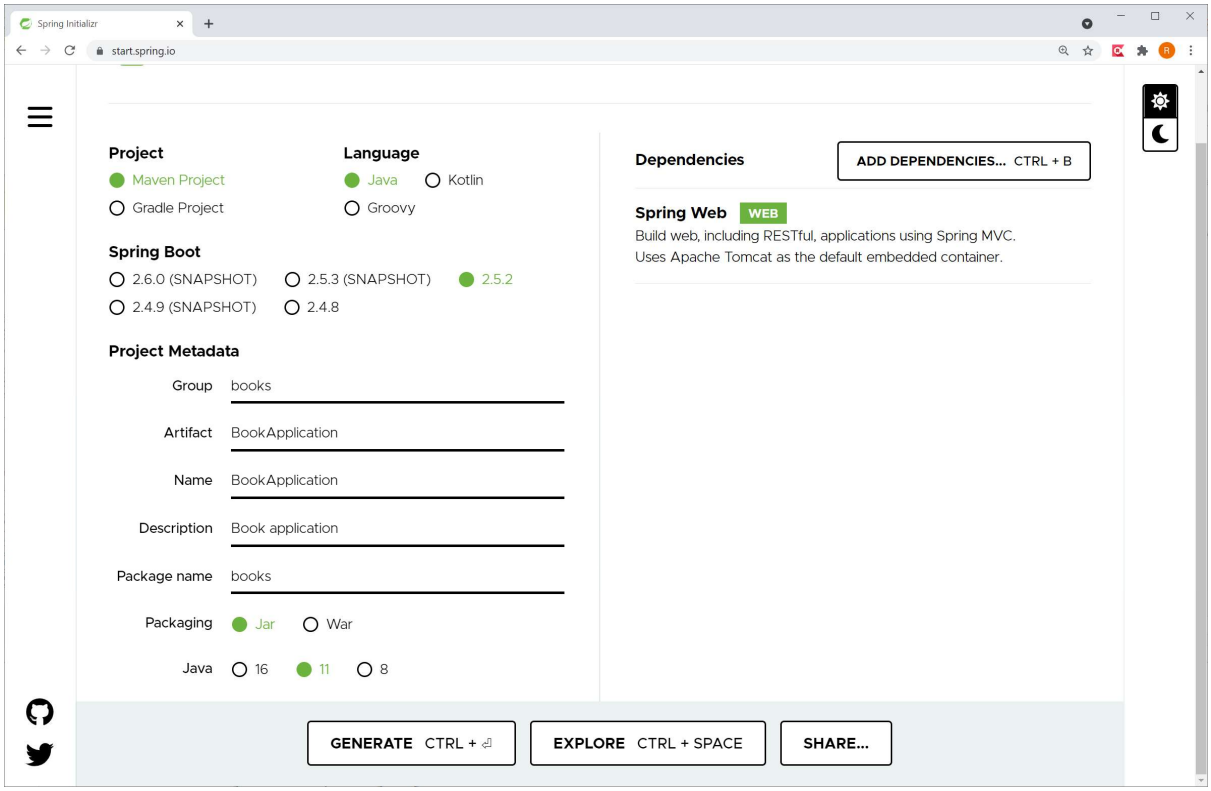
Lab 2

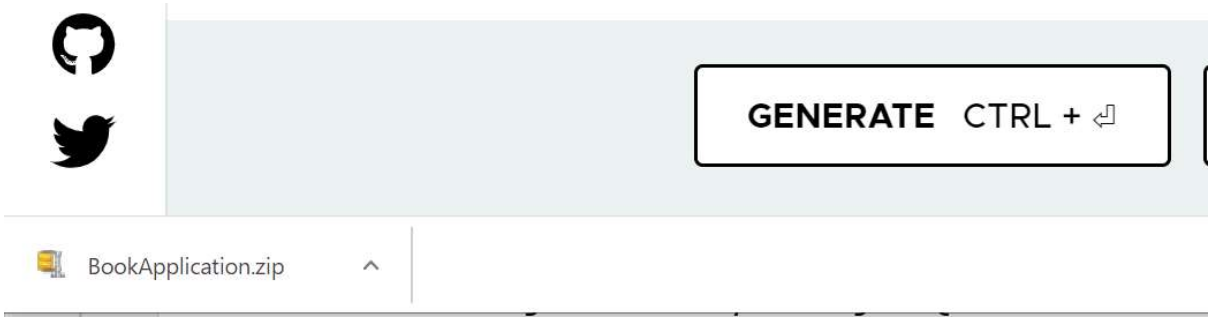
Part 1

In the browser go to https://start.spring.io/



Fill in the fields as shown above.

