

№	Название	Предусловие	Шаги воспроизведения	Ожидаемый результат	Фактический результат
1	Книга добавляется с незаполненным полем name		<pre>1.Отправить Post запрос: @Test void AddBookNameNotFiled(){     given()         .body("{             \"name\": \"\",             \"author\": \"Керри Мнорозков\",             \"isElectronicBook\": true,             \"year\": 2012         }")         .contentType(ContentType.JSON)         .post("http://localhost:5000/api/books")         .then()         .log().all()         .assertThat()             .statusCode(400)             .statusLine("HTTP/1.1 400 BAD REQUEST")             .statusLine(containsString("BAD REQUEST"))             .body("error", Matchers.is("Required name field is not filled")); }</pre>	<pre>Книга не добавилась: ответ: 400 BAD REQUEST. {   "error": "Required name field is not filled " }</pre>	<pre>Книга добавилась: ответ: HTTP/1.1 201 CREATED Server: Werkzeug/2.2.2 Python/3.8.16 Date: Tue, 09 May 2023 06:30:42 GMT Content-Type: application/json Content-Length: 187 Connection: close  {   "book": {     "author": "Керри Мнорозков",     "id": 3,     "isElectronicBook": true,     "name": "",     "year": 2012   } }</pre>
	Критичность				
	Приоритет				
№	Название	Предусловие	Шаги воспроизведения	Ожидаемый результат	Фактический результат
2	Книга добавляется если поле name заполнить пробелами		<pre>1.Отправить Post запрос: @Test void AddBookNameFieldWithSpaces(){     given()         .body("{             \"name\": \" \",             \"author\": \"Томас Кормен\",             \"isElectronicBook\": true,             \"year\": 2022         }")         .contentType(ContentType.JSON)         .post("http://localhost:5000/api/books")         .then()         .log().all()         .assertThat()             .statusCode(400)             .statusLine("HTTP/1.1 400 BAD REQUEST")             .statusLine(containsString("BAD REQUEST"))             .body("error", Matchers.is("Required name field is not filled")); }</pre>	<pre>Книга не добавилась: ответ: 400 BAD REQUEST. {   "error": "Required name field is not filled " }</pre>	<pre>Книга добавилась: ответ: HTTP/1.1 201 CREATED Server: Werkzeug/2.2.2 Python/3.8.16 Date: Tue, 09 May 2023 06:34:48 GMT Content-Type: application/json Content-Length: 148 Connection: close  {   "book": {     "author": "Томас Кормен",     "id": 4,     "isElectronicBook": true,     "name": " ",     "year": 2022   } }</pre>
	Критичность				
	Приоритет				
№	Название	Предусловие	Шаги воспроизведения	Ожидаемый результат	Фактический результат
3	Книга добавляется если в поле year указать не существующую дату		<pre>1.Отправить Post запрос: @Test void AddBookInvalidDate(){     given()         .body("{             \"name\": \"1945\",             \"year\": -1         }")         .contentType(ContentType.JSON)         .post("http://localhost:5000/api/books")         .then()         .log().all()         .assertThat()             .statusCode(400)             .statusLine("HTTP/1.1 400 BAD REQUEST")             .statusLine(containsString("BAD REQUEST"))             .body("error", Matchers.is("Invalid date format")); }</pre>	<pre>Книга не добавилась: ответ: 400 BAD REQUEST. {   "error": "Invalid date format" }</pre>	<pre>Книга добавилась: ответ: HTTP/1.1 201 CREATED Server: Werkzeug/2.2.2 Python/3.8.16 Date: Tue, 09 May 2023 06:35:48 GMT Content-Type: application/json Content-Length: 79 Connection: close  {   "book": {     "author": "",     "id": 5,     "isElectronicBook": false,     "name": "1945",     "year": -1   } }</pre>
	Критичность				
	Приоритет				
№	Название	Предусловие	Шаги воспроизведения	Ожидаемый результат	Фактический результат
