

jdbc:h2:mem:testdb

BOOK

ID

AUTHOR

GENRE

PAGES

TITLE

Indexes

INFORMATION_SCHEMA

Sequences

Users

H2 1.4.200 (2019-10-14)

Max rows: 1000

Auto complete

Auto select

On

Run

Run Selected

Auto complete

Clear

SQL statement:

SELECT * FROM BOOK;

SELECT * FROM BOOK;

ID	AUTHOR	GENRE	PAGES	TITLE
1	Mr Thomas	Thriller	200	Life of Coding
2	Mr Franklin	Crime	200	Life of Grime
3	Mr Smith	Horror	400	Life of Crime

(3 rows, 1 ms)

Edit

Finished after 5.716 seconds

Runs: 11/11

Errors: 0

Failures: 0

- ServiceUnitTest [Runner: JUnit 5] (0.088 s)
 - testGetById() (0.063 s)
 - testCreate() (0.004 s)
 - testDelete() (0.004 s)
 - testGetAll() (0.009 s)
 - testUpdate() (0.004 s)
- BookIntegrationTest [Runner: JUnit 5] (0.320 s)
 - testGetById() (0.131 s)
 - testCreate() (0.049 s)
 - testDelete() (0.096 s)
 - testGetAll() (0.029 s)
 - testUpdate() (0.013 s)
- BookShelfApplicationTests [Runner: JUnit 5] (0.002 s)
 - contextLoads() (0.002 s)

Failure Trace

localhost:8080/box

localhost:8080/box

localhost:8080/box

localhost:8080/box

+

...

No Environment



POST

localhost:8080/book/create

Params

Send

Save

Authorization

Headers (1)

Body

Pre-request Script

Tests

Code

☐ form-data☐ x-www-form-urlencoded☒ raw☐ binary

JSON (application/json)

```
1 {  
2   "title": "life of Coding",  
3   "author": "Mr f",  
4   "pages": 200,  
5   "genre": "Thriller"  
6 }
```

Body

Cookies

Headers (5)

Test Results

Status: 201 Created

Time: 131 ms

Pretty

Raw

Preview

JSON



```
1 {  
2   "id": 1,  
3   "title": "life of Coding",  
4   "author": "Mr f",  
5   "pages": 200,  
6   "genre": "Thriller"  
7 }
```

localhost:8080/bo

localhost:8080/bo

localhost:8080/bo

localhost:8080/bo

+

...

No Environment

GET

localhost:8080/book/books

Params

Send

Save

Authorization

Headers

Body

Pre-request Script

Tests

Code

Type

No Auth

Body

Cookies

Headers (5)

Test Results

Status: 200 OK

Time: 123 ms

Pretty

Raw

Preview

JSON



```
1 [
2   {
3     "id": 1,
4     "title": "life of Coding",
5     "author": "Mr f",
6     "pages": 200,
7     "genre": "Thriller"
8   }
9 ]
```

PUT localhost:8080/book/update/1 Params Send Save

Authorization Headers (1) Body Pre-request Script Tests Code

form-data x-www-form-urlencoded raw binary JSON (application/json)

```
1 {  
2   "title": "Python Coding 2",  
3   "author": "Mr T",  
4   "pages": 200,  
5   "genre": "Thriller"  
6 }
```

Body Cookies Headers (5) Test Results Status: 202 Accepted Time: 28 ms

Pretty Raw Preview JSON

```
1 {  
2   "id": 1,  
3   "title": "Python Coding 2",  
4   "author": "Mr T",  
5   "pages": 200,  
6   "genre": "Thriller"  
7 }
```

DELETE

localhost:8080/book/remove/1

Params

Send

Save

AuthorizationHeaders (1)BodyPre-request ScriptTestsCode

☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary JSON (application/json)

1

BodyCookiesHeaders (3)Test ResultsStatus: 204 No ContentTime: 42 ms

PrettyRawPreviewText

1

Package Explorer JUnit x

Finished after 4.41 seconds

Runs: 5/5 Errors: 0 Failures: 0

BookIntegrationTest [Runner: JUnit 5] (0.474 s)

- testGetById() (0.272 s)
- testCreate() (0.051 s)
- testDelete() (0.103 s)
- testGetAll() (0.031 s)
- testUpdate() (0.014 s)

Failure Trace

```
84 @Test
85 void testUpdate() throws Exception {
86     final Book testBook = new Book(1, "Design Patterns", "Freeman", 300, "Educ
87     final String testBookAsJSON = this.mapper.writeValueAsString(testBook);
88
89     RequestBuilder request = put("/book/update/1").contentType(MediaType.APPLI
90
91     ResultMatcher checkStatus = status().isAccepted();
92     ResultMatcher checkContent = content().json(testBookAsJSON);
93
94     this.mvc.perform(request).andExpect(checkStatus).andExpect(checkContent);
95 }
96
97 @Test
98 void testDelete() throws Exception {
99     this.mvc.perform(delete("/book/remove/1")).andExpect(status().isNoContent(
100 }
101
102 void testCreateAbridged() throws Exception {
103     final String testBookAsJSON = this.mapper
104         .writeValueAsString(new Book(null, "Design Patterns", "Freeman", 30
105     final String savedBookAsJSON = this.mapper
106         .writeValueAsString(new Book(1, "Design Patterns", "Freeman", 300,
107
108     this.mvc.perform(post("/book/create").contentType(MediaType.APPLICATION_J
109         .andExpect(status().isCreated()).andExpect(content().json(savedBook
```

