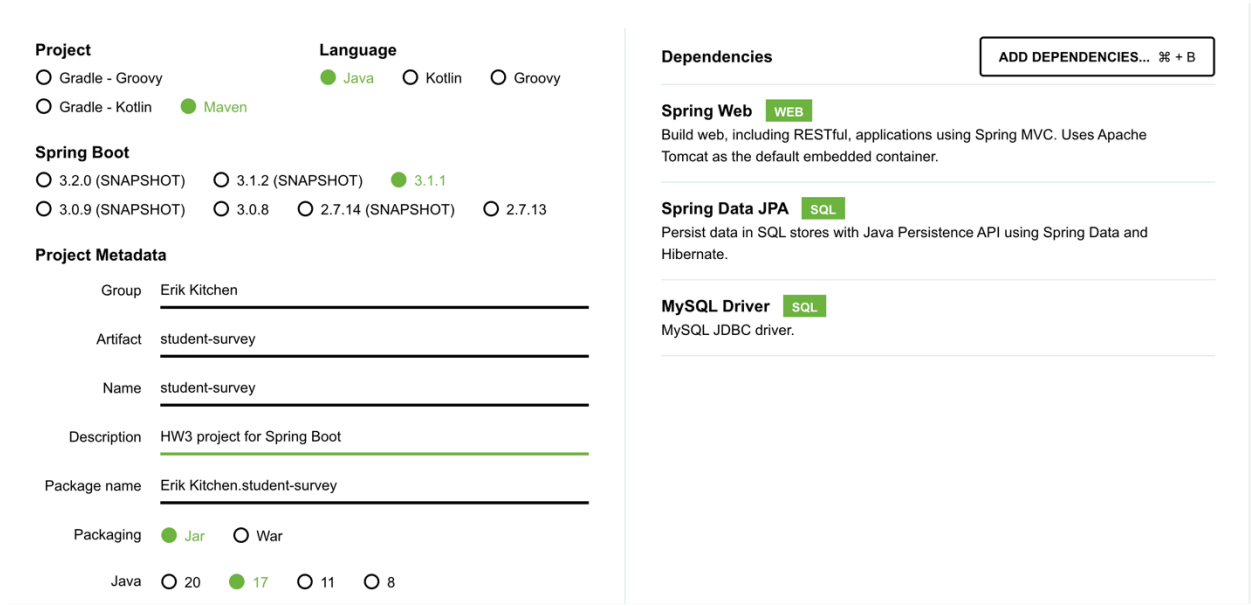
Erik Kitchen
SWE 645 HW 3

Class Home Page: http://swe645-erik-kitchen.com.s3-website-us-east-1.amazonaws.com/
HW3 Repository: https://github.com/erikkitchen/SWE_645_HW3
Spring Boot application link: 34.74.69.88/studentSurvey
Spring Boot health link: 34.74.69.88/health

Prerequisites:
Git downloaded on computer
Github account
Must have Docker installed on local computer. Set up account here: https://hub.docker.com/

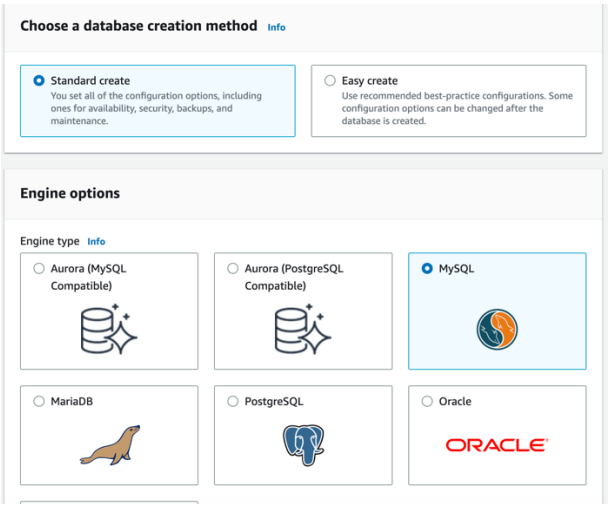
create and download a maven and SpringBoot project:

- Create a Maven project (https://start.spring.io/)
- Do the following
    - Under Project choose “Maven”
    - Under Language choose “Java”
    - Under Spring Boot choose 3.1.1
    - Under Dependencies choose the following
        - “Spring Web”
        - “Spring Data JPA”
        - “MySQL Driver”
        - Name your Group, Artifact, Name, Description, and Package Name
    - Under Packaging select “Jar”
    - Under Java select “17”
    - Click on Generate, unzip download, and move folder where you would like to store it.

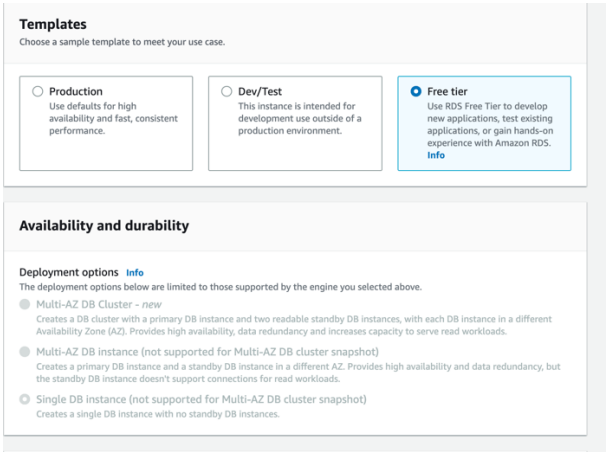


Setup Database

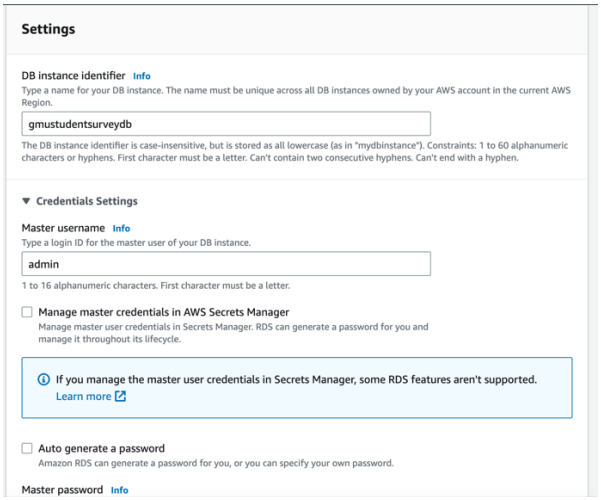
- Go to Amazon RDS: https://us-east-2.console.aws.amazon.com/rds/home?region=us-east-2#databases:
- Click on Create Database
- Choose the following options:
    - Choose “Standard create”
    - Choose “MySQL”



    - Choose “Free tier”

