Lab 3 - Externalizing Configuration with Spring Boot

Refactoring to Externalize the Config

1. Change to the lab directory (or continue to build off of lab_01 or lab_02):

   ```
   $ cd $COURSE_HOME/day_01/session_01/lab_03/initial/hello-spring-boot
   ```

2. Import the project's `pom.xml` into your editor/IDE of choice (if you haven't in a previous lab).

3. Rename `src/main/resources/application.properties` to `src/main/resources/application.yml`. Into that file, paste the following:

   ```
   greeting: Hello
   ```

4. To the class `io.pivotal.spring.hello.HelloSpringBootApplication`, add a `greeting` field and inject its value:

   ```
   @Value("${greeting}")
   String greeting;
   ```

5. Also `io.pivotal.spring.hello.HelloSpringBootApplication`, change the return statement of `hello()` to the following:

   ```
   return String.format("%s World!", greeting);
   ```

6. Build the application:

```
$ mvn package
```

7. Run the application:

```
$ java -jar target/hello-spring-boot-0.0.1-SNAPSHOT.jar
```

8. Visit the application in the browser (http://localhost:8080), and verify that the output is still the following:

```
Hello World!
```

9. Stop the application.

Using Environment Variables for Config

1. Run the application again, this time setting the `GREETING` environment variable:

```
$ GREETING=Ohai java -jar target/hello-spring-boot-0.0.1-SNAPSHOT.jar
```

2. Visit the application in the browser (http://localhost:8080), and verify that the output has changed to the following:

```
Ohai World!
```

3. Stop the application.

Using Spring Profiles for Config

1. Add a `spanish` profile to `application.yml`. Your finished configuration should reflect the following:

```
greeting: Hello

---

spring:
  profiles: spanish

greeting: Hola
```

2. Build the application:

```
$ mvn package
```

3. Run the application, this time setting the `SPRING_PROFILES_ACTIVE` environment variable:

```
$ SPRING_PROFILES_ACTIVE=spanish java -jar target/hello-spring-boot-0.0.1-SNAPSHOT.jar
```

4. Visit the application in the browser (http://localhost:8080), and verify that the output has changed to the following:

```
Hola World!
```

5. Stop the application.

Resolving Conflicts

1. Run the application, this time setting both the `SPRING_PROFILES_ACTIVE` and `GREETING` environment variables:

```
$ SPRING_PROFILES_ACTIVE=spanish GREETING=Ohai java -jar target/hello-spring-boot-0.0.1-SNAPSHOT.jar
```

2. Visit the application in the browser (http://localhost:8080), and verify that the output has changed to the following:

```
Ohai World!
```

3. Visit http://docs.spring.io/spring-boot/docs/current/reference/html/boot-features-external-config.html to learn more about this outcome and the entire priority scheme for conflict resolution.