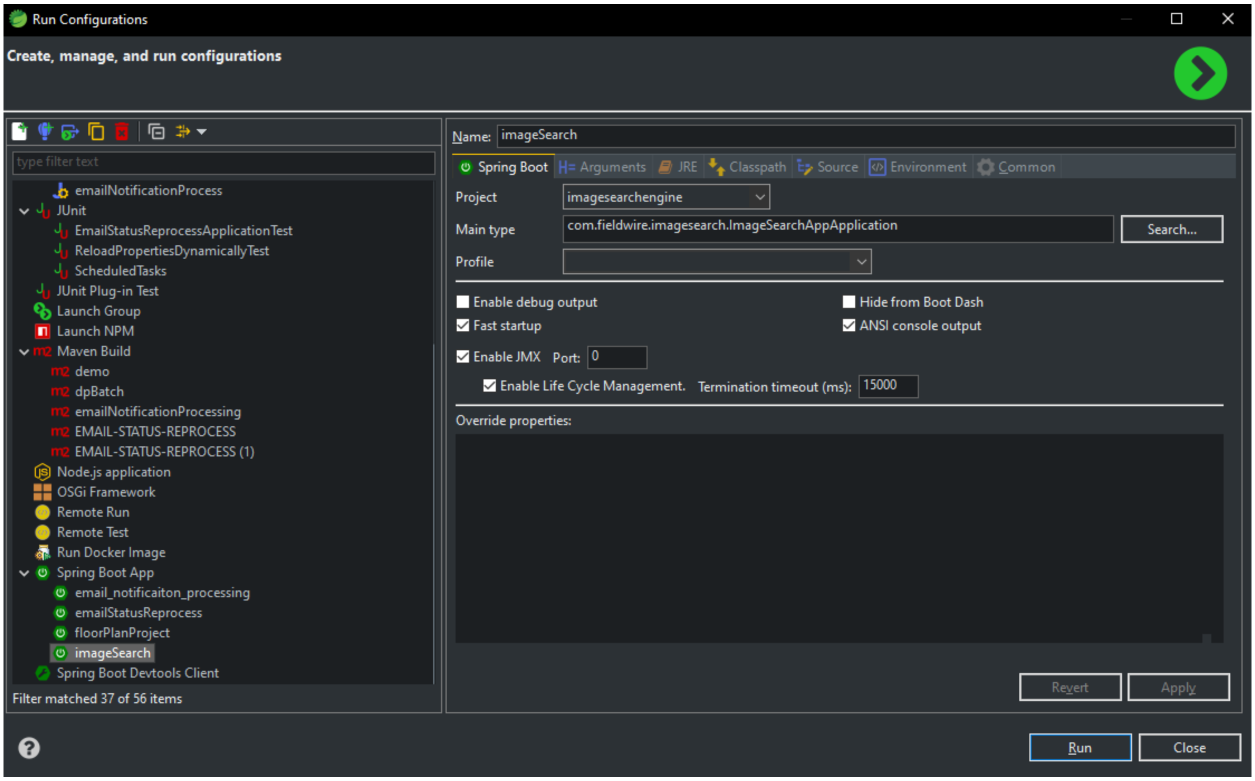
Fieldwire BackEnd project:

Running instructions:

Unzip the attached project in your local machine. Since, it's a maven project so you should be able to import this in any IDE. I prefer to run it in Spring boot Application with java 17.

Set the configurations as below:



Run Configurations

Create, manage, and run configurations

Name: imageSearch

Project: imagesearchengine

Main type: com.fieldwire.imagesearch.ImageSearchAppApplication

Profile:

☐ Enable debug output ☐ Hide from Boot Dash

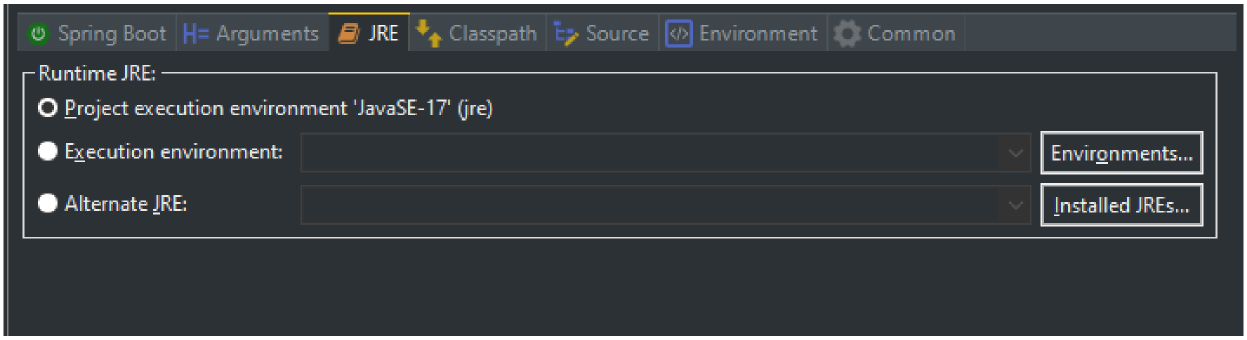
☑ Fast startup ☑ ANSI console output

☑ Enable JMX Port: 0

☑ Enable Life Cycle Management. Termination timeout (ms): 15000

Override properties:

