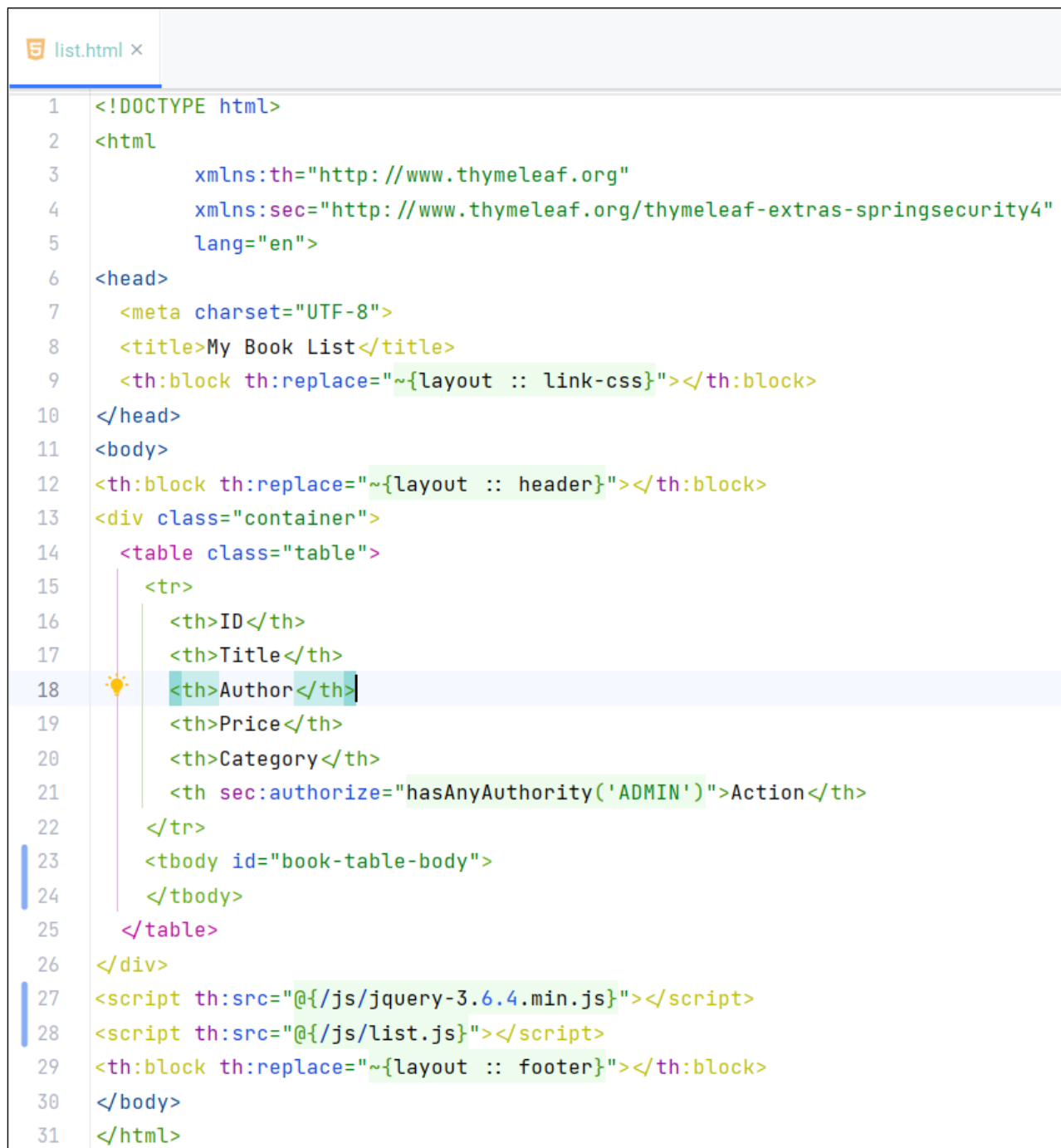


## 8.4 Chỉnh sửa List.html cho phù hợp

Chỉnh sửa giao diện **List.html** cho phù hợp với tính năng xây dựng API.



```
1 <!DOCTYPE html>
2 <html
3     xmlns:th="http://www.thymeleaf.org"
4     xmlns:sec="http://www.thymeleaf.org/thymeleaf-extras-springsecurity4"
5     lang="en">
6 <head>
7     <meta charset="UTF-8">
8     <title>My Book List</title>
9     <th:block th:replace="~{layout :: link-css}"></th:block>
10 </head>
11 <body>
12 <th:block th:replace="~{layout :: header}"></th:block>
13 <div class="container">
14     <table class="table">
15         <tr>
16             <th>ID</th>
17             <th>Title</th>
18             <th>Author</th>
19             <th>Price</th>
20             <th>Category</th>
21             <th sec:authorize="hasAnyAuthority('ADMIN')">Action</th>
22         </tr>
23         <tbody id="book-table-body">
24         </tbody>
25     </table>
26 </div>
27 <script th:src="@{/js/jquery-3.6.4.min.js}"></script>
28 <script th:src="@{/js/list.js}"></script>
29 <th:block th:replace="~{layout :: footer}"></th:block>
30 </body>
31 </html>
```