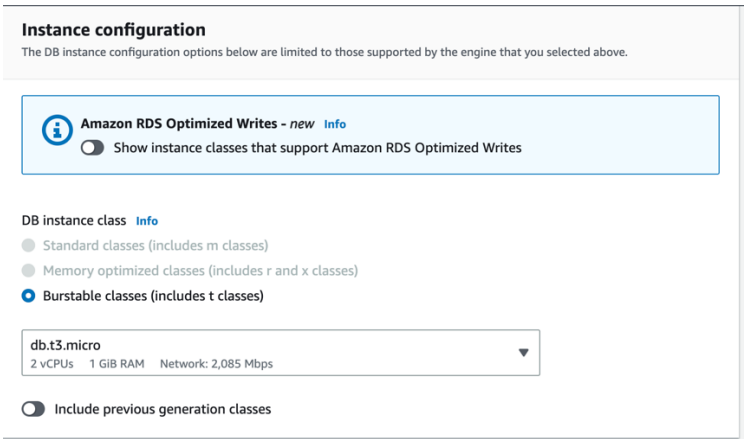


- Name DB instance identifier “gmustudentsurveydb”
- Set up username

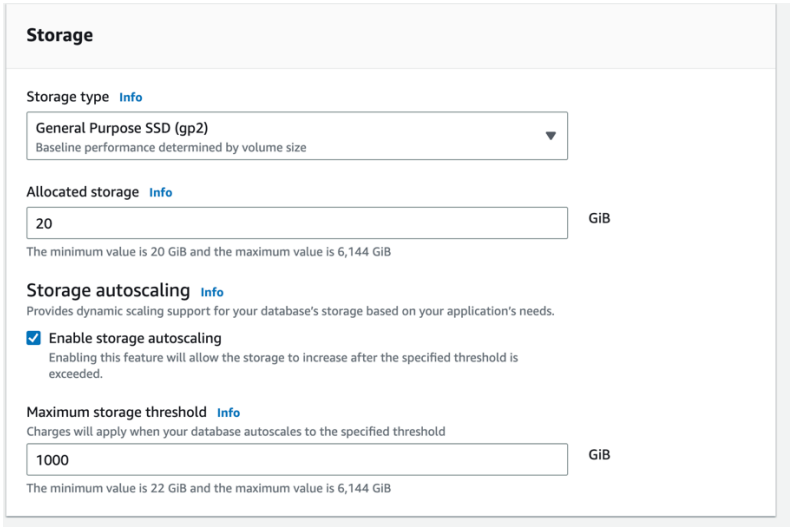


- Setup password

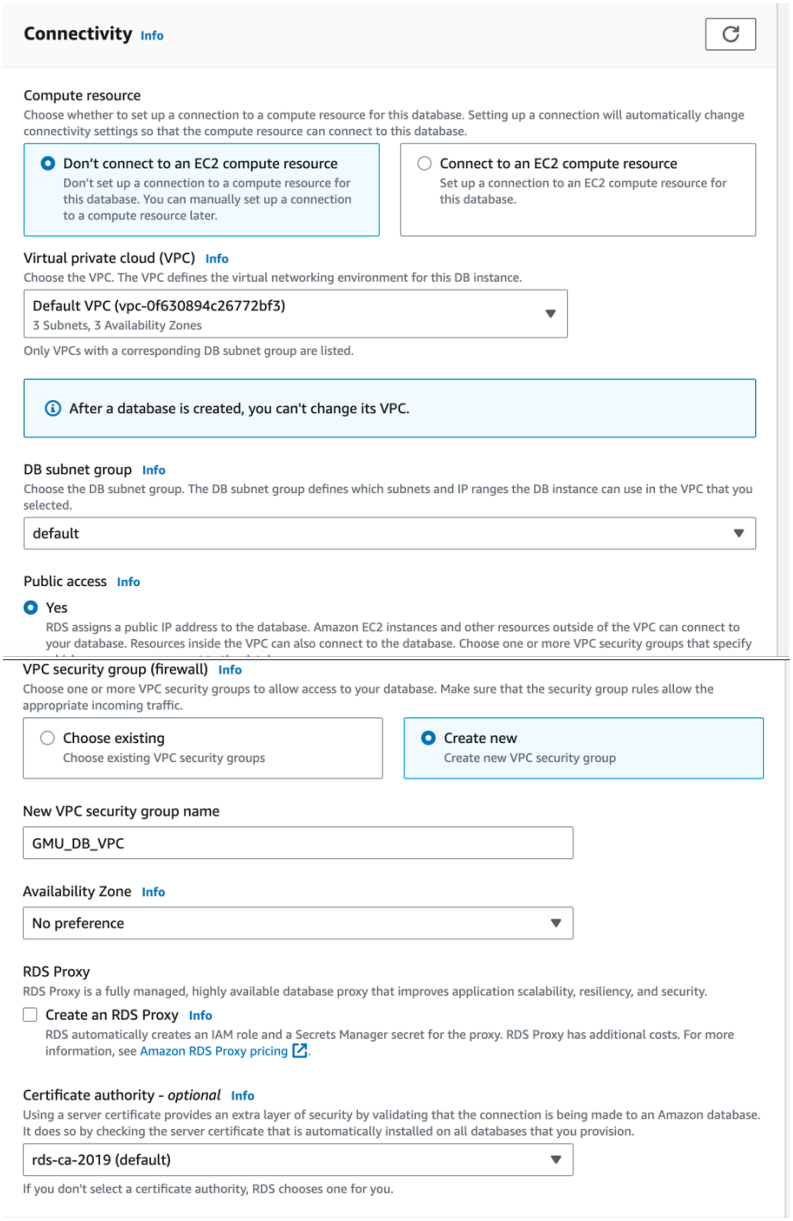
- Choose “Burstable classes”
- Then choose “db.t3.micro”



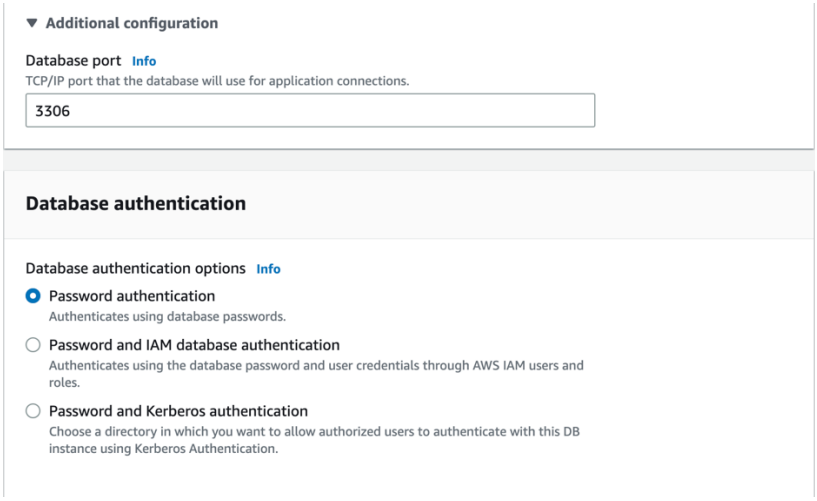
- Choose the following for storage:
  - "General Purpose SSDc(gp2)"
  - "20" GiB
  - Check "Enable storage autoscaling"
  - "1000" GiB



