Business Requirements:

- User can upload HAPPY/SAD/NEUTRAL

- User can get his/her mood frequency distribution

- User can get the list of nearby locations where he/she is happy

Assumptions:

- frequency distribution is to return the frequency of different mood status for a given user.

- I made the uploadMood API to take in latitude and longitude as well. This is because in reality there should be physical IPs from client that would involve a 3rd party data/API to convert the IPs into GPS coordinates. I was not able to any great services that are free and easy to use so I decided to mock up the client position as well as the nearby place locations as well. In this demo application, you will see that I set up home to be latitude 1 and longitude 1, office to be 3, 4 and mall to be 10, 10.

Technical Specifications:

- Tech stack:

Java (JDK 11)

Spring boot (Web, Security)

Maven

JPA

H2 embedded database

- API

/uploadMood (Post) (moodStatus, latitude, longitude)

/getMoodDistribution (Get)

/getHappyNearByLocations (Get)

- Data Model

Mood (Table to track mood activities for all users)

Location (Table to store nearby location coordinates)

Coordinate (To calculate nearby location)

MoodDistribution (To return distribution data)

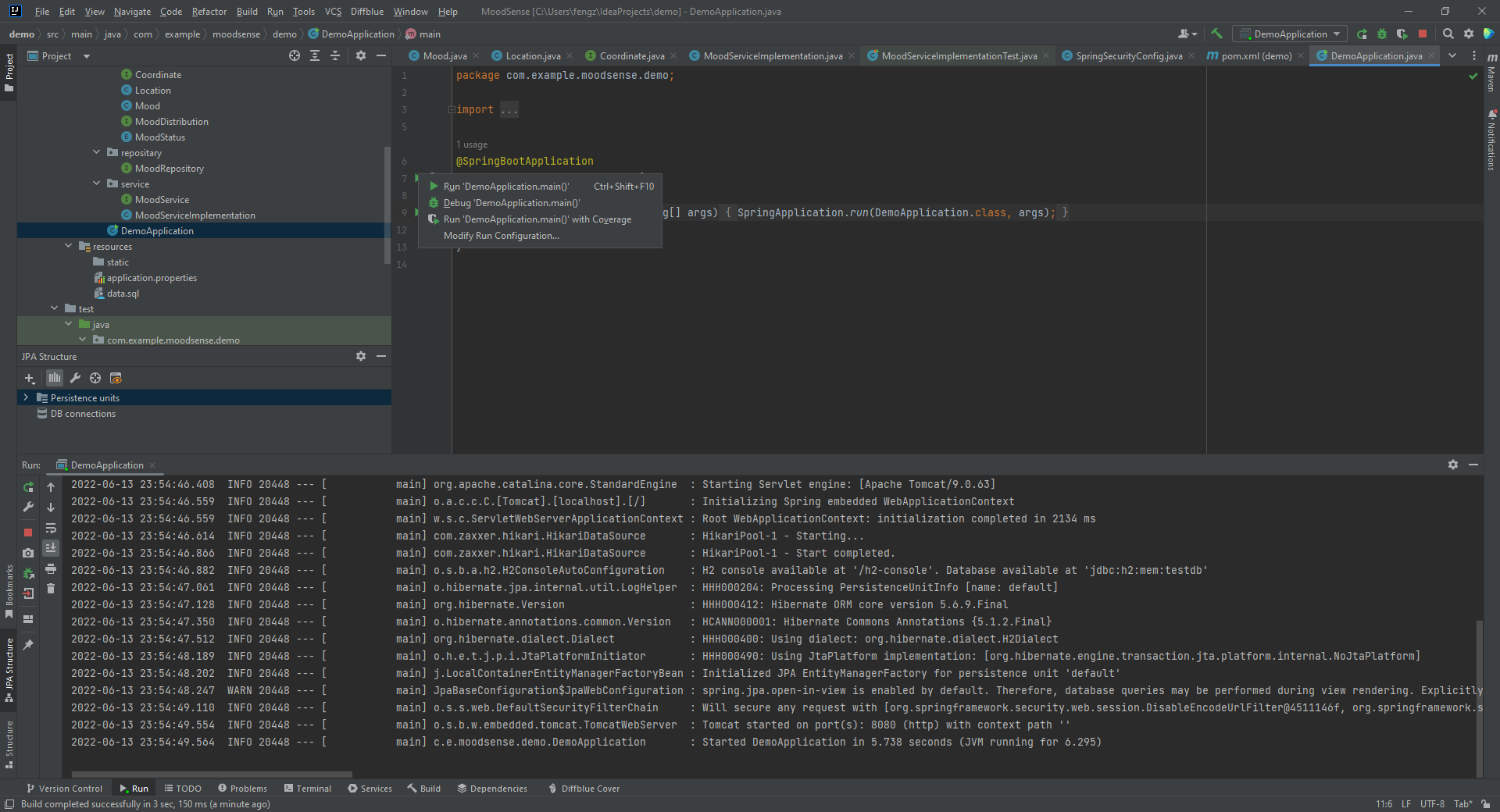
MoodStatus (Enum for different mood status)

