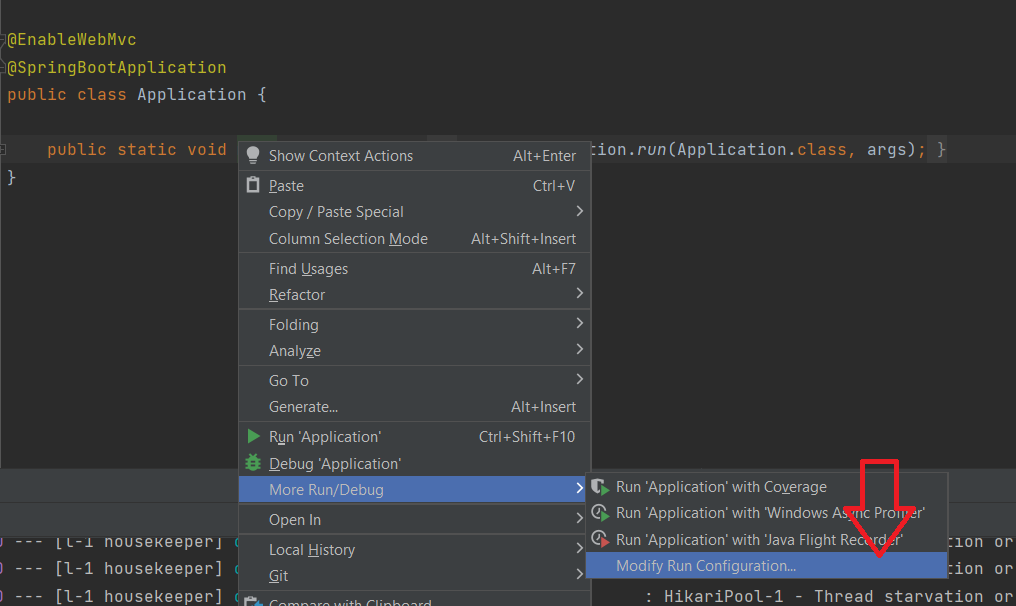
MK Tech Backend Setup.

The backend of the MK Tech site is currently being re-written using SpringBoot and JPA. To get the backend running on your workstation, here are the steps. Feel free to reach out as you run into issues and I’ll see what I can do to help you work past them!