4	Книга добавляется если в поле year указать дату больше текущей.		<pre>1.Отправить Post запрос: @Test void AddBookYearOfFiveCharacters(){     given()         .body("{             \"name\": \"Год 2024 еще не настал\",             \"year\": 2024         }")         .contentType(ContentType.JSON)         .post("http://localhost:5000/api/books")         .then()         .log().all()         .assertThat()             .statusCode(400)             .statusLine("HTTP/1.1 400 CREATED")             .statusLine(containsString("BAD REQUEST"))             .body("error", Matchers.is("Invalid date format")); }</pre>	<pre>Книга не добавилась: ответ: 400 BAD REQUEST. {   "error": "Invalid date format" }</pre>	<pre>Книга добавилась: ответ: HTTP/1.1 201 CREATED Server: Werkzeug/2.2.2 Python/3.8.16 Date: Tue, 09 May 2023 06:36:30 GMT Content-Type: application/json Content-Length: 169 Connection: close  {   "book": {     "author": "",     "id": 6,     "isElectronicBook": false,     "name": "Год 2024 еще не настал",     "year": 2024   } }</pre>
	Критичность				
	Приоритет				
№	Название	Предусловие	Шаги воспроизведения	Ожидаемый результат	Фактический результат
5	Книга успешно редактируется если не заполнить name		<pre>1.Отправить Put запрос: @Test void UpdateBookNameNotFiled(){     String id = given()         .body("{             \"name\": \"Сказка про Шрека\",             \"author\": \"Ю.Оган\",             \"isElectronicBook\": false,             \"year\": 2000         }")         .contentType(ContentType.JSON)         .post("http://localhost:5000/api/books")         .then()         .log().all()         .assertThat()             .statusCode(201)             .statusLine("HTTP/1.1 201 CREATED")             .statusLine(containsString("CREATED"))             .extract()             .jsonPath()             .get("book.id")             .toString();      given()         .body("{             \"name\": \"\",             \"author\": \"Фюона\",             \"isElectronicBook\": true,             \"year\": 2001         }")         .contentType(ContentType.JSON)         .put("http://localhost:5000/api/books/"+id)         .then()         .log().all()         .assertThat()             .statusCode(400)             .statusLine("HTTP/1.1 400 BAD REQUEST")             .statusLine(containsString("BAD REQUEST"))             .body("error", Matchers.is("Required name field is not filled")); }</pre>	<pre>Книга не добавилась: ответ: 400 BAD REQUEST. {   "error": "Required name field is not filled " }</pre>	<pre>Книга отредактировался: ответ HTTP/1.1 200 OK Server: Werkzeug/2.2.2 Python/3.8.16 Date: Tue, 09 May 2023 06:37:25 GMT Content-Type: application/json Content-Length: 106 Connection: close  {   "book": {     "author": "Фюона",     "id": 7,     "isElectronicBook": true,     "name": "",     "year": 2001   } }</pre>
	Критичность				
	Приоритет				
№	Название	Предусловие	Шаги воспроизведения	Ожидаемый результат	Фактический результат
6	Книга успешно редактируется если заполнить name пробелами.		<pre>1.Отправить Put запрос: @Test void UpdateBookNameFieldWithSpaces(){     String id = given()         .body("{             \"name\": \"Король лев\",             \"author\": \"Тимон и Пумба\",             \"isElectronicBook\": false,             \"year\": 1996         }")         .contentType(ContentType.JSON)         .post("http://localhost:5000/api/books")         .then()         .log().all()         .statusCode(201)         .statusLine("HTTP/1.1 201 CREATED")         .statusLine(containsString("CREATED"))         .extract()         .jsonPath()         .get("book.id")         .toString();      given()         .body("{             \"name\": \"\",             \"author\": \"Тимон и пумба\",             \"isElectronicBook\": true,             \"year\": 1947         }")         .contentType(ContentType.JSON)         .put("http://localhost:5000/api/books/"+id)         .then()         .log().all()         .assertThat()             .statusCode(400)             .statusLine("HTTP/1.1 400 BAD REQUEST")             .statusLine(containsString("BAD REQUEST"))             .body("error", Matchers.is("Required name field is not filled")); }</pre>	<pre>Книга не добавилась: ответ: 400 BAD REQUEST. {   "error": "Required name field is not filled " }</pre>	<pre>Книга отредактировался: ответ HTTP/1.1 200 OK Server: Werkzeug/2.2.2 Python/3.8.16 Date: Tue, 09 May 2023 06:38:54 GMT Content-Type: application/json Content-Length: 147 Connection: close  {   "book": {     "author": "Тимон и пумба",     "id": 8,     "isElectronicBook": true,     "name": " ",     "year": 1947   } }</pre>
	Критичность				
	Приоритет				