

|  |  |
| --- | --- |
| Employee Name: | MANOJ KUMAR YEKOLLU |
| Training Supervised by: | **KIRSHNA BODDULURI** |
| Training Task #: | **ASSIGNMENT03012019** |
| Training Resource Materials: | <https://www.mkyong.com/spring-boot/spring-rest-hello-world-example/> |
| Training Task Date: | **03/01/2019** |
| Task Due Date: | **03/15/2019** |
| Task Submitted Date: | **03/15/2019** |
| Github link: | **https://github.com/NightFury546/RestDemo.git** |
| Technologies used for Training | * Spring Boot 2.1.2.RELEASE * Spring 5.1.4.RELEASE * Spring Data JPA 2.1.4.RELEASE * H2 In-memory Database 1.4.197 * Tomcat Embed 9.0.14 * JUnit 4.12 * Maven 3 * Java 8 |

**Task Description/Requirement:**

Spring REST Hello World Example

**High Level Synopsis:**

## **3. Spring REST**

3.1 A REST controller to create the following REST API endpoints :

| **HTTP Method** | **URI** | **Description** |
| --- | --- | --- |
| GET | /books | List all books. |
| POST | /books | Save a book. |
| GET | /books/{id} | Find a book where id = {:id}. |
| PUT | /books/{id} | Update a book where id = {:id}, or save it. |
| PATCH | /books/{id} | Update a single field where id = {:id}. |
| DELETE | /books/{id} | Delete a book where id = {:id}. |

Start Spring Boot application, and test the REST API endpoints with curl command.

$ mvn spring-boot:run

Copy

7.1 Find All – GET /books

> curl -v localhost:8080/books

\* Trying ::1...

\* TCP\_NODELAY set

\* Connected to localhost (::1) port 8080 (#0)

> GET /books HTTP/1.1

> Host: localhost:8080

> User-Agent: curl/7.55.1

> Accept: <strong>/</strong>

>

< HTTP/1.1 200

< Content-Type: application/json;charset=UTF-8

< Transfer-Encoding: chunked

< Date: Tue, 19 Feb 2019 06:33:57 GMT

<

[

{"id":1,"name":"A Guide to the Bodhisattva Way of Life","author":"Santideva","price":15.41},

{"id":2,"name":"The Life-Changing Magic of Tidying Up","author":"Marie Kondo","price":9.69},

{"id":3,"name":"Refactoring: Improving the Design of Existing Code","author":"Martin Fowler","price":47.99}

]

Copy

7.2 Find One - GET /books/1

> curl -v localhost:8080/books/1

\* Trying ::1...

\* TCP\_NODELAY set

\* Connected to localhost (::1) port 8080 (#0)

> GET /books/1 HTTP/1.1

> Host: localhost:8080

> User-Agent: curl/7.55.1

> Accept: <strong>/</strong>

>

< HTTP/1.1 200

< Content-Type: application/json;charset=UTF-8

< Transfer-Encoding: chunked

< Date: Tue, 19 Feb 2019 06:37:26 GMT

<

{

"id":1,

"name":"A Guide to the Bodhisattva Way of Life",

"author":"Santideva",

"price":15.41

}

Copy

7.3 Test 404 - GET /books/5

> curl -v localhost:8080/books/5

\* Trying ::1...

\* TCP\_NODELAY set

\* Connected to localhost (::1) port 8080 (#0)

> GET /books/5 HTTP/1.1

> Host: localhost:8080

> User-Agent: curl/7.55.1

> Accept: <strong>/</strong>

>

< HTTP/1.1 404

< Content-Type: application/json;charset=UTF-8

< Transfer-Encoding: chunked

< Date: Tue, 19 Feb 2019 06:38:45 GMT

<

{

"timestamp":"2019-02-19T06:38:45.743+0000",

"status":404,

"error":"Not Found",

"message":"Book id not found : 5",

"path":"/books/5"

}

Copy

7.4 Test Save - POST /books -d {json}

#Run this on Windows, need \"

> curl -v -X POST localhost:8080/books -H "Content-type:application/json" -d "{\"name\":\"Spring REST tutorials\",\"author\":\"mkyong\",\"price\":\"9.99\"}"

Note: Unnecessary use of -X or --request, POST is already inferred.

\* Trying ::1...

