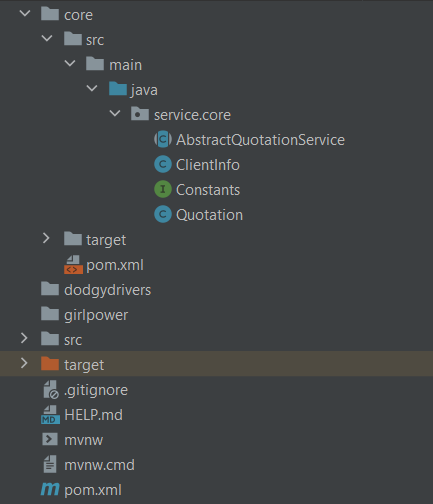
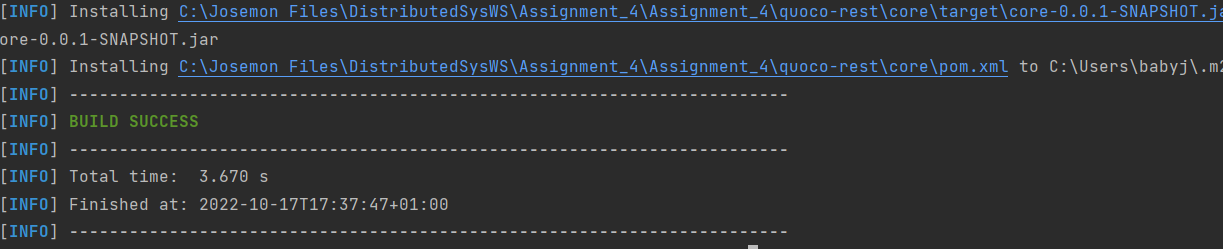
**Task 1:**

Configured the base project and set up core as well, the project structure looks like this:-

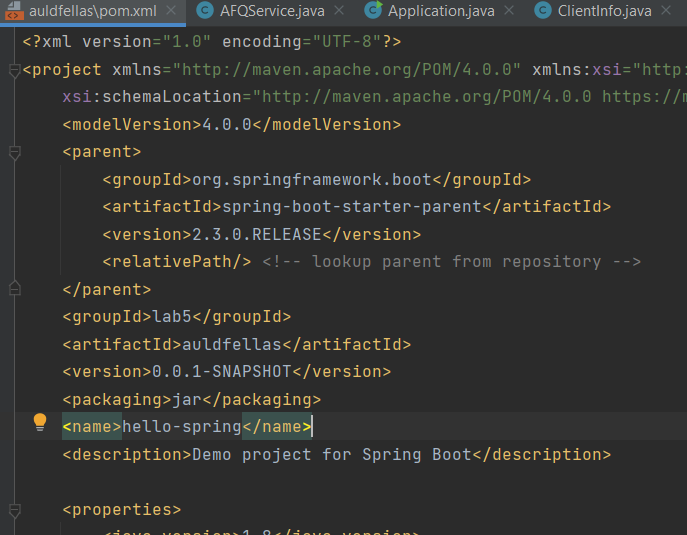


Modified and compiled core with result:-

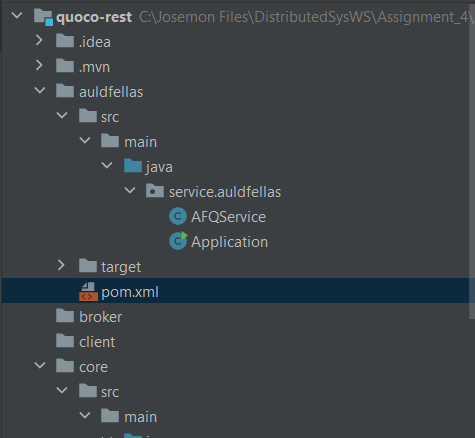


**Task 2:**

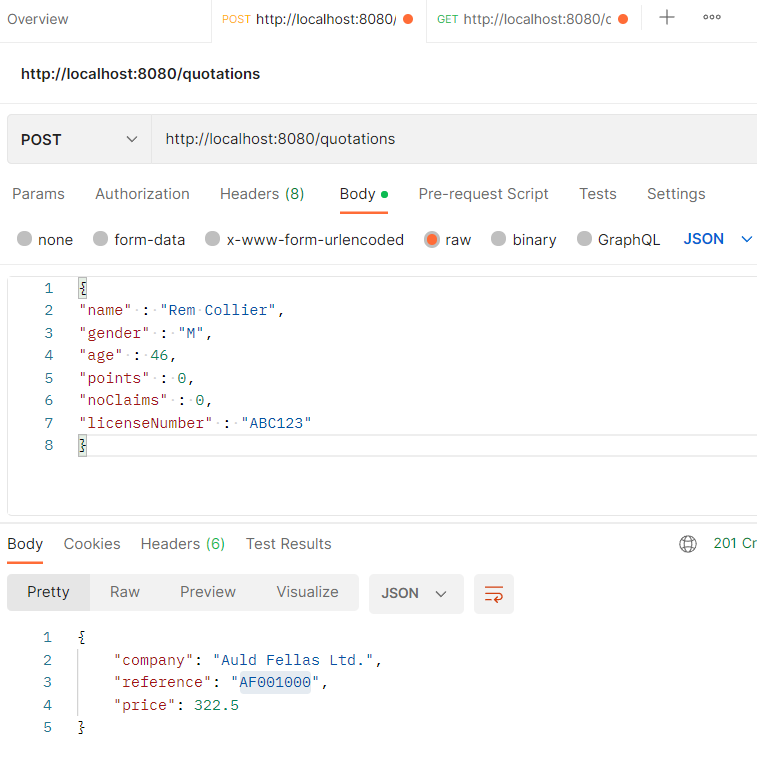
Modified the auldfellas pom file to look like the following:-



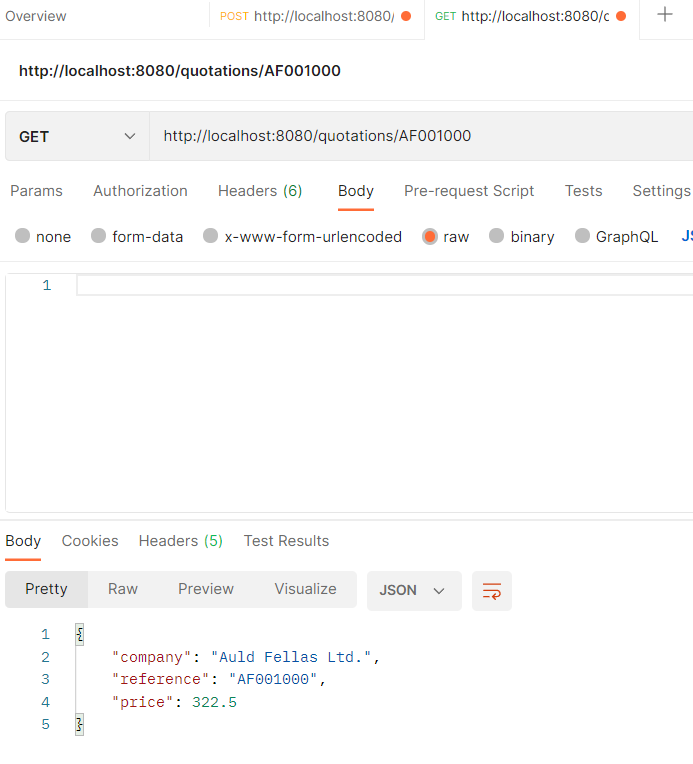
Adding all the required files this is the resultant file structure:-



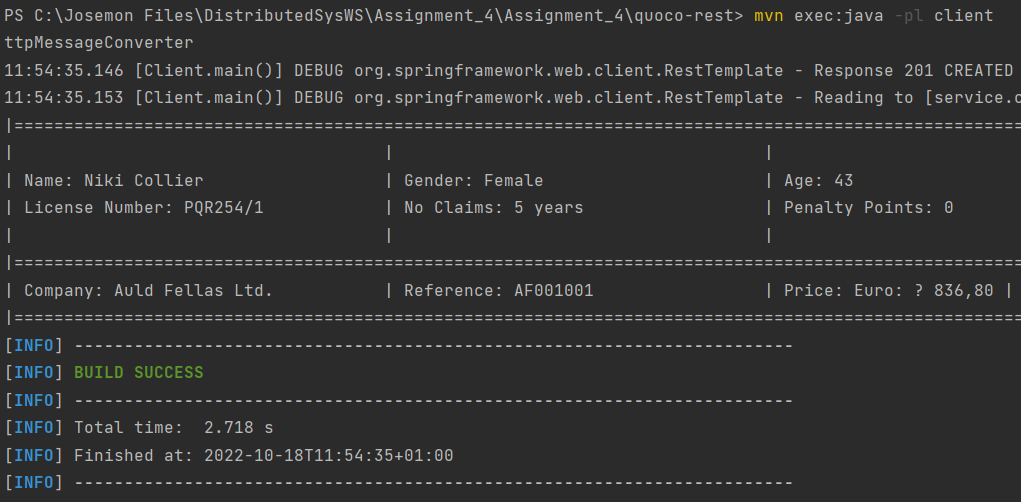
Running the application and testing on POSTMAN:-



