Министерство науки и высшего образования Российской Федерации   
Федеральное государственное бюджетное образовательное   
учреждение высшего образования   
«Алтайский государственный технический   
университет им. И.И. Ползунова»

Факультет информационных технологий

Кафедра прикладной математики

Отчёт защищён с оценкой\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Преподаватель Третьяков А.А.

«\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2022 г.

Отчёт

по лабораторной работе № 5

«AJAX»

по дисциплине «Современные средства разработки Web приложений»

Студент группы ПИ-92 Шульпов В.М.

Старший преподаватель Третьяков А.А.

Барнаул 2022

Код:

Search-books.js

const ***requestURL*** = 'http://localhost:8081/api/books/';  
  
//отправить запрос общего вида  
function sendRequest(method, url) {  
 const headers = {  
 'Content-Type': 'application/json'  
 }  
 return fetch(url, {  
 method: method,  
 headers: headers  
 }).then(response => {return response.json()})  
}  
  
  
let ***inputSearch*** = ***document***.getElementById('book-search')  
let ***booksContainer*** = ***document***.getElementById('books-container')  
  
//удалить потомков блока  
function removeChildren(parent) {  
 while (parent.firstChild) {  
 parent.removeChild(parent.firstChild);  
 }  
}  
  
//обновить спискок книг путем динамического добавления блоков  
function generateBooksContainers(books) {  
 removeChildren(***booksContainer***)  
 books.forEach(book => {  
 let div = ***document***.createElement('div');  
 let html = `<a href="/books/${book.id}">  
 <span>${book.name}</span>,  
 <span>${book.author}</span>,  
 <span>${book.year}</span>  
 </a>`  
 div.insertAdjacentHTML('beforeend', html);  
 ***booksContainer***.appendChild(div);  
 })  
}  
  
//добавить обработчик события для input'a поиска  
***inputSearch***.addEventListener('input', (e) => {  
 const fullRequestUrl = ***requestURL*** + ***inputSearch***.value;  
 sendRequest('GET', fullRequestUrl)  
 .then(data => {  
 generateBooksContainers(data)  
 })  
 .catch(err => ***console***.log(err))  
})

PeopleRestController.java

@RestController  
@RequestMapping("api/books")  
public class BooksRestController {  
 private final BookDAO bookDAO;  
  
 @Autowired  
 public BooksRestController(BookDAO bookDAO) {  
 this.bookDAO = bookDAO;  
 }  
  
 @GetMapping()  
 public List<Book> allBooks() {  
 return bookDAO.index();  
 }  
  
 //запрос на получение страницы со списком книг, у которых есть подстрока word  
 @GetMapping("/{word}")  
 public List<Book> selectedBooksPage(@PathVariable String word) {  
 return bookDAO.getBooksByWordMatching(word);  
 }  
}

BookDAO.java

//получить список всех книг  
public List<Book> index() {  
 return jdbcTemplate.query("SELECT \* FROM Books", new BookRowMapper());  
}  
  
//получить список книг, в которых встречается слово word (без учета регистра)  
public List<Book> getBooksByWordMatching(String word) {  
 return jdbcTemplate.query("SELECT \* FROM Books WHERE lower(name) LIKE lower(?);",  
 new BookRowMapper(),  
 "%"+word+"%");  
}

All-books.html

<!DOCTYPE html>  
<html xmlns:th="http://www.thymeleaf.org" lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>Book list</title>  
 <link rel="stylesheet" type="text/css" href="../assets/css/style.css">  
 <script src="../assets/js/search-books.js" defer></script>  
</head>  
<body>  
<h2>Страница со списком всех книг</h2>  
<form>  
 <input type="text" placeholder="Поиск" id="book-search" autofocus>  
</form>  
<div id="books-container">  
 <div th:each="book : ${books}">  
 <a th:href="@{/books/{id}(id=${book.getId()})}">  
 <span th:text="${book.getName()}"></span>,  
 <span th:text="${book.getAuthor()}"></span>,  
 <span th:text="${book.getYear()}"></span>  
 </a>  
 </div>  
</div>  
  
<hr>  
<a href="/books/new">Добавить</a>  
  
</body>  
</html>

Результат:













