Developing the Springboot Application

> Due to our Lab session time constraint, we already provided the code for the app, so you have time to experience other exiting capabilities available in the platform.
>
> This version of the app implements the same logic, exposing the exact same resources, but now using **SpringBoot** as its underline Runtime.
>
> Feel free to go through the source code. But **you don't need to code any Java Class in this section** .
>
> **You can jump right to the Running the Springboot Application Locally** section and just run the Maven commands (or run the Tasks using the IDE Task Manager) to run and test the app inside your DevWorkspace.

In this section you will:

- Develop a REST API with Spring Boot that consumes memory and CPU (the exact same algorithm than before)
- Add a Statistics persistent entity to store metrics in a PostgreSQL database
- Configure the application
- Develop some tests to validate the behavior of the application
- Test and run the application locally
- Check a few metrics locally

You should have a directory called `springboot-app` inside your project repo (`$PROJECT_SOURCE/`). This is the root of the Springboot microservice source code that we will be working on during this this section.

The Spring Boot REST Resource

The Spring Boot application is a simple REST resource that exposes a single endpoint to consume memory and CPU. Before creating the REST resource, let's check the existing main Spring Boot class that will bootstrap the application. Check the following code in the `SpringbootApplication` class, under the `io/containerapps/javaruntime/workshop/springboot` package.

Bootstrapping Spring Boot Class

JAVA

```
package io.containerapps.javaruntime.workshop.springboot;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class SpringbootApplication {

    public static void main(String[] args) {
        SpringApplication.run(SpringbootApplication.class, args);
    }

}
```

The REST resource is defined in the `SpringbootResource` class. Create a new file called `SpringbootResource.java` , under the `io.containerapps.javaruntime.workshop.springboot` directory. Then add the following to the header of this class file (replace any existing content generated by the IDE).

As you can see in the header of the class, the resource is exposed on the `/springboot` path.

Header of the Spring Boot REST Resource

JAVA

```
package io.containerapps.javaruntime.workshop.springboot;

import static java.lang.System.Logger.Level.INFO;
import static java.lang.invoke.MethodHandles.lookup;

import java.lang.System.Logger;
import java.time.Duration;
import java.time.Instant;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;

import org.springframework.http.MediaType;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RequestMapping;
```