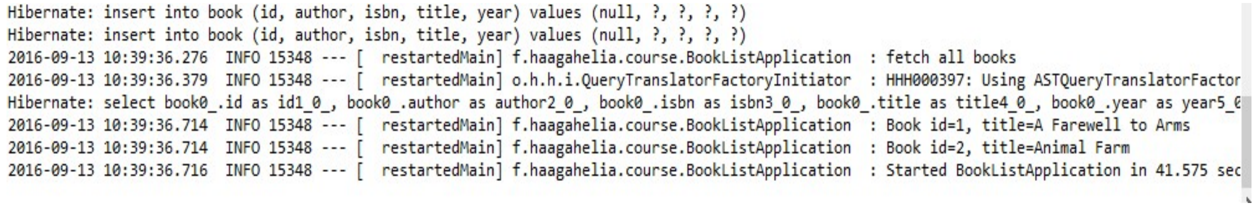
Chapter 3

Use your own bookstore application from the GitHub.

1. Bookstore: Add JPA and database

a.) Add JPA support and H2 database to your bookstore project by modifying pom.xml file.
b.) Create an entity from your Book class
c.) Check again that you DO NOT have attribute year – it is a reserved word – make it for example publicationYear
d.) Add command line runner to insert few example data to your database
e.) Modify properties to show SQL statements in the console
f.) Enable H2 console

You should see similar messages in the console after you run your application

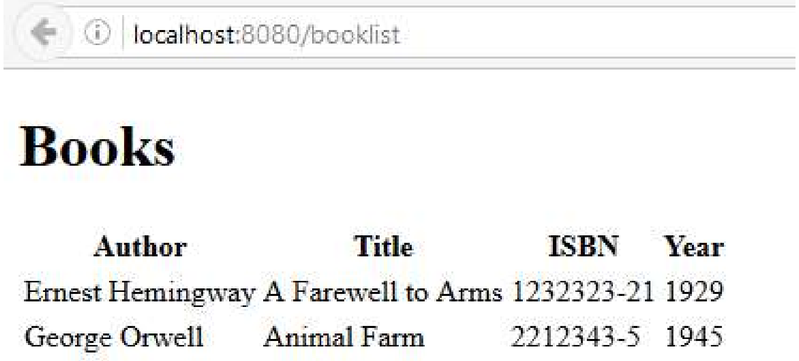


```
Hibernate: insert into book (id, author, isbn, title, year) values (null, ?, ?, ?, ?)
Hibernate: insert into book (id, author, isbn, title, year) values (null, ?, ?, ?, ?)
2016-09-13 10:39:36.276  INFO 15348 --- [  restartedMain] f.haagahelia.course.BookListApplication  : fetch all books
2016-09-13 10:39:36.379  INFO 15348 --- [  restartedMain] o.h.h.i.QueryTranslatorFactoryInitiator  : HHH000397: Using ASTQueryTranslatorFactor
Hibernate: select book0_.id as id1_0_, book0_.author as author2_0_, book0_.isbn as isbn3_0_, book0_.title as title4_0_, book0_.year as year5_0
2016-09-13 10:39:36.714  INFO 15348 --- [  restartedMain] f.haagahelia.course.BookListApplication  : Book id=1, title=A Farewell to Arms
2016-09-13 10:39:36.714  INFO 15348 --- [  restartedMain] f.haagahelia.course.BookListApplication  : Book id=2, title=Animal Farm
2016-09-13 10:39:36.716  INFO 15348 --- [  restartedMain] f.haagahelia.course.BookListApplication  : Started BookListApplication in 41.575 sec
```

Push all changes to GitHub.

2. Bookstore: Listpage

Create a template called booklist.html which shows all books from the database in html table. To show the books you have to add new method to controller that handle GET request from endpoint like /booklist.



Books

Author	Title	ISBN	Year
Ernest Hemingway	A Farewell to Arms	1232323-21	1929
George Orwell	Animal Farm	2212343-5	1945

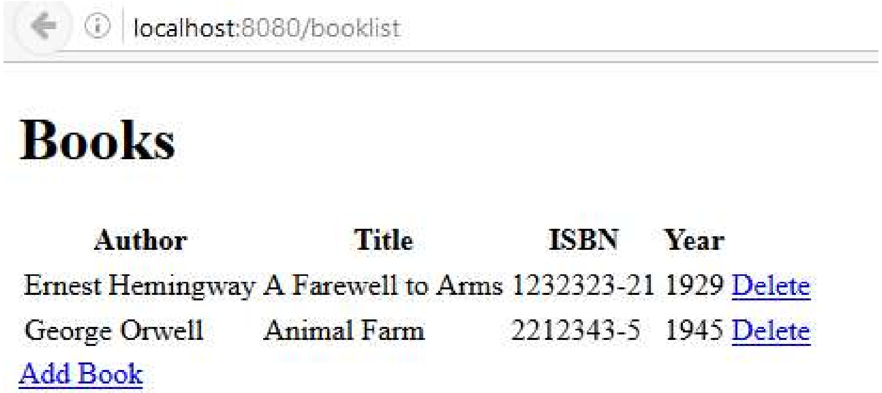
Push all changes to GitHub.