How to set up the project:

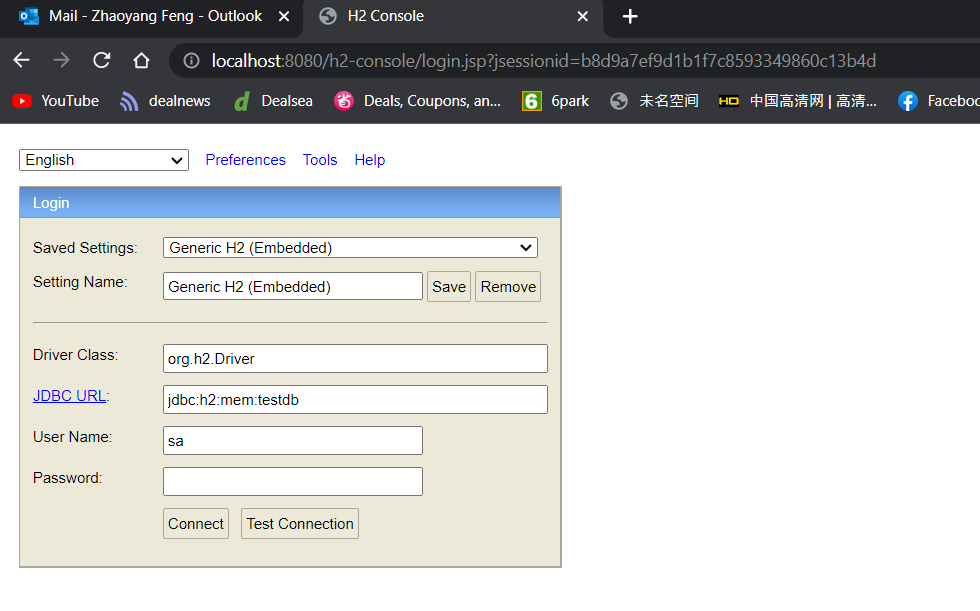
* Note that the entire project is in memory, meaning all steps should happen while the application is running.
* Two users are pre-defined in the project. user1 with pwd1 and user2 with pwd2. Make sure to use basic authentication to run the APIs.
* Use Intellij to load the project(demo folder) as a Maven project. Make sure Maven is able to download all dependency libraries in pom.xml.

How to run the project:

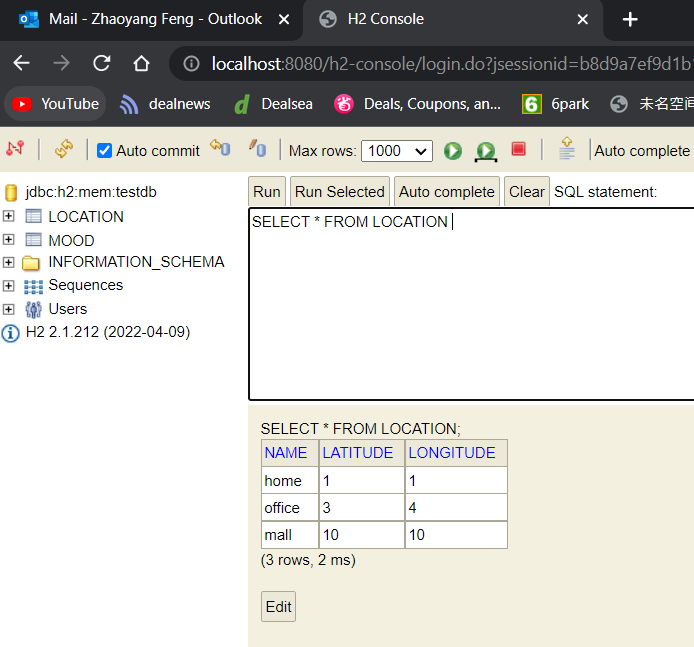
Click run DemoApplication to start the project:



In Chrome, enter <http://localhost:8080/h2-console/> to enter H2 console, password is blank:



Once connected, make sure select \* from LOCATION returns 3 records as pre-defined:



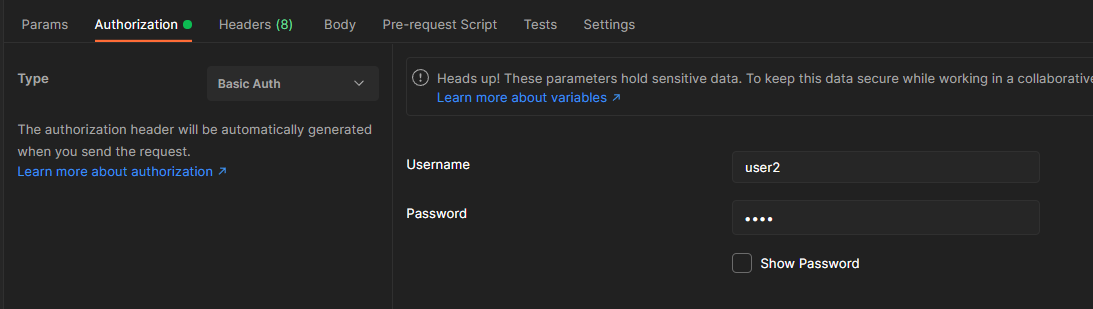
**TEST /uploadMood:**

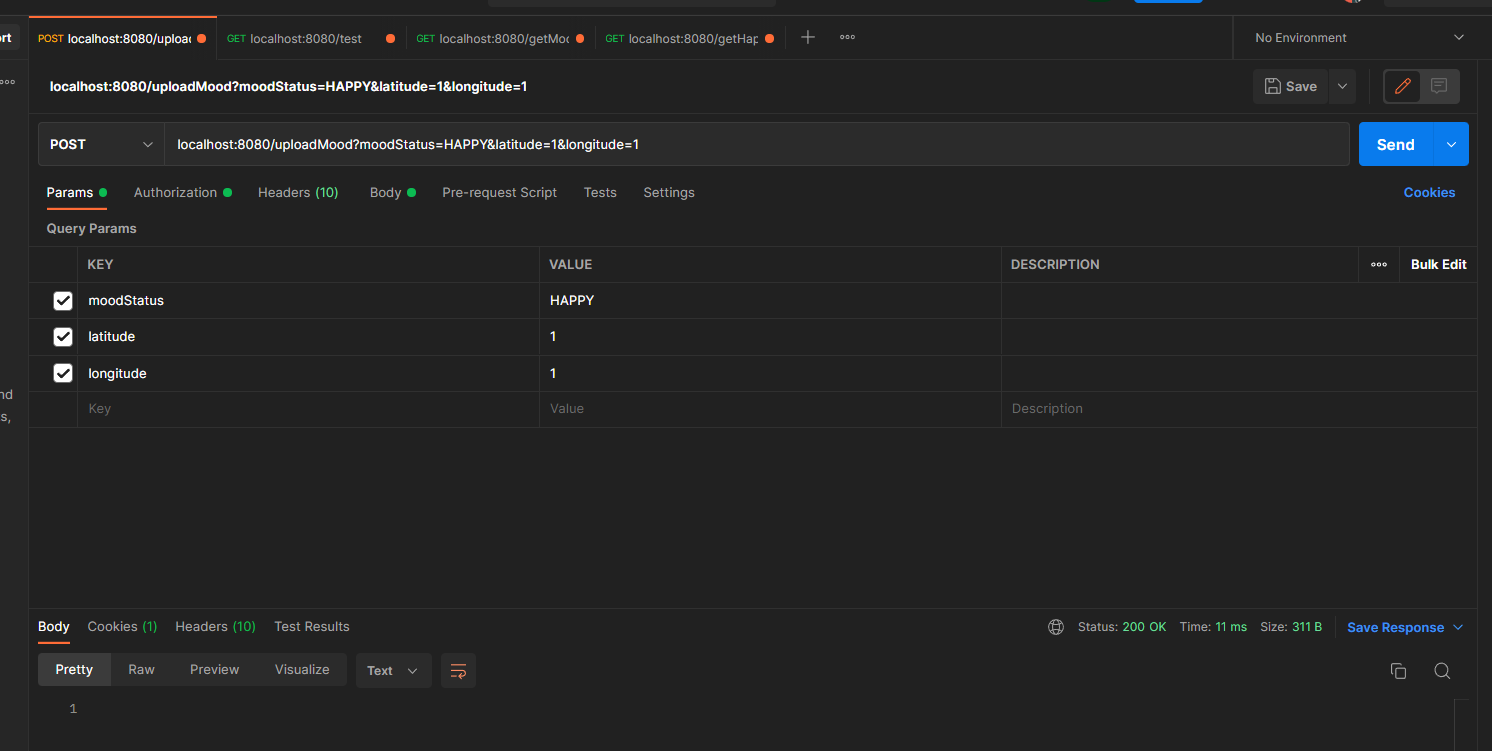
User Postman to do /uploadMood POST request:

Make sure you use ‘Basic Auth’ and enter Username and Password, user1 with pwd1 or user2 with pwd2, else request will be denied.

You can enter HAPPY, SAD or NEUTRAL as moodStatus.

latitude and longitude are restricted from 1 to 10 for demo purpose.



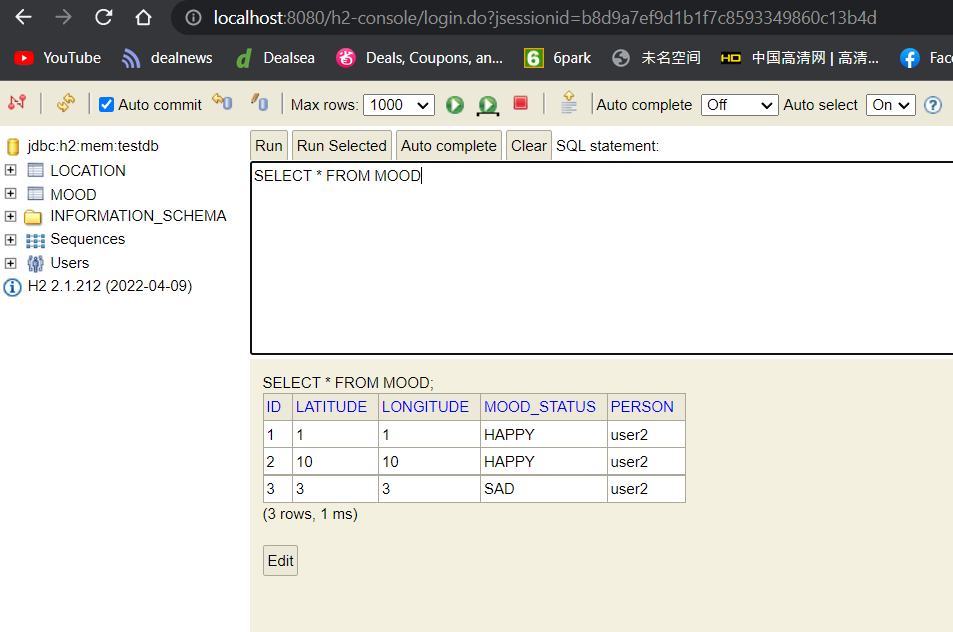


Click send and make sure you get 200 success response back.

Repeat the same to upload HAPPY, 10, 10.