And checking the GET :-



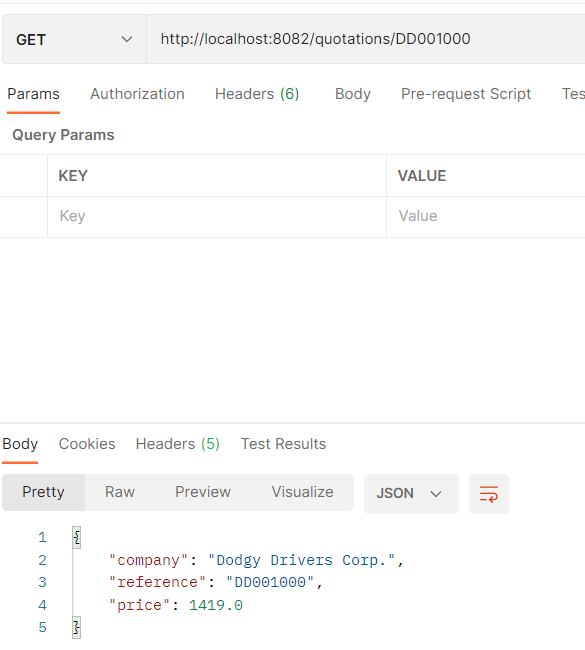
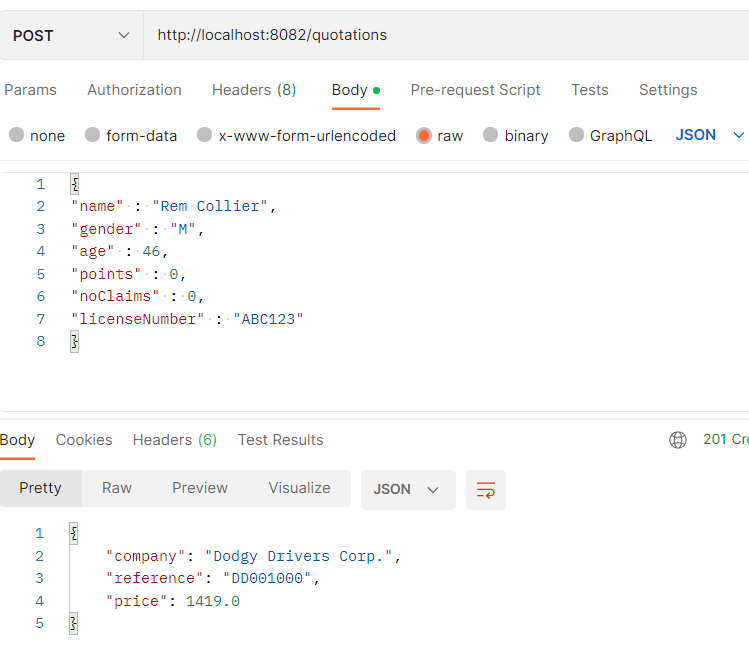
Copied the auldfellas pom and after adding the necessary changes to it, then creating a client main file with data from previous lab client and imports. Clean compile install and running along with the Auldfellas service resulted in the following output:-



