1. Module 5 Overview

The goal of the fifth module is testing - from simple, self-contained unit tests all the way to complex, end-to-end live testing of the API.

2. Goals

Understand what Live / API tests are, why they're useful and how to implement one with rest-assured.

3. Lesson Notes

What are API tests?

Simply put - an API test is a test that hits a live/deployed API.

API tests are a lot more end-to-end than integration/unit types of test.

What libraries can we use for API testing?

Library options:

- HttpClient 4

- The Spring RestTemplate
- *rest-assured*
- any other HTTP library

We're going to go forward with *rest-assured*, but other options exist and can be used in a similar way.

Introducing rest-assured for the first Live / API Test

We're going to do a simple test doing a GET with preemptive basic authentication and hitting the API.

After that, we're also going to test a couple of other scenarios as well:

- **scenario 1** - create, extract the URI from the Location header, find one
- **scenario 2** - invalid update - 400 Bad Request

Running a live test

To **run the live tests in Maven**:

```
mvn -e clean verify –Plive
```

Notice that we're using the *verify* phase, not just the *test* phase here.

Also note that we're using the *live* profile.

We can also **run all the live tests manually from the IDE**, via the pre-defined suite: *LiveSuite*

4. Resources

- rest-assured library

- Testing RESTful Web Services made easy using the REST-assured Framework

- The Module 5 Branch of the project on Github