Problems Javadoc Declaration Console x Terminal 1

<terminated> com.qa.bookshelf.rest [JUnit] /Applications/SpringToolSuite4.app/Contents/Eclipse/plugins/org.eclipse.justj.openjdk.hotspot.jre.full.macosx.x86_64_16.0.2.v20210721-1149/jre/bin/java (15 Oct 2021-10-15 11:20:53.461 INFO 62644 --- [ionShutdownHook] j.LocalContainerEntityManagerFac

2021-10-15 11:20:53.461 INFO 62644 --- [ionShutdownHook] .SchemaDropperImpl\$DelayedDropAc

Hibernate: drop table if exists book CASCADE

2021-10-15 11:20:53.462 WARN 62644 --- [ionShutdownHook] o.h.engine.jdbc.spi.SqlException

2021-10-15 11:20:53.462 ERROR 62644 --- [ionShutdownHook] o.h.engine.jdbc.spi.SqlException

2021-10-15 11:20:53.463 WARN 62644 --- [ionShutdownHook] o.h.engine.jdbc.spi.SqlException

2021-10-15 11:20:53.463 ERROR 62644 --- [ionShutdownHook] o.h.engine.jdbc.spi.SqlException

2021-10-15 11:20:53.463 WARN 62644 --- [ionShutdownHook] o.s.b.f.support.DisposableBeanAd

2021-10-15 11:20:53.463 INFO 62644 --- [ionShutdownHook] com.zaxxer.hikari.HikariDataSour

2021-10-15 11:20:53.465 INFO 62644 --- [ionShutdownHook] com.zaxxer.hikari.HikariDataSour

Package Explorer | JUnit

Finished after 5.098 seconds

Runs: 11/11 | Errors: 0 | Failures: 0

- ServiceUnitTest [Runner: JUnit 5] (0.079 s)
 - testGetById() (0.059 s)
 - testCreate() (0.003 s)
 - testDelete() (0.003 s)
 - testGetAll() (0.008 s)
 - testUpdate() (0.003 s)
- BookIntegrationTest [Runner: JUnit 5] (0.317 s)
 - testGetById() (0.129 s)
 - testCreate() (0.049 s)
 - testDelete() (0.096 s)
 - testGetAll() (0.029 s)
 - testUpdate() (0.012 s)
- BookShelfApplicationTests [Runner: JUnit 5] (0.002 s)
 - contextLoads() (0.002 s)

Failure Trace

```

13
74 Mockito.when(this.repo.findById(id)).thenReturn(optionalBook);
75 Mockito.when(this.repo.save(newBook)).thenReturn(newBook);
76
77 assertThat(this.service.updateBook(book.getId(), newBook)).isEqualTo(newBook);
78
79 Mockito.verify(this.repo, Mockito.times(1)).findById(id);
80 Mockito.verify(this.repo, Mockito.times(1)).save(newBook);
81 }
82
83 @Test
84 void testDelete() {
85     final Integer id = 1;
86
87     Mockito.when(this.repo.existsById(id)).thenReturn(false);
88
89     assertThat(this.service.removeBook(id)).isEqualTo(true);
90
91     Mockito.verify(this.repo, Mockito.times(1)).existsById(id);
92 }
93
94 }
95

```

Problems | Javadoc | Declaration | Console | Terminal 1 X

Lizs-MacBook-Pro.local X

```

[test-2 3c52cea] get all service unit test - passed
1 file changed, 13 insertions(+)
→ Book-Shelf git:(test-2) x git add .
→ Book-Shelf git:(test-2) x git commit -m "update service unit test - passed"
[test-2 bc60886] update service unit test - passed
2 files changed, 40 insertions(+)
→ Book-Shelf git:(test-2) git add .
→ Book-Shelf git:(test-2) x git commit -m "create service unit test - passed"
[test-2 81aaadd] create service unit test - passed
1 file changed, 13 insertions(+), 1 deletion(-)
→ Book-Shelf git:(test-2) git add .
→ Book-Shelf git:(test-2) x git commit -m "delete service unit tested - passed"
[test-2 1d36c8b] delete service unit tested - passed
1 file changed, 11 insertions(+)
→ Book-Shelf git:(test-2)

```

Boot Dashboard X

Type tags, projects, or working set names to match (incl. * and >)

> local