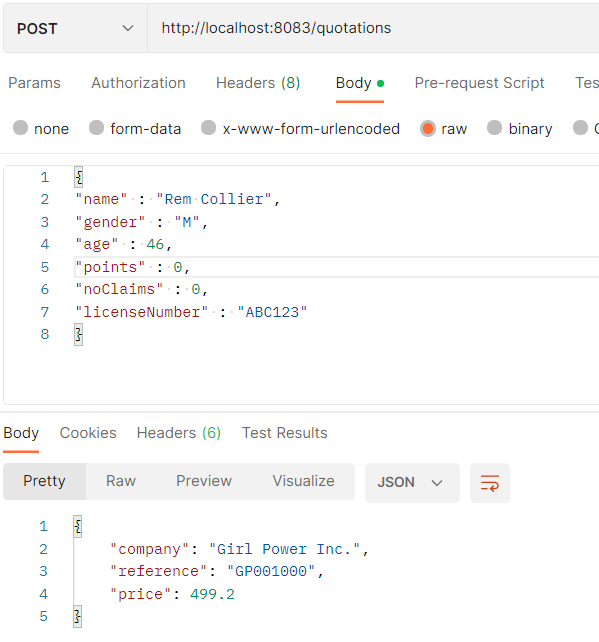
**Task 3:**

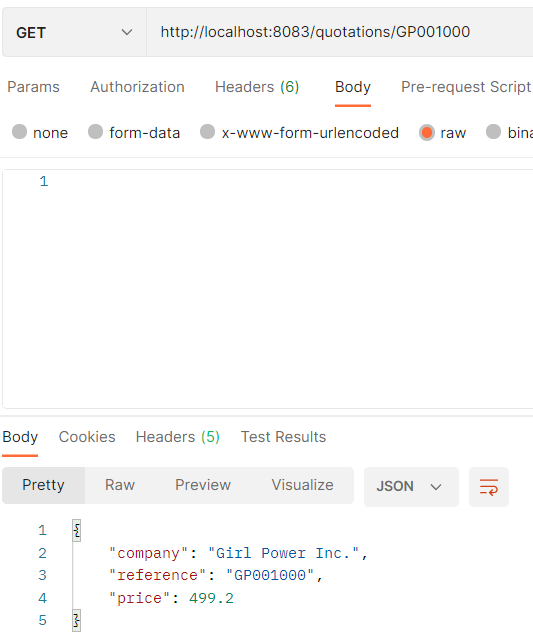
Implementing the changes in auldfellas into dodgydrivers and girlpower with the port changes we see the following results:-