- Under Connectivity Choose the following:
  - "Don't connect to an EC2 compute resource"
  - "Default VPC"
  - "Default"
  - Check "Yes" for public access
  - "Create new"
  - Name VPC group name
  - "No Preference"
  - Choose "rds-ca-2019 (default)"



- Under Additional configuration, select port “3306” and check “password authentication”



- Under Additional configuration choose the following:
  - Insert Initial database name (Use the same name as the DB instance identifier)
  - Choose “default.mysql8.0”
  - Choose “default.mysql8.0”
  - Check “Enable automated backups”
  - Choose “1” day
  - Choose “no preference”
  - Check “copy tags to snapshots”
  - Check “Enable encryption”
  - Choose “(default)aws/rds”
  - Check “Enable auto minor version upgrade”
  - Choose “No preference”

Additional configuration

Database options, encryption turned on, backup turned on, backtrack turned off, maintenance, CloudWatch Logs, delete protection turned off.

Database options

Initial database name Info

gmustudentsurveydb

If you do not specify a database name, Amazon RDS does not create a database.

DB parameter group Info

default.mysql8.0

Option group Info

default:mysql-8-0

Backup

Enable automated backups

Creates a point-in-time snapshot of your database

⚠ Please note that automated backups are currently supported for InnoDB storage engine only. If you are using MyISAM, refer to details here.

Backup retention period Info

The number of days (1-35) for which automatic backups are kept.

1 day

Backup window Info

The daily time range (in UTC) during which RDS takes automated backups.

Choose a window

No preference

Copy tags to snapshots

Backup replication Info

Enable replication in another AWS Region

Enabling replication automatically creates backups of your DB instance in the selected Region, for disaster recovery, in addition to the current Region.

Encryption

Enable encryption

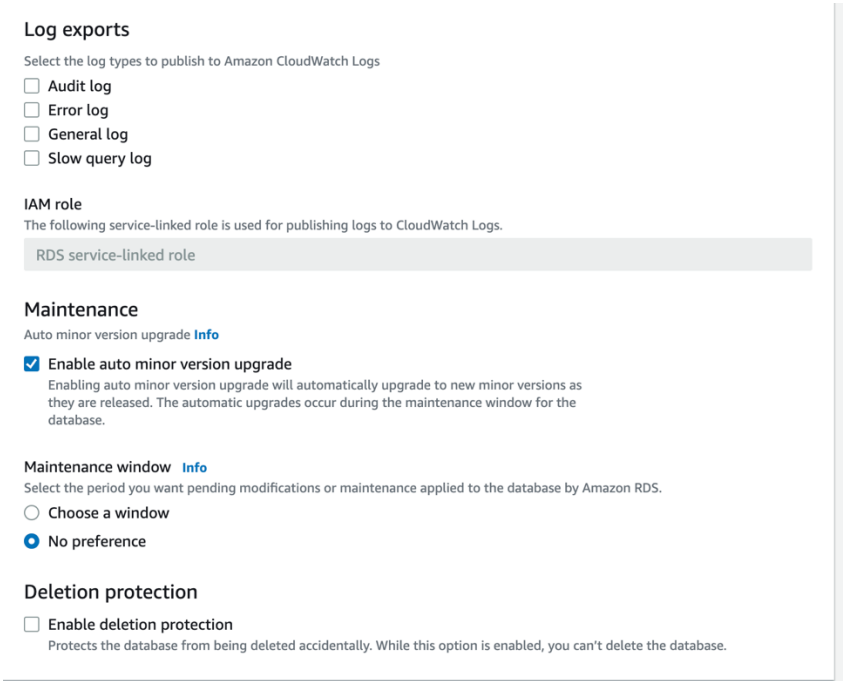
Choose to encrypt the given instance. Master key IDs and aliases appear in the list after they have been created using the AWS Key Management Service console. Info