3. Bookstore: Crud

Add Create and Delete functionalities to your Bookstore application

For create functionality add new thymeleaf template called addbook.html

For deletion use *@PathVariable* annotation.



localhost:8080/booklist

Books

Author	Title	ISBN	Year	
Ernest Hemingway	A Farewell to Arms	1232323-21	1929	Delete
George Orwell	Animal Farm	2212343-5	1945	Delete

Add Book

Push all changes to GitHub.

4. Edit Bookstore

Add edit functionality to your bookstore. Create edit link after delete link to your listpage. Edit link will open current book in the edit page.

Tip. Check ‘Add book’-functionality from the controller. In add functionality you added new book object to model but now you will add current book object to model. You also have to send current book id from the list page to controller (like you did in delete link).

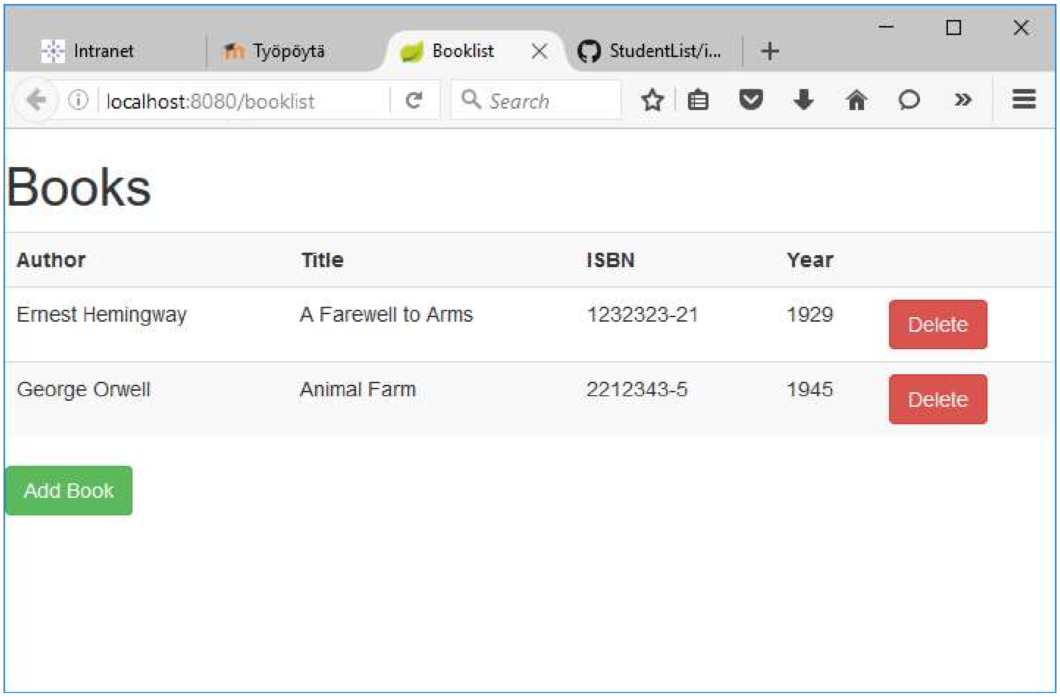
Push all changes to GitHub.

5. Bookstore: A Little bit Front-end (Extra)

1.) Add Bootstrap to your bookstore application. Download *bootstrap.min.css* from the bootstrap site. Copy it to your projects `resources\static\css` folder (Create the folder if it does not exist). Add link to your booklist template file.

```
<link type="text/css" rel="stylesheet" href="/css/bootstrap.min.css" th:href="@{/css/bootstrap.min.css}" />
```

2.) Add ***class="table table-striped"*** to your table html element
3.) Add ***class="btn btn-success"*** to your ‘Add Book’- button html element
4.) Add ***class="btn btn-danger"*** to your ‘Delete’- button html element



Books

Author	Title	ISBN	Year	
Ernest Hemingway	A Farewell to Arms	1232323-21	1929	Delete
George Orwell	Animal Farm	2212343-5	1945	Delete

Push all changes to GitHub.