Hình 8.6. Tùy chỉnh lại giao diện list.html cho phù hợp

```

1 $(document).ready(function () : void { Lấy dữ liệu từ API
2     $.ajax({ url: {
3         url: 'http://localhost:8080/api/v1/books',
4         type: 'GET',
5         dataType: 'json',
6         success: function (data) : void {
7             let trHTML : string = '';
8             $.each(data, callback: function (i, item) : void {
9                 trHTML = trHTML + '<tr id="book-' + item.id + '>' +
10                     '<td>' + item.id + '</td>' +
11                     '<td>' + item.title + '</td>' +
12                     '<td>' + item.author + '</td>' +
13                     '<td>' + item.price + '</td>' +
14                     '<td>' + item.categoryName + '</td>' +
15                     '<td sec:authorize="hasAnyAuthority(\"ADMIN\")">' +
16                     '<a href="# + item.id + '" class="text-primary">Edit</a> | ' +
17                     '<a href="/books" + item.id + ' class="text-danger" onclick="apiDeleteBook(' + item.id + ', this); return false;" sec:authorize="hasAnyAuthority' +
18                     '(\"ADMIN\")">Delete</a>' +
19                     '</td>' +
20                     '</tr>';
21             });
22             $('#book-table-body').append(trHTML);
23         });
24     });
25
26 Xoá bằng API
27 function apiDeleteBook(id) : void {
28     if (confirm('Are you sure you want to delete this book?')) {
29         $.ajax({ url: {
30             url: 'http://localhost:8080/api/v1/books/' + id,
31             type: 'DELETE',
32             success: function () : void {
33                 alert('Book deleted successfully!');
34                 $('#book-' + id).remove();
35             }
36         });
37     }
38 }

```

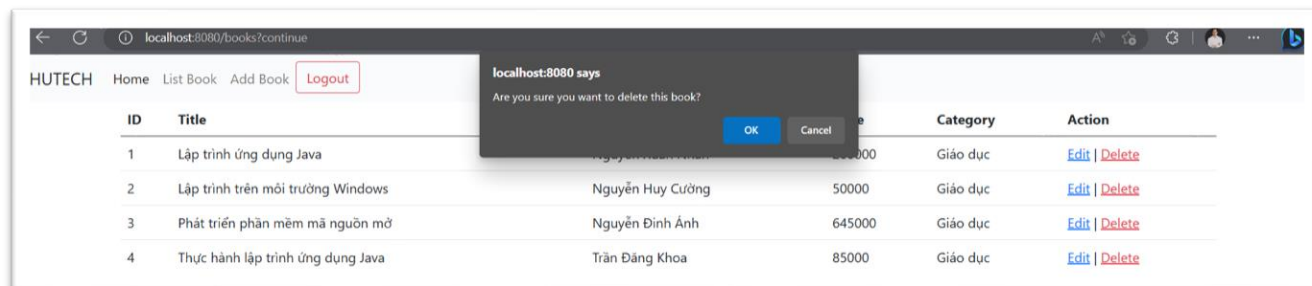
Hình 8.7. Các hàm lấy danh sách books và Delete book

Giao diện lấy danh sách Books sử dụng dữ liệu Json từ API khi đăng nhập bằng người dùng có quyền là USER

HUTECH	Home	List Book	Add Book	Logout
ID	Title	Author	Price	Category
1	Lập trình ứng dụng Java	Nguyễn Xuân Nhân	200000	Giáo dục
2	Lập trình trên môi trường Windows	Nguyễn Huy Cường	50000	Giáo dục
3	Phát triển phần mềm mã nguồn mở	Nguyễn Đình Ánh	645000	Giáo dục
4	Thực hành lập trình ứng dụng Java	Trần Đăng Khoa	85000	Giáo dục

Hình 8.8. Giao diện hiển thị danh sách Books bằng API

Giao diện Delete Book sử dụng API khi đăng nhập bằng người dùng có quyền là ADMIN



Hình 8.9. Giao diện xoá sách bằng API

## 8.5 Thực hiện

Chức