AWS KMS key Info

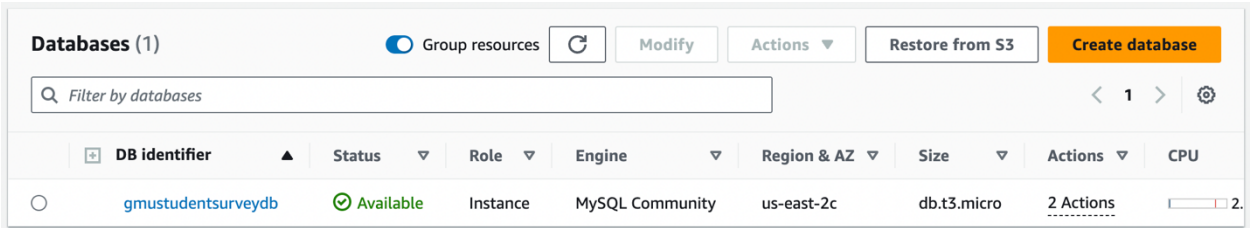
(default) aws/rds

Account

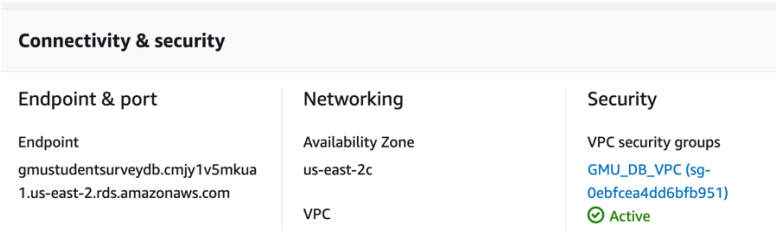
927832898956



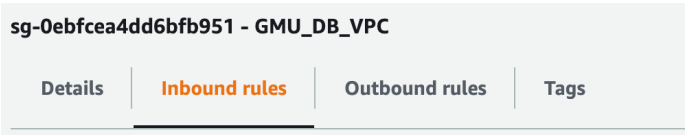
- Click on “Create Database”
- Your new database will now show up on the RDS dashboard
- Click on your new database DB identifier



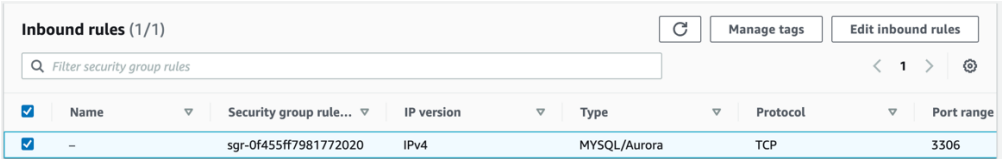
- Click on the VPC security groups link



- Click on Inbound rules

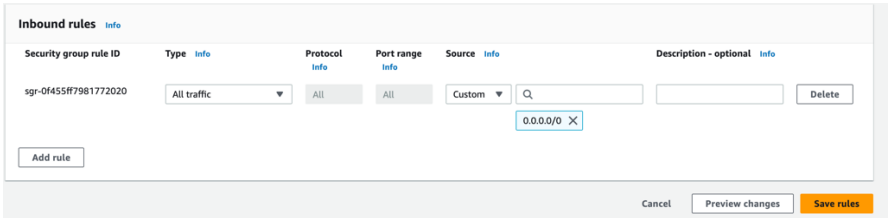


- Click on Edit inbound rules



- Update the following:

    - Type to “All traffic”
    - Source to “Custom”
    - Insert “0.0.0.0/0”
- Click Save rules

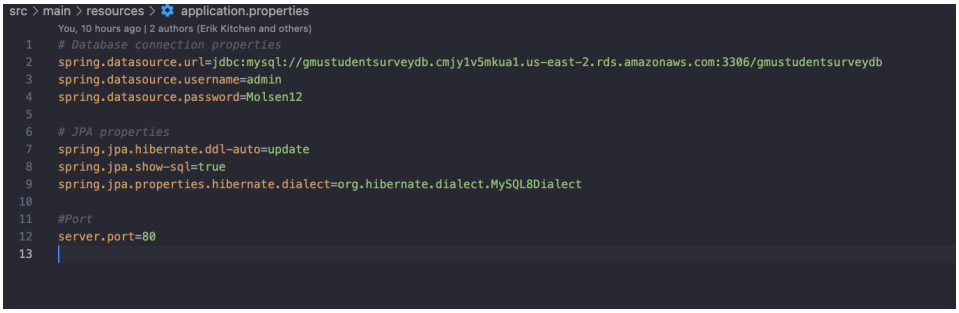


Download dependencies (For Visual Studios code IDE)

