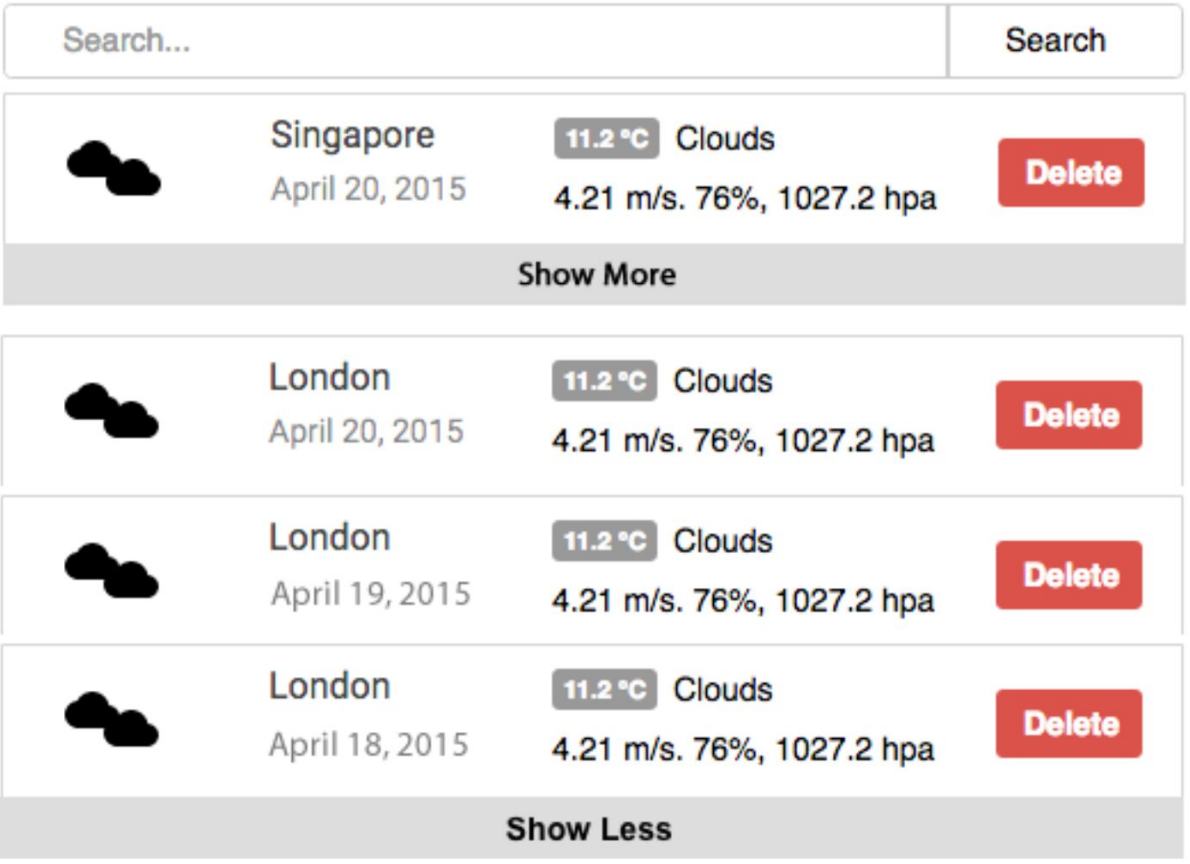
Spring Hibernate Test

Create a Java Spring Hibernate web application that consumes an external API. The web application should:

1. Consume weather API (https://openweathermap.org/api) to retrieve current weather data using city name.
2. One city should have 1-n weather logs.
3. After retrieval of the weather data, all data should be stored into database as weather log for archive purposes.
4. Show all past weather logs grouped by city name.
5. Ability to add new / remove city from the database.
6. See screenshot below for more information.



Requirements

1. Demonstrate coding best practices.
2. Demonstrate understanding of Spring + Hibernate framework.

Assignment Guide

Prerequisite

Open up the boilerplate project and make sure you are able to deploy it on Eclipse's tomcat. You can use Maven command to build the project. The project requires Java 8 and above.

How to run the project

1. Load it up on Eclipse with Maven plugin.
2. Navigate to project properties and ensure that "Is a Tomcat Project" option is check under Tomcat tab.
3. Do a Maven "clean install".
4. Deploy and run project on to Tomcat.
5. Open a browser and navigate to http://localhost:<port number>/list

Things to note

1. Ensure that you understand the requirements of the assignment before continuing.
2. Add in adequate comments to explain your codes. (If you need to write a documentation to explain your solution, please do so)
3. Design and attach your .sql file when you submit your project.
4. You should submit your full code solution and not the war file. Submitting war file alone will be disqualified from the interview.
5. If additional setup is required, please attach a readme.txt that explains the steps to get the assignment working.

Implementation guide

1. Frontend display (Understanding JSTL and JSP concept)
    a. Search implementation.
    b. List of weathers based on countries.
2. API call to retrieve weathers based on city name (Understanding Spring MVC and demonstrating ability to use external libraries)
    a. Make a call to openweathermap with city name.
    b. Consume returned results.
3. Database design; this illustrate the familiarity of ORM concept (Understanding Hibernate)
    a. Database design.
4. Create, retrieve and download of weather logs from database (Understanding the concepts of Spring and Hibernate)
    a. Model and DAO implementation.
5. If you think JSP / JSTL is not sexy enough, you can always deliver the same and demonstrate Spring & Hibernate using SpringBoot with Thymeleaf.
    a. It must be deployable as an uber jar.

Testing guide

1. Identify and explain edge cases.
2. Implement test cases.