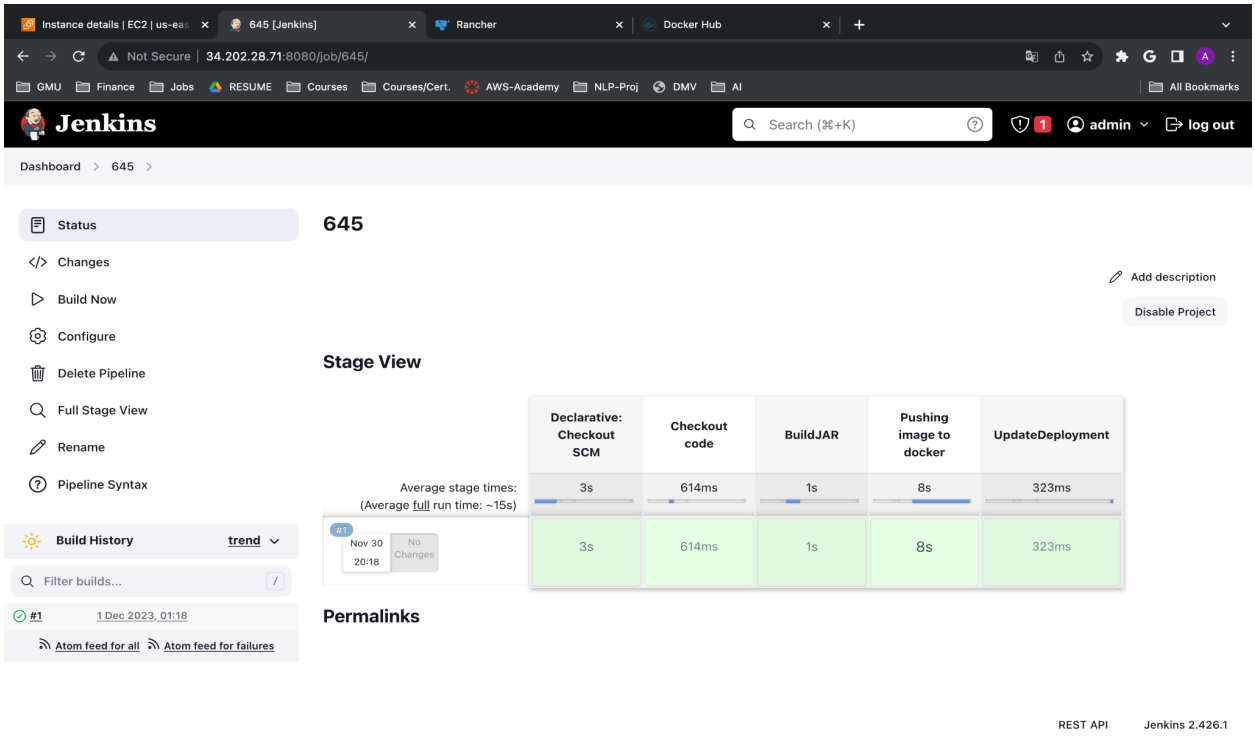
- Go to Dashboard and click on a new item, enter a name, select a pipeline and click OK.



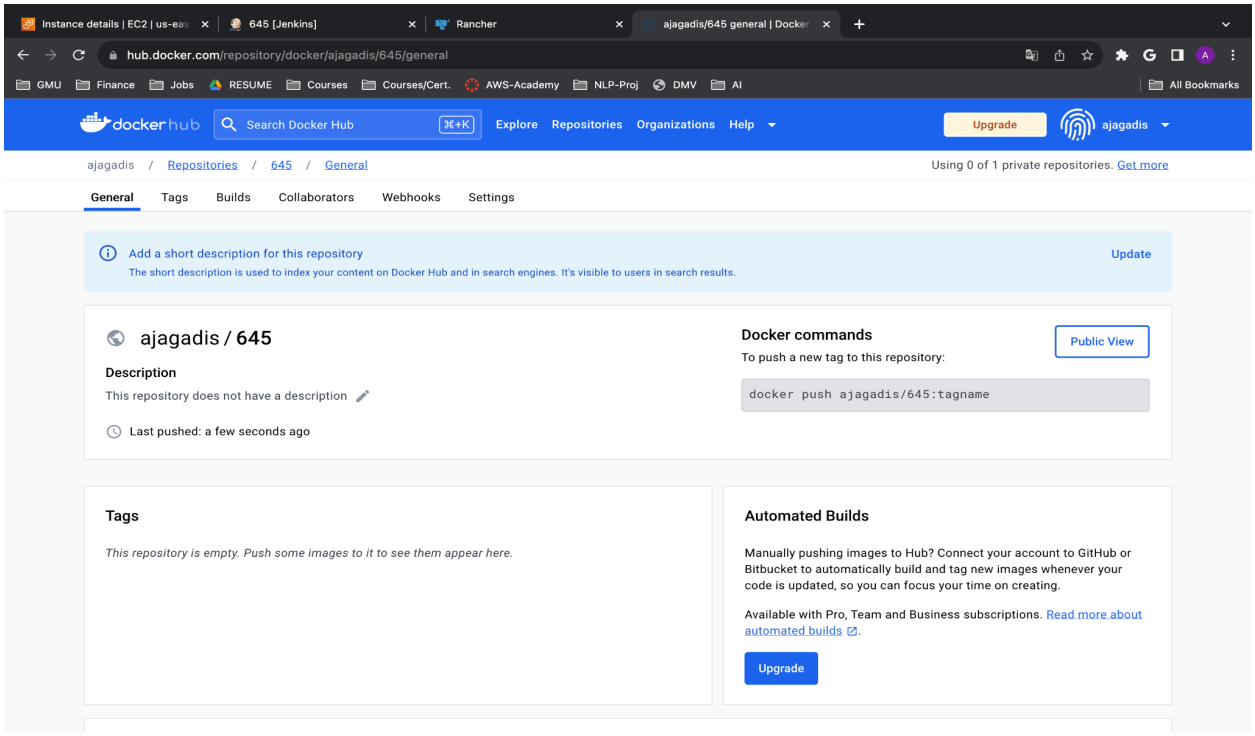
- In the pipeline configuration,under pipeline select pipeline script from SCM and git as SCM and provide github repo url.