- Download the following
    - Spring Boot Tools
    - Extension Pack for Java

Set up/update files

- Update the following in applications.properties
    - Update spring.datasource.url=jdbc:<database DNS>:3306/gmustudentsurveydb
    - Update username and password (From above when creating database)
    - Copy JPA properties in picture
    - Server Port should be 80



- Update pom.xml
    - Update main path to yours (<mainClass>Erik.Kitchen.studentsurvey.StudentSurveyApplication</mainClass>)
- Create the following folders at the same level as the main application class (StudentSurveyApplication.java) and create files inside (See github repository on what to put in)
    - controller
        - StudentSurveyController.java
        - HealthController.java
    - entity
        - StudentSurvey.java
    - repository
        - StudentRepository.java
    - Service
        - StudentSurveyService.java

Set up git repository (Follow instructions here with a few updates below: https://github.com/erikkitchen/SWE_645_HW2/blob/main/README.pdf):

- Create GitHub repository named 'SWE_645_HW3' instead of 'SWE_645_HW3'
- Follow the rest of the steps in the link above

Set up Rancher (Follow instructions here: https://github.com/erikkitchen/SWE_645_HW2/blob/main/README.pdf)

Create Docker Image (Follow instructions here with a few updates below: https://github.com/erikkitchen/SWE_645_HW2/blob/main/README.pdf):

- Save file in Spring Boot repository instead of the HW2 repository
- Use updated dockerfile in HW3 github repository

Set up Kubernetes Cluster on GKE (Follow instructions here: https://github.com/erikkitchen/SWE_645_HW2/blob/main/README.pdf)

Import GKE Cluster into Rancher (Follow instructions here: https://github.com/erikkitchen/SWE_645_HW2/blob/main/README.pdf)

Install Jenkins (Follow instructions here with a few updates below: https://github.com/erikkitchen/SWE_645_HW2/blob/main/README.pdf):

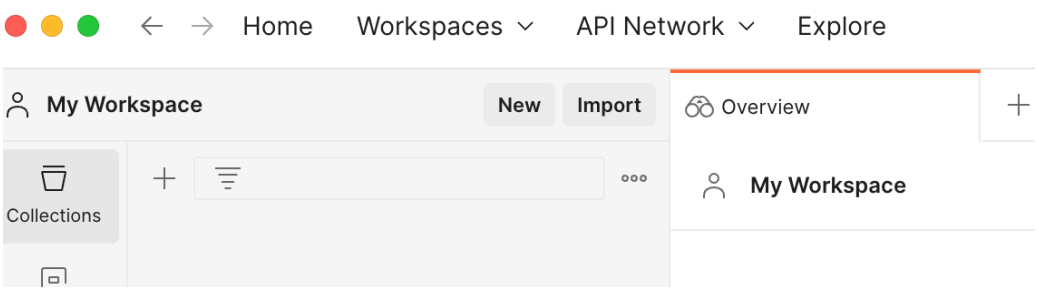
- Install Maven
    - sudo apt update
    - sudo apt install maven
- Install jdk 17
    - sudo apt update
    - sudo apt upgrade
    - sudo apt install openjdk-17-jdk
- Go to Manage Jenkins
- Click on Plugins
- Search “Maven Integration plugin” and install
- Go back to Manage Jenkins
- Go to Tools
- Go down to Maven and click “Add Maven”
- Name “Maven”
- Unclick Install automatically and in MAVEN_HOME enter the location of your Maven installation (/usr/share/maven)

Add Pods and expose webapp (Follow instructions here with a few updates below: https://github.com/erikkitchen/SWE_645_HW2/blob/main/README.pdf):