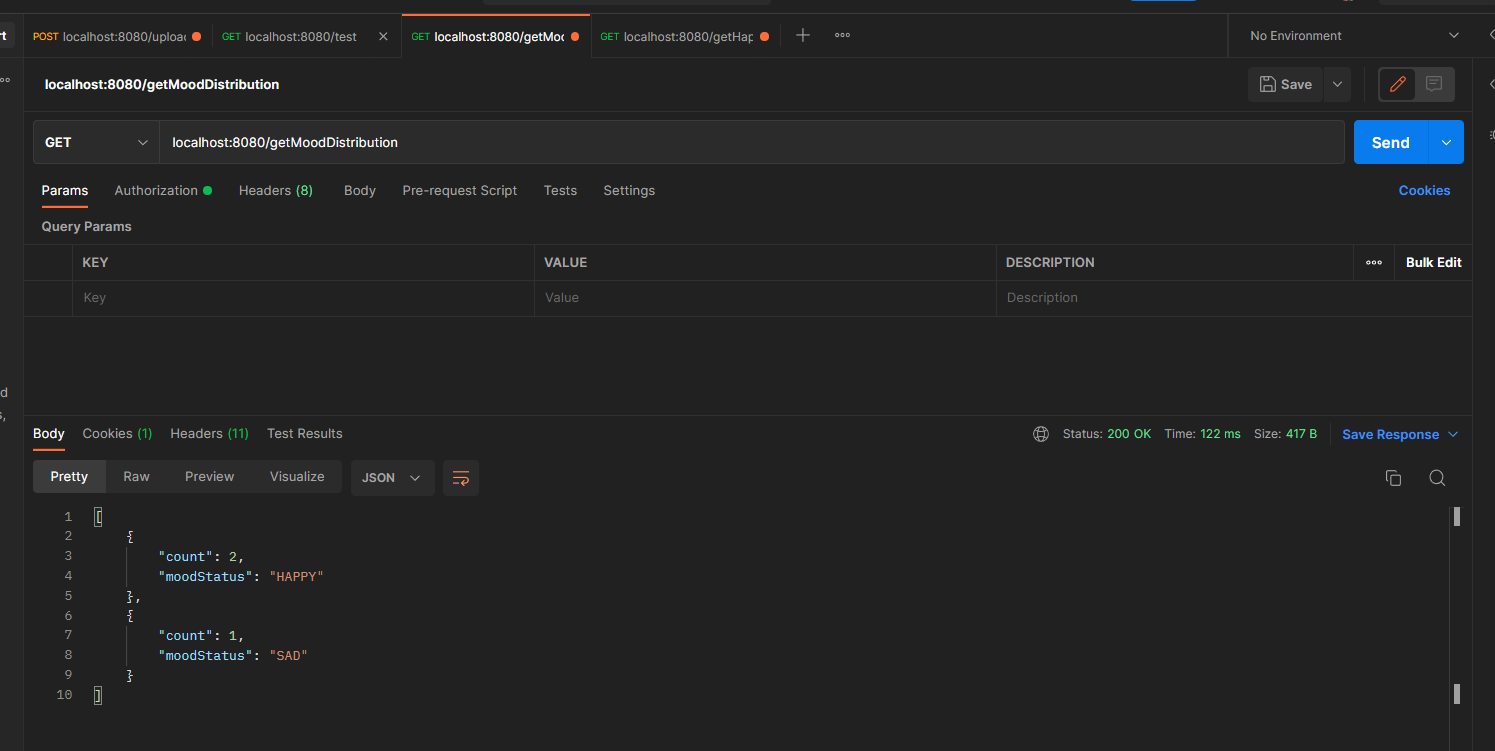
Repeat the same to upload SAD, 3, 3.

Check in H2 on MOOD table and you should see 3 records:



TEST getMoodDistribution:

Use post man to do a GET on localhost:8080/getMoodDistribution



You should see HAPPY count is 2 and SAD count is 1.

TEST getHappyNearByLocations:

Use post man to do a GET on localhost:8080/getHappyNearByLocations:

Since there are two HAPPY locations, for each of the location, the process will calculate the most nearby location based on the client coordinate. In this example, home and mall will be returned.