Dodgydrivers:-

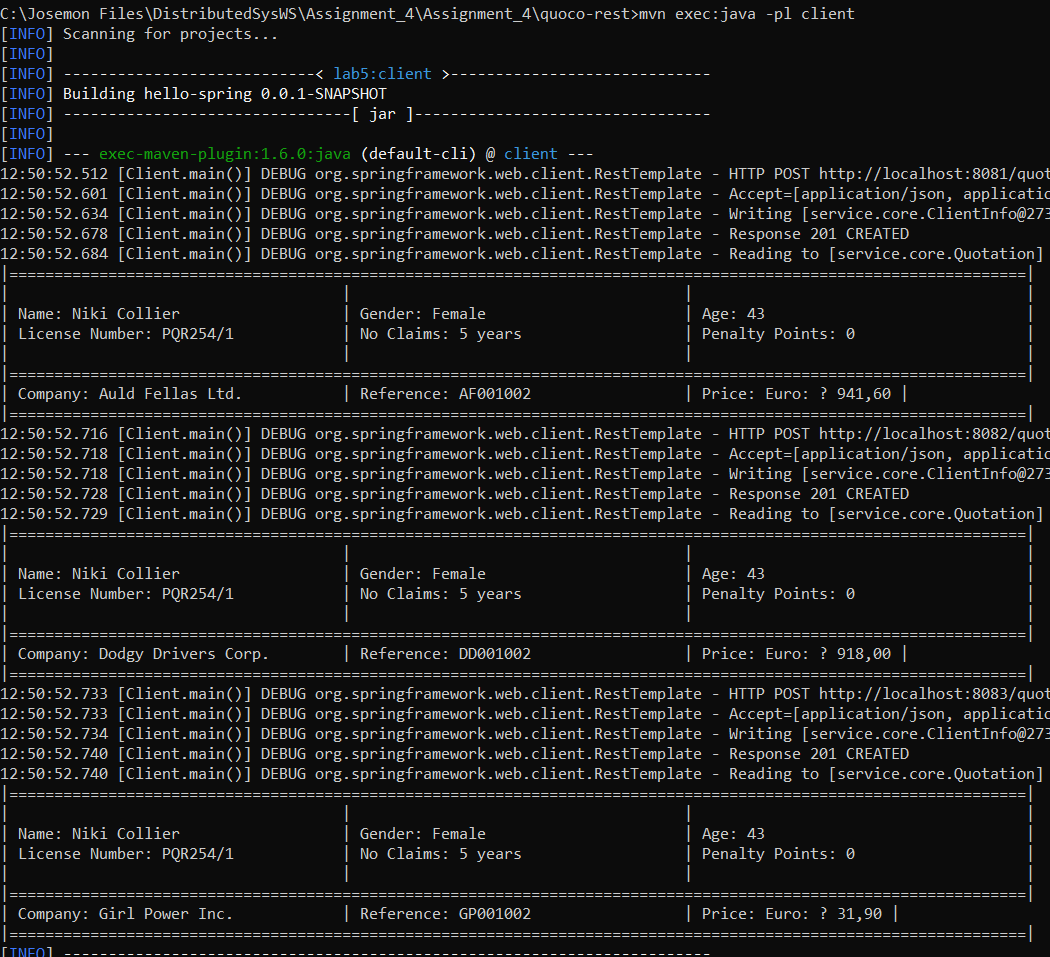


Girlpower:-



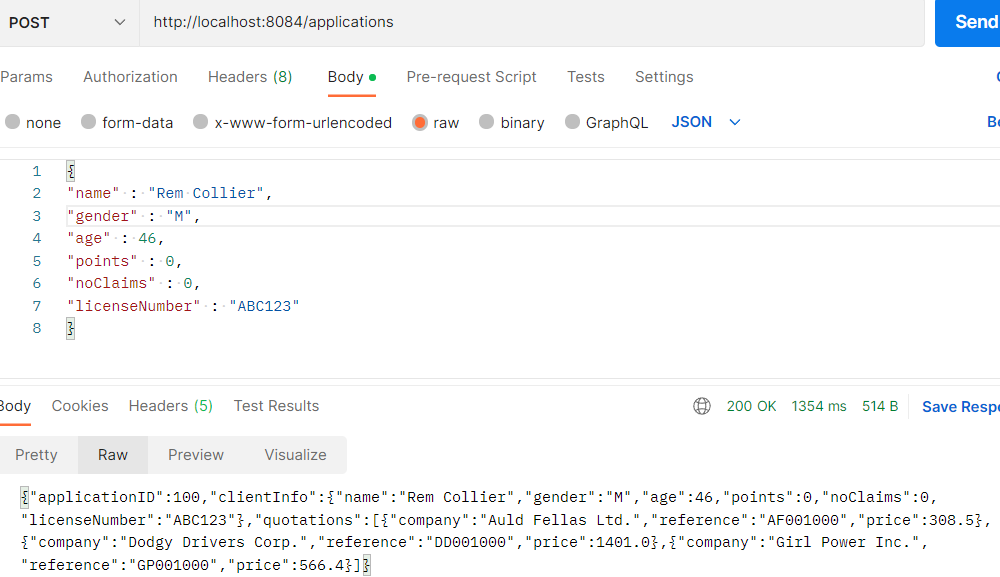


Running them all together to get all the values from client with different ports:-

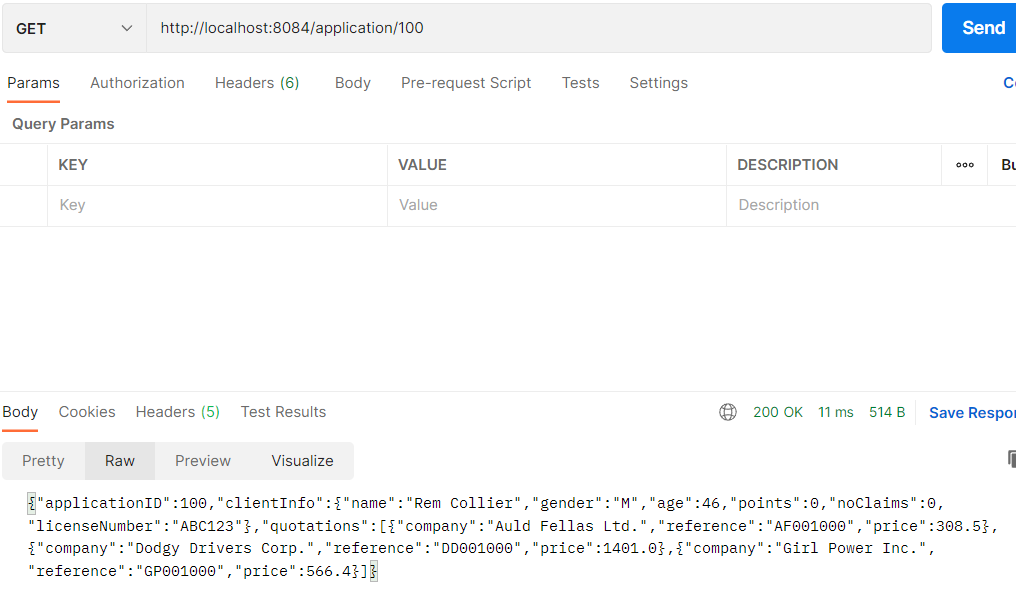


**Task 4:**

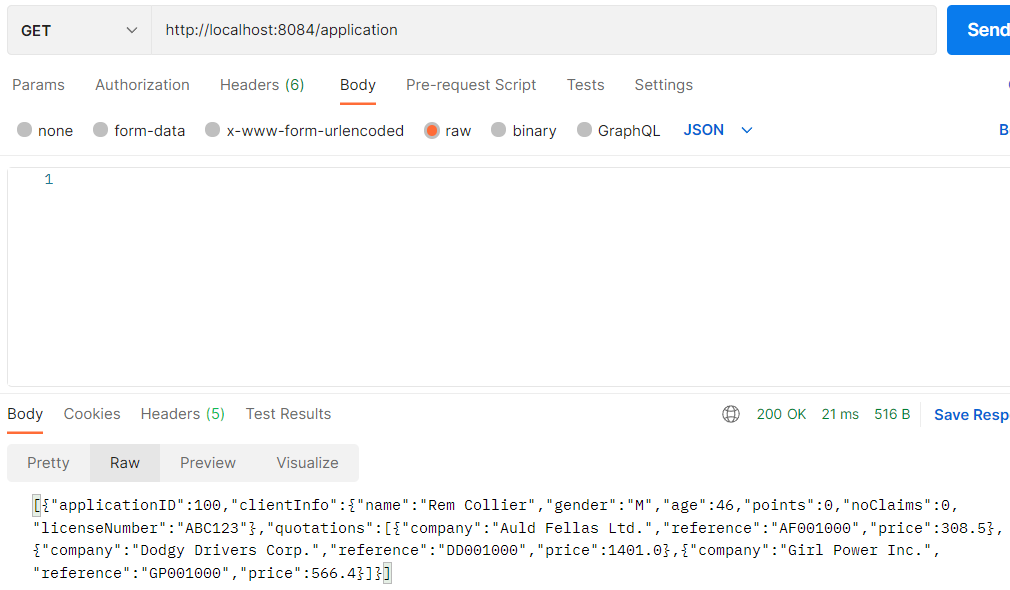
1. Created a starter class in Application
2. Copied the LocalBrokerService and implemented a URI /application with POST mapping that is capable of returning a list of quotations. The list of quotations will be returned as a java bean ClientApplication. Created the java bean and added a Dictionary of quotations that will be used later. Implemented the getQuotations() that is same as the client we implemented before. The getApplications POST mapping will call the getQuotations. Implementing all of this and running all the services we get the following:-