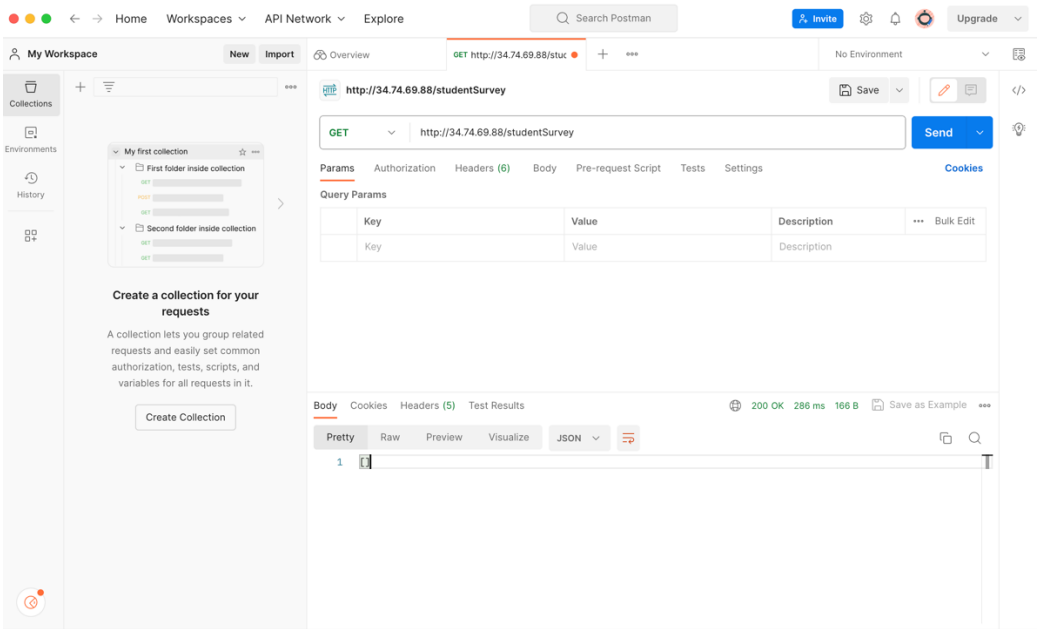
- Port should be 80 (Unless you chose different port for Spring Boot application)

Test Spring Boot application with Postman

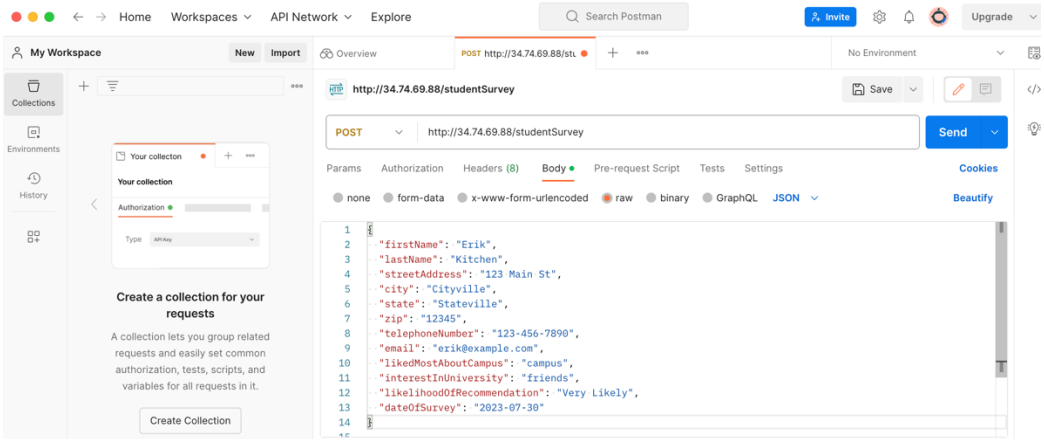
- Download Postman (https://www.postman.com/downloads/)
- Click on “New”



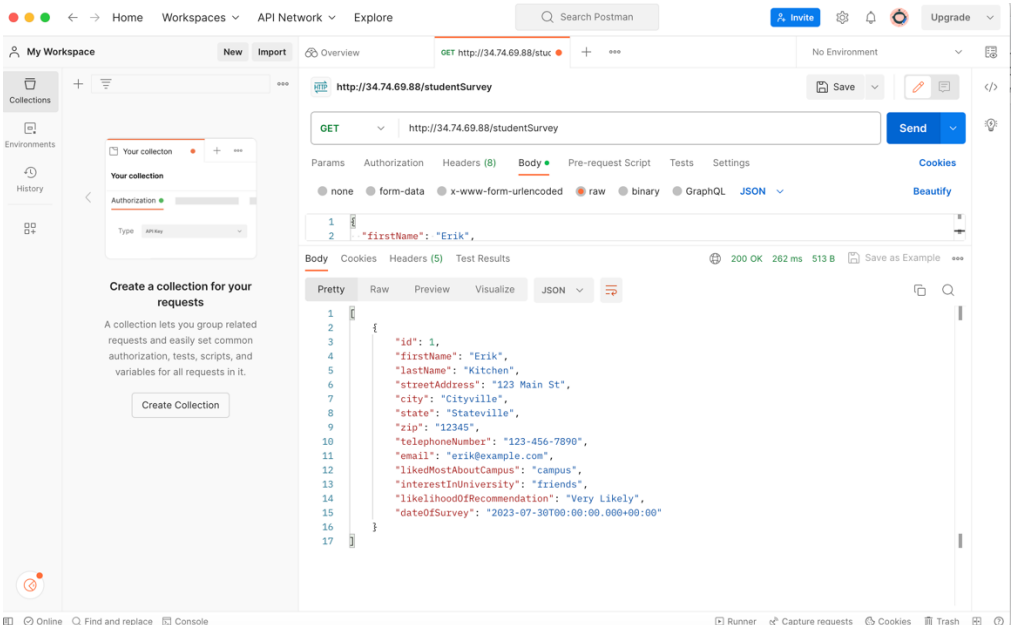
- Insert url with port and controller request mapping and click send (Should get an empty JSON file)