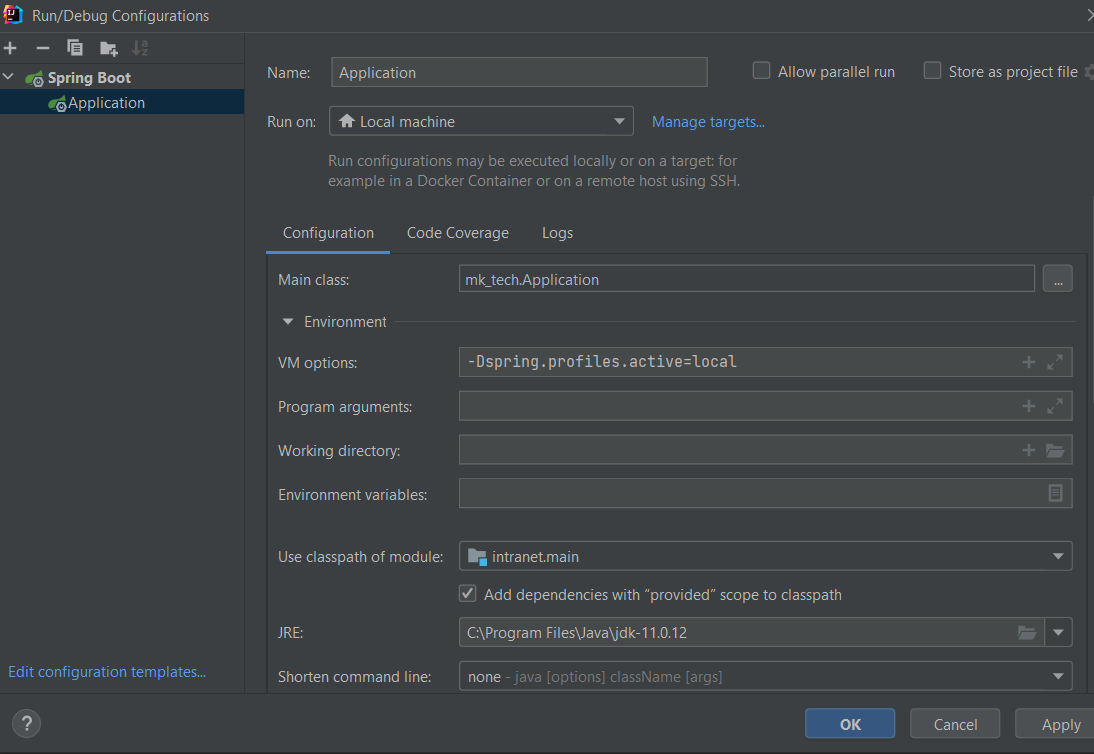
1. Download and Install IntelliJ – Get the Ultimate Edition. I believe it’s free for 30 days, then the company can provide you with a License. Here is the URL to get that downloaded / installed: [Download IntelliJ IDEA: The Capable & Ergonomic Java IDE by JetBrains](https://www.jetbrains.com/idea/download/#section=windows)
2. Download and Install Git: [Install Git | Atlassian Git Tutorial](https://www.atlassian.com/git/tutorials/install-git)
3. Create a folder you would like to work from and pull down the backend code. This can be done by running the command: git clone <https://github.com/melissa-kirkpatrick1/intranet_gradle.git>.
4. Open IntelliJ and go to File -> New -> Project From Existing Sources. Browse to where you downloaded the code and click on the folder: intranet
5. Check the radio button for: Import project from external model – then highlight Gradle and click: Finish
6. Once the project is imported, you’ll want to go to navigate to Application.java. Then right click on the file – and open the Run Configurations as below:



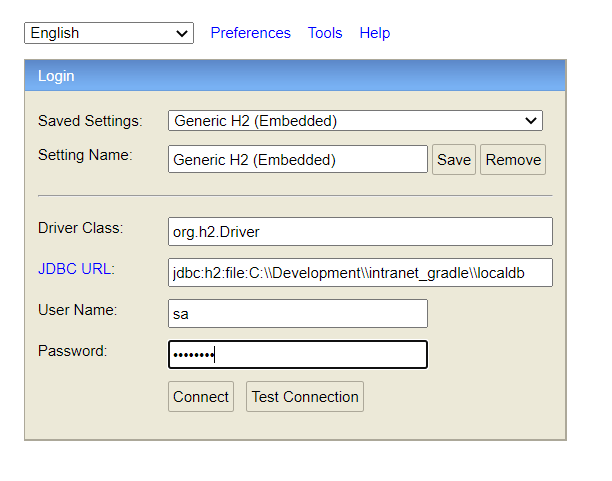
And in the VM options – you’ll want to set the VM Options to: -Dspring.profiles.active=local

This tells Spring to pick up the application-local.properties file (which is attached and you’ll want to drop it in src/main/resources so Spring can find it.

My Run Configurations



1. To run - click Run in the top toolbar – then Run in the menu. You should see a console appear in the bottom section of IntelliJ logging things as the application starts up. If somewhere in the last few lines, you see something like: “Started Application” and / or “Completed initialization” – then the application started successfully
2. Once the application is running – you should be able to go to this URL: [http://localhost: 8443/h2-console/login.jsp](http://localhost:5000/h2-console/login.jsp). This is your database, with SpringBoot – the tables get created on startup based on the objects in the model package. The @Entity annotation does this. The user name / password for the database are specified in the application-local.properties file. I just user = sa and password = password



1. This is just the restful backend – so there’s no UI through these steps, but you can hit the GET APIs in the browser for example: <http://localhost:8443/company-api/contracts> (this returns an array of contracts in your local database)