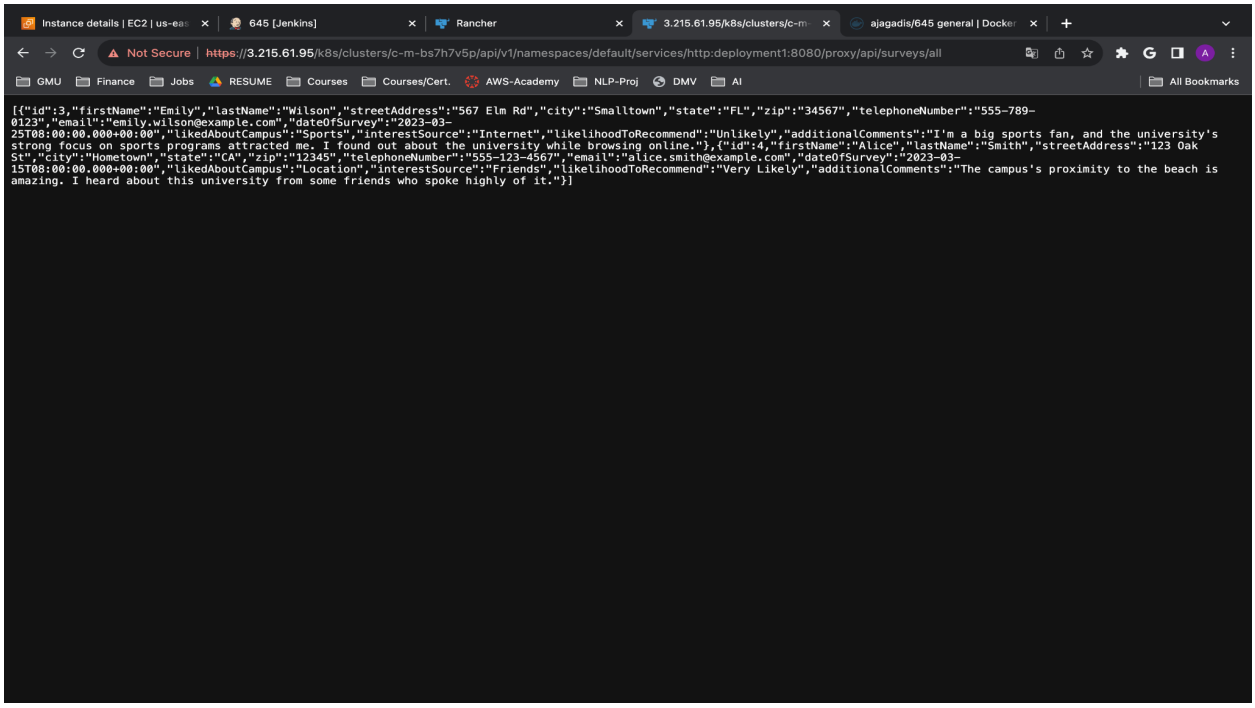


- Push the code and Jenkinsfile to a GitHub repository.
- Go back to the Jenkins dashboard and click on the pipeline job and click on Build.
- Now to trigger the pipeline.



- Once the build is successful, we go back to the rancher and open the deployment url and docker. We can confirm that our Jenkins is built successfully by seeing the tags in the docker and the build deployment in the rancher.




Instance details | EC2 | us-eas
645 [Jenkins]
Rancher
3.215.61.95/k8s/clusters/c-m-
ajagadis/645 general | Docker
Not Secure
https://3.215.61.95/k8s/clusters/c-m-bs7h7v5p/api/v1/namespaces/default/services/http:deployment1:8080/proxy/api/surveys/all
GMU
Finance
Jobs
RESUME
Courses
Courses/Cert.
AWS-Academy
NLP-Proj
DMV
AI
All Bookmarks
[{"id":3,"firstName":"Emily","lastName":"Wilson","streetAddress":"567 Elm Rd","city":"Smalltown","state":"FL","zip":"34567","telephoneNumber":"555-789-0123","email":"emily.wilson@example.com","dateOfSurvey":"2023-03-25T08:00:00.000+00:00","likedAboutCampus":"Sports","interestSource":"Internet","likelihoodToRecommend":"Unlikely","additionalComments":"I'm a big sports fan, and the university's strong focus on sports programs attracted me. I found out about the university while browsing online."},{"id":4,"firstName":"Alice","lastName":"Smith","streetAddress":"123 Oak St","city":"Hometown","state":"CA","zip":"12345","telephoneNumber":"555-123-4567","email":"alice.smith@example.com","dateOfSurvey":"2023-03-15T08:00:00.000+00:00","likedAboutCampus":"Location","interestSource":"Friends","likelihoodToRecommend":"Very Likely","additionalComments":"The campus's proximity to the beach is amazing. I heard about this university from some friends who spoke highly of it."}]