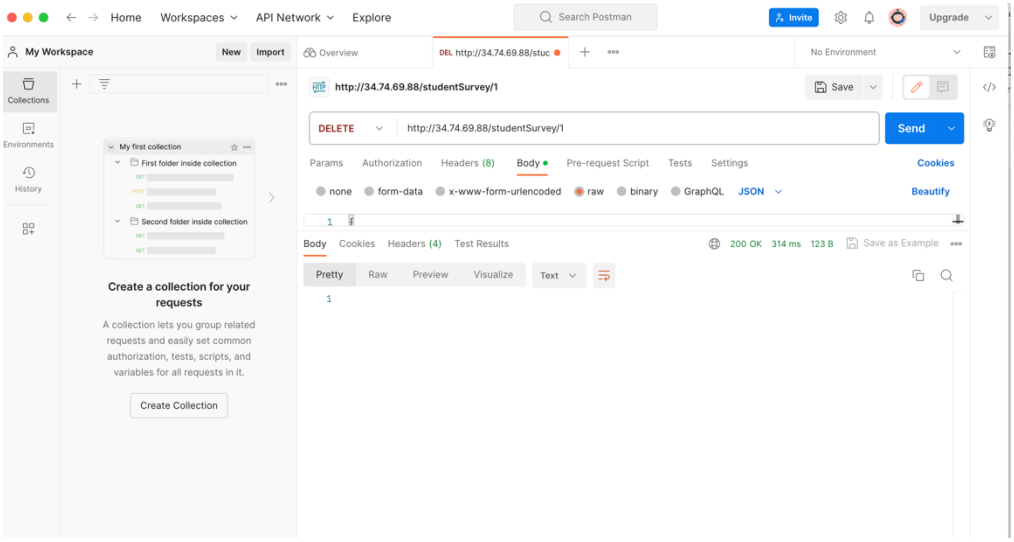
- Change to POST
- Click on “Body”
- Change to JSON
- Insert the below from picture



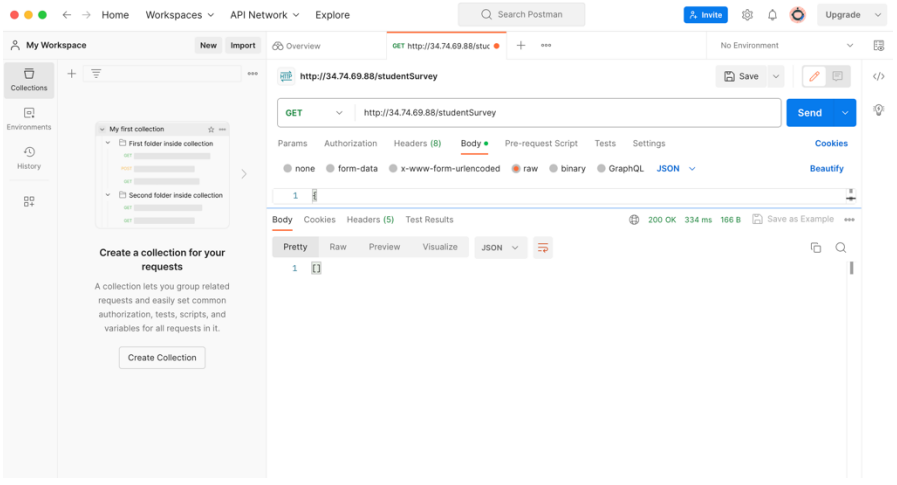
- Test GET again, The JSON you just posted will show up



- Test DELETE, Update url and add /<id> (id will show up in JSON from the GET above)



- Test GET, will come back empty



For videos to set up Rancher, Set up Kubernetes clusters, set up GKE, and set up Jenkins here:
https://github.com/erikkitchen/SWE_645_HW2/tree/main/Videos