\* TCP\_NODELAY set

\* Connected to localhost (::1) port 8080 (#0)

> POST /books HTTP/1.1

> Host: localhost:8080

> User-Agent: curl/7.55.1

> Accept: <strong>/</strong>

> Content-type:application/json

> Content-Length: 65

>

\* upload completely sent off: 65 out of 65 bytes

< HTTP/1.1 201

< Content-Type: application/json;charset=UTF-8

< Transfer-Encoding: chunked

< Date: Tue, 19 Feb 2019 07:33:01 GMT

<

{

"id":4,

"name":

"Spring REST tutorials",

"author":"mkyong",

"price":9.99

}

Copy

7.5 Test Update - PUT /books/4 -d {json}

> curl -v -X PUT localhost:8080/books/4 -H "Content-type:application/json" -d "{\"name\":\"Spring Forever\",\"author\":\"pivotal\",\"price\":\"9.99\"}"

\* Trying ::1...

\* TCP\_NODELAY set

\* Connected to localhost (::1) port 8080 (#0)

> PUT /books/4 HTTP/1.1

> Host: localhost:8080

> User-Agent: curl/7.55.1

> Accept: <strong>/</strong>

> Content-type:application/json

> Content-Length: 59

>

\* upload completely sent off: 59 out of 59 bytes

< HTTP/1.1 200

< Content-Type: application/json;charset=UTF-8

< Transfer-Encoding: chunked

< Date: Tue, 19 Feb 2019 07:36:49 GMT

<

{

"id":4,

"name":"Spring Forever",

"author":"pivotal",

"price":9.99

}

> curl localhost:8080/books/4

{"id":4,"name":"Spring Forever","author":"pivotal","price":9.99}

Copy

7.6 Test Update a 'author' field - PATCH /books/4 -d {json}

> curl -v -X PATCH localhost:8080/books/4 -H "Content-type:application/json" -d "{\"author\":\"oracle\"}"

\* Trying ::1...

\* TCP\_NODELAY set

\* Connected to localhost (::1) port 8080 (#0)

> PATCH /books/4 HTTP/1.1

> Host: localhost:8080

> User-Agent: curl/7.55.1

> Accept: <strong>/</strong>

> Content-type:application/json

> Content-Length: 19

>

\* upload completely sent off: 19 out of 19 bytes

< HTTP/1.1 200

< Content-Type: application/json;charset=UTF-8

< Transfer-Encoding: chunked

< Date: Tue, 19 Feb 2019 07:39:53 GMT

{

"id":4,

"name":"Spring Forever",

"author":"oracle",

"price":9.99

}

Copy

7.6.1 Test Update a 'name' field - PATCH /books/4 -d {json}

> curl -v -X PATCH localhost:8080/books/4 -H "Content-type:application/json" -d "{\"name\":\"New Spring REST\"}"

\* Trying ::1...

\* TCP\_NODELAY set

\* Connected to localhost (::1) port 8080 (#0)

> PATCH /books/4 HTTP/1.1

> Host: localhost:8080

> User-Agent: curl/7.55.1

> Accept: <strong>/</strong>

> Content-type:application/json

> Content-Length: 26

>

\* upload completely sent off: 26 out of 26 bytes

< HTTP/1.1 405

< Content-Type: application/json;charset=UTF-8

< Transfer-Encoding: chunked

< Date: Tue, 19 Feb 2019 07:40:47 GMT

<

{

"timestamp":"2019-02-19T07:40:47.740+0000",

"status":405,

"error":"Method Not Allowed",

"message":"Field [name] update is not allow.",

"path":"/books/4"

}

Copy

7.7 Test delete - DELETE /books/4

> curl -v -X DELETE localhost:8080/books/4

\* Trying ::1...

\* TCP\_NODELAY set

\* Connected to localhost (::1) port 8080 (#0)

> DELETE /books/4 HTTP/1.1

> Host: localhost:8080

> User-Agent: curl/7.55.1

> Accept: <strong>/</strong>

>

< HTTP/1.1 200

< Content-Length: 0

< Date: Tue, 19 Feb 2019 07:44:24 GMT

<

> curl localhost:8080/books/4

{

"timestamp":"2019-02-19T07:44:39.432+0000",

"status":404,

"error":"Not Found",

"message":"Book id not found : 4",

"path":"/books/4"

}

> curl localhost:8080/books

[

{"id":1,"name":"A Guide to the Bodhisattva Way of Life","author":"Santideva","price":15.41},

{"id":2,"name":"The Life-Changing Magic of Tidying Up","author":"Marie Kondo","price":9.69},

{"id":3,"name":"Refactoring: Improving the Design of Existing Code","author":"Martin Fowler","price":47.99}

]

**I acknowledge that this document can be supplied to USCIS in compliance with CPT/OPT/STEM OPT audit:**

**Output:**