The LockManager App Case Study: Test Driver Implementation

Douglas C. Schmidt
d.schmidt@vanderbilt.edu
www.dre.vanderbilt.edu/~schmidt

Professor of Computer Science
Institute for Software Integrated Systems
Vanderbilt University
Nashville, Tennessee, USA
Learning Objectives in this Part of the Lesson

- Understand the implementation of the LockManagerTest class & associated client code that invoke synchronous methods on the LockManagerController.

See [github.com/douglascraigschmidt/LiveLessons/tree/master/WebMVC/ex5](https://github.com/douglascraigschmidt/LiveLessons/tree/master/WebMVC/ex5)
Implementing the LockManagerTest Driver
Implementing the LockManagerTest Driver

```java
/**
   * This program tests the features of the {@link LockManagerApplication} microservice, which uses Spring WebMVC to
   * provide a distributed lock manager for synchronous Spring client applications using quasi-asynchronous controller methods that
   * return Spring {@link DeferredResult} objects. It also shows how to
   * use the synchronous HTTP interface features in Spring framework 6,
   * which enables the definition of declarative HTTP services using
   * Java interfaces.
   */

@SpringBootTest(classes = LockManagerApplication.class,
               webEnvironment = SpringBootTest.WebEnvironment.DEFINED_PORT)

class LockManagerTests {
    /**
     * The auto-wired {@link LockAPI} that accesses the {@link LockManagerApplication} microservice
     */
```
End of the LockManager App Case Study: Test Driver