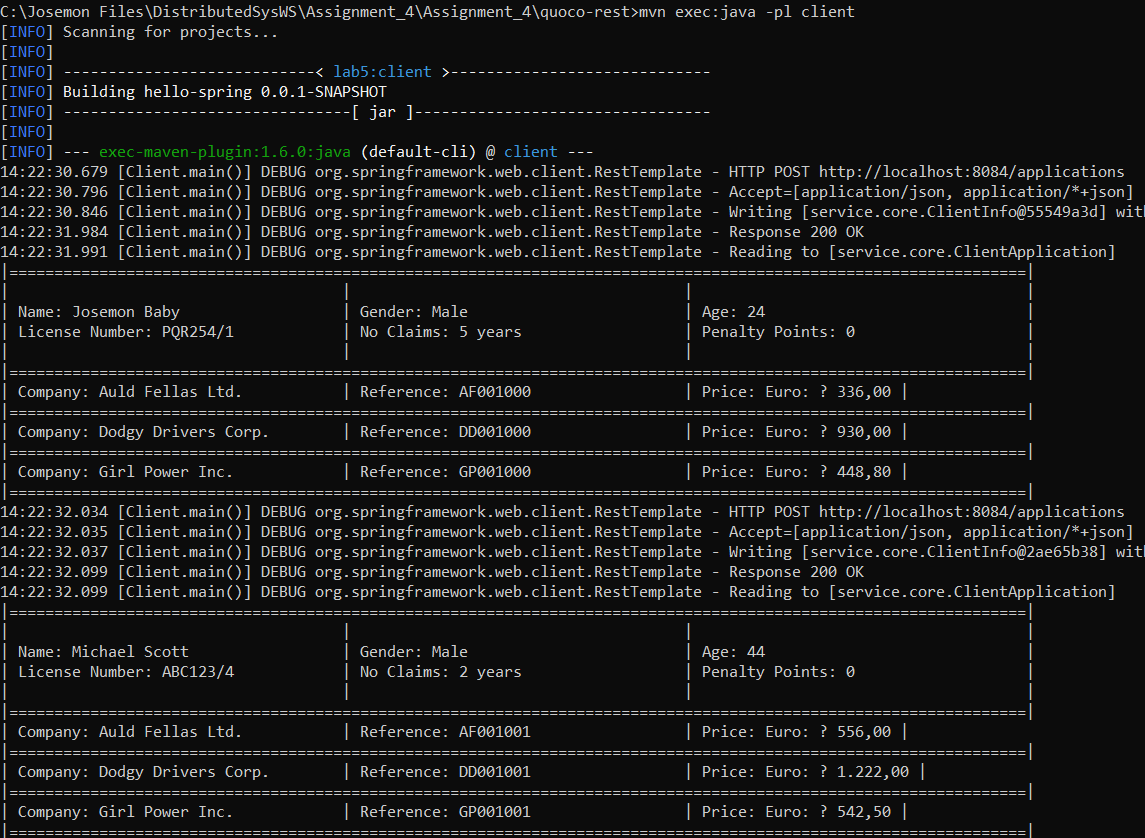
1. Implementing getSpecificApplication() that returns a specific service using the dictionary of applications using the unique ID