Filter matched 37 of 56 items



Runtime JRE:

● Project execution environment 'JavaSE-17' (jre)

○ Execution environment:

○ Alternate JRE:

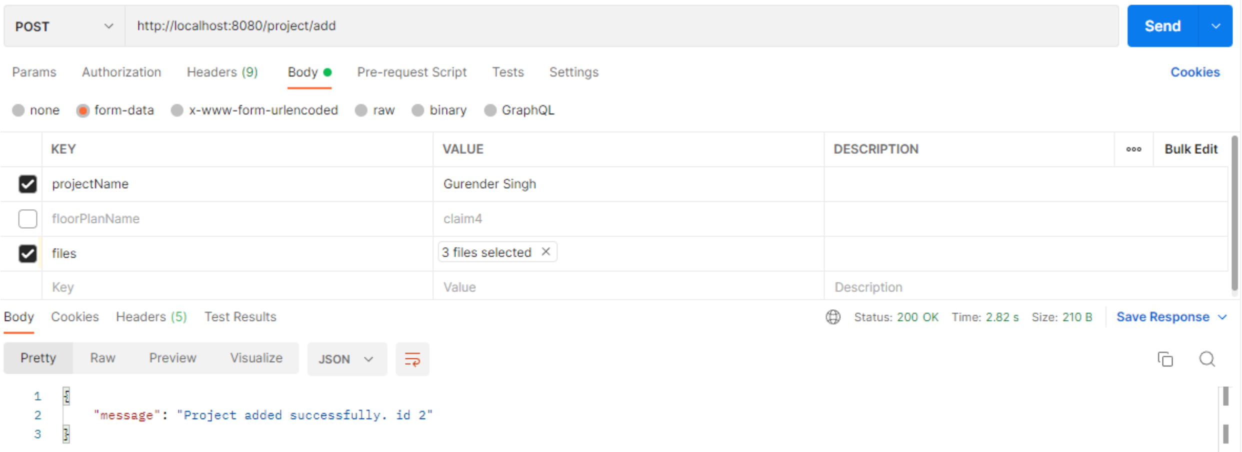
Then run this application.

Then it will run on your local tomcat server port 8080.

In this project we don't have any data in local h2 database when we start the application in our location machine.

We have following REST Api's to maintain the floorplan and project in h2 database. I used postman application to call these api's. For knowing what parms should use provide to the apis, please take a look at class `FloorPlanRestApiController and ProjectRestApiController`:

1. Post: http://localhost:8080`/project/add`

   POST http://localhost:8080/project/add Send

   ○ none ● form-data ○ x-www-form-urlencoded ○ raw ○ binary ○ GraphQL

	KEY	VALUE	DESCRIPTION
☑	projectName	Gurender Singh	
□	floorPlanName	claim4	
☑	files	3 files selected ×	
	Key	Value	Description

   Body · Status: 200 OK Time: 2.82 s Size: 210 B

   ```
   {
       "message": "Project added successfully. id 2"
   }
   ```

   Set the parameters as shown above to insert the floorplan name with default floorplan name as file name under project with projectName: Gurender Singh.
   It will return response 200 if it gets inserted in to h2 database. You can access database in your local machine by going to url: http://localhost:8080/h2-console/login.do?jsessionid=3564c62f10e8527ed11f800bb8f750f9
   You can see the password and user name in application.properties file under src/main/resources.
   Use below query to see the data:
   select * from project_info; SELECT id, floor_plan_name, project_fk, original_image from FLOOR_PLAN;
   This way you can use these apis to update/add/delete projects or floorplans.
2. Put: http://localhost:8080`/project/updateProjectNameByProjectId/`

   ```
   projectName, floorPlanName, files
   ```

   ```
   attache the files as images and  if you would like to give the name to
   floorplan then use floorplanName  = floorplanName1,floorplanName2 and it will
   store floorplan with those specified names.
   ```
3. Get: http://localhost:8080`/project/getAll`

   Provide the count of how man projects you would like to see in response. E.g.

   Limit = 2
4. Get: http://localhost:8080`/getbyid`

   ```
   id = 2  then api call will return the response boby of that project.
   ```
5. Delete: http://localhost:8080`/project/removeById`
6. Post: http://localhost:8080`/project/addNewFloorPlansByProjectId`
7. Delete: http://localhost:8080`/floorplan/removeProjectById`
8. Get: http://localhost:8080`/floorplan/getFloorPlanById`
9. Get: http://localhost:8080`/floorplan/getFloorPlansByProjectId`
10. Post: http://localhost:8080`/floorplan/addFloorPlan`
11. Put: http://localhost:8080`/floorplan/updateFloorPlanByProjectId`

12. Put: http://localhost:8080`/floorplan/moveFloorPlanByProjectId`