The BerraQuotes App Case Study: Overview

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 how functional programming & various Java frameworks are applied in a case study using Spring WebMVC to obtain Yogi Berra quotes

See [github.com/douglascraigschmidt/LiveLessons/tree/master/WebMVC/ex2](https://github.com/douglascraigschmidt/LiveLessons/tree/master/WebMVC/ex2)
Overview of the Berra Quotes App Case Study
Overview of the BerraQuotes App Case Study

• This case study shows how Spring WebMVC can be used to send & receive HTTP GET requests to/from a sequential microservice
Overview of the BerraQuotes App Case Study

- This case study shows how Spring WebMVC can be used to send & receive HTTP GET requests to/from a sequential microservice

**BerraQuotesTest**

The client can send requests individually or in bulk

See WebMVC/ex2/src/test/java/berraquotes/client
Overview of the BerraQuotes App Case Study

• This case study shows how Spring WebMVC can be used to send & receive HTTP GET requests to/from a sequential microservice.

The server (microservice) can receive requests individually or in bulk.

See WebMVC/ex2/src/main/java/server
Overview of the BerraQuotes App Case Study

• This case study shows how Spring WebMVC can be used to send & receive HTTP GET requests to/from a sequential microservice

The BerraQuotesController automatically converts HTTP GET requests into Java types & forwards them to the BerraQuotesService

See WebMVC/ex2/src/main/java/berraquotes/server/BerraQuotesController.java
Overview of the BerraQuotes App Case Study

- This case study shows how Spring WebMVC can be used to send & receive HTTP GET requests to/from a sequential microservice

The BerraQuotesService performs the requested service & returns a List of Yogi Berra quotes

See WebMVC/ex2/src/main/java/berraquotes/server/BerraQuotesService.java
Structure of the BerraQuotes App Project
Structure of the BerraQuotes App Project

- The BerraQuotes App project source code is organized into several packages.

See github.com/douglascraigschmidt/LiveLessons/tree/master/WebMVC/ex2
The BerraQuotes App project source code is organized into several packages:

- main
  - berraquotes
    - Contains the “app” entry point, the controller, & the service
Structure of the BerraQuotes App Project

- The BerraQuotes App project source code is organized into several packages
  - main
    - berraquotes
      - Contains the “app” entry point, the controller, & the service
    - Consolidates various project-specific helper classes & the model
Structure of the BerraQuotes App Project

- The BerraQuotes App project source code is organized into several packages
  - main
    - berraquotes
  - resources
    - Defines various application properties
      - e.g., name & port number
Structure of the BerraQuotes App Project

- The BerraQuotes App project source code is organized into several packages
  - `test`
    - BerraQuotesTest
      - This test driver initiates calls to the BerraQuotes microservice
Structure of the BerraQuotes App Project

- The BerraQuotes App project source code is organized into several packages
  - test
    - BerraQuotesTest
  - client
    - Sends HTTP GET requests to the BerraQuotes microservice
Structure of the BerraQuotes App Project

- The BerraQuotes App project source code is organized into several packages:
  - test
    - BerraQuotesTest
  - client
  - utils
    - Consolidates various reusable helper classes
End of the BerraQuotes App Case Study: Overview