The LockManager App Case Study: Implementing the Server Components

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 WebFlux implementation of the LockManagerController & LockManagerService classes in the reactive LockManagerApplication microservice

See [github.com/douglascraigschmidt/LiveLessons/tree/master/WebFlux/ex1](https://github.com/douglascraigschmidt/LiveLessons/tree/master/WebFlux/ex1)
Implementing the LockManager App Server
Implementing the LockManager App Server

public class LockManagerService {
    /**
     * An {@link ArrayBlockingQueue} that limits concurrent access to
     * the fixed number of available locks managed by the {@link LockManagerService}.
     */
    private ArrayBlockingQueue<Lock> mAvailableLocks;

    /**
     * Initialize the {@link Lock} manager.
     *
     * @param permitCount The number of {@link Lock} objects to manage
     * @return A {@link Mono} that emits {@link Boolean#TRUE} if
     * the {@code permitCount} changed the state of the
     * lock manager and {@link Boolean#FALSE} otherwise.
     */
    public Mono<LockManagerResponse>(Integer permitCount) {
        ...
    }

See WebFlux/ex1/src/main/java/edu/vandy/lockmanager/server
End of the LockManager App
Case Study: Implementing the Server Components