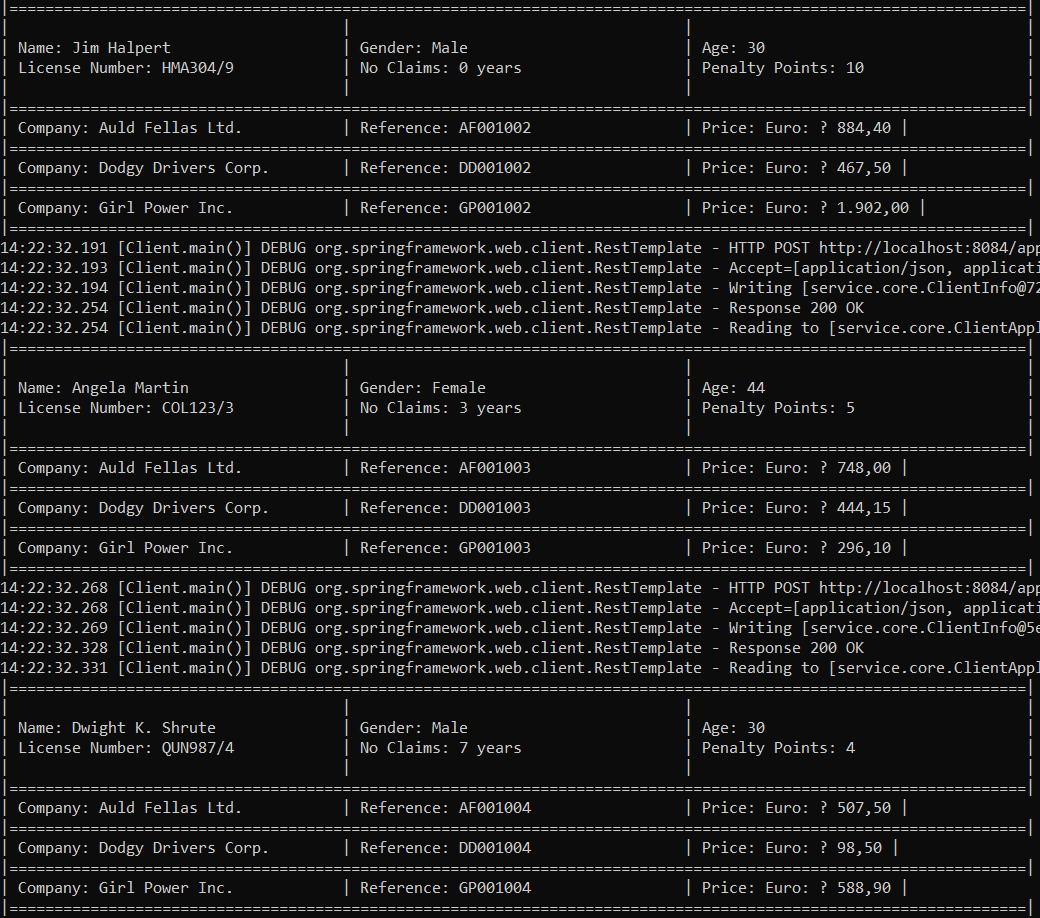


1. Implemented a listOfApplications() method that sends back the whole dictionary of quotations irrespective of the unique seedID.



1. Modified the client main class to loop through all the clients and call the broker for every client and then inside loop through all the quotations received for that client and print them out on console. The result after running all modules is as follows:-



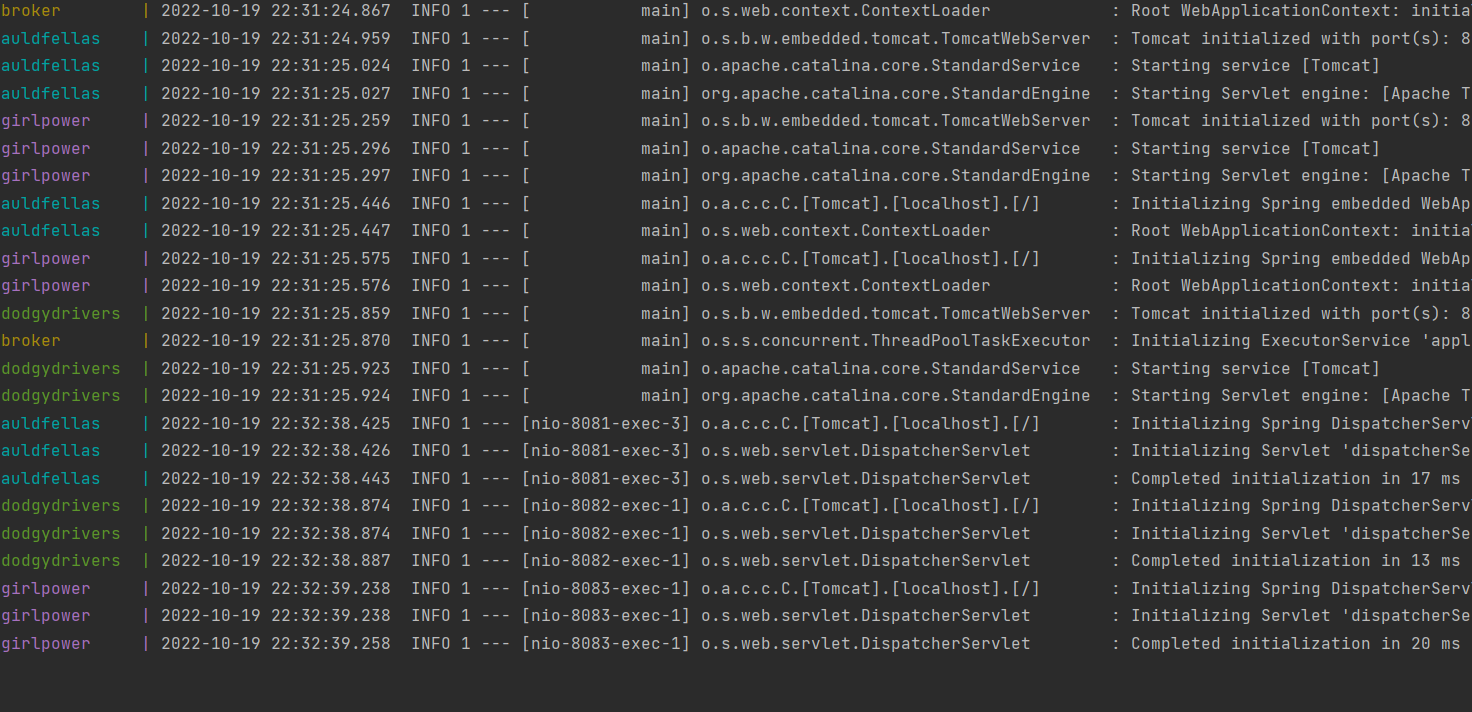
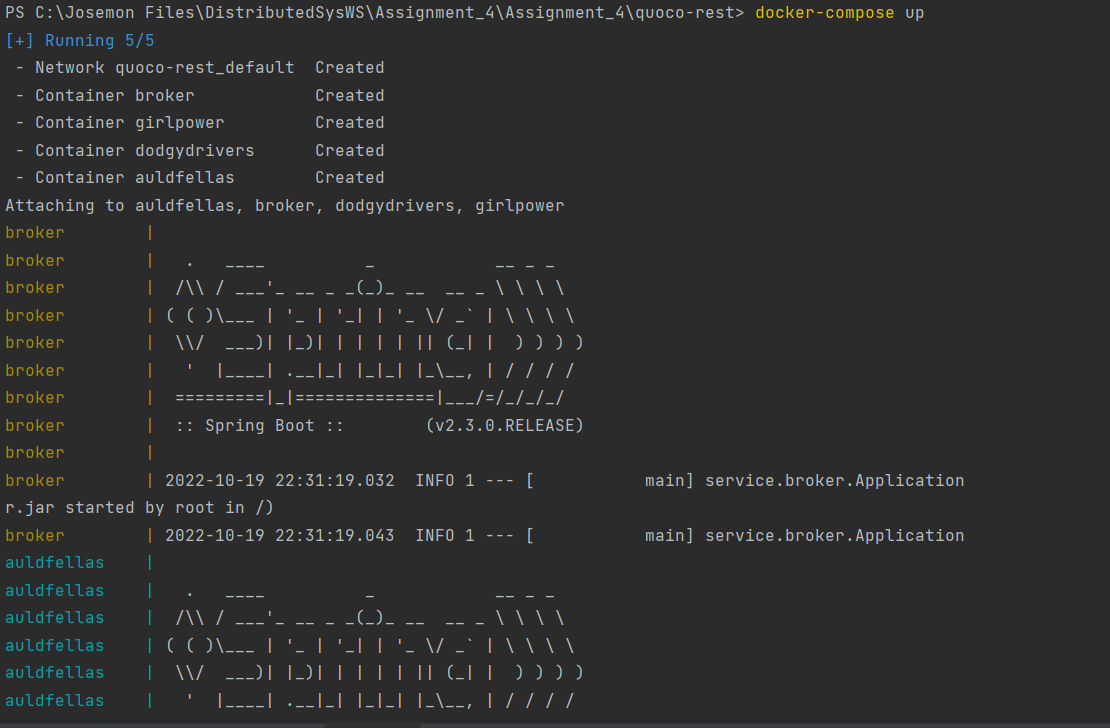


**Task 5:**

Made a default dockerfiles for auldfellas, dodgydrivers and girlpower and made a dockerfile for broker and passing the container names with the ports as command line argument.

Modified the broker starter application to handle arguments and set it to a static variable in LocalBrokerService. Modified LocalBrokerService to handle if nothing is passed in the arguments.

Building the images and running docker-compose up resulted in the following:-



And then running the client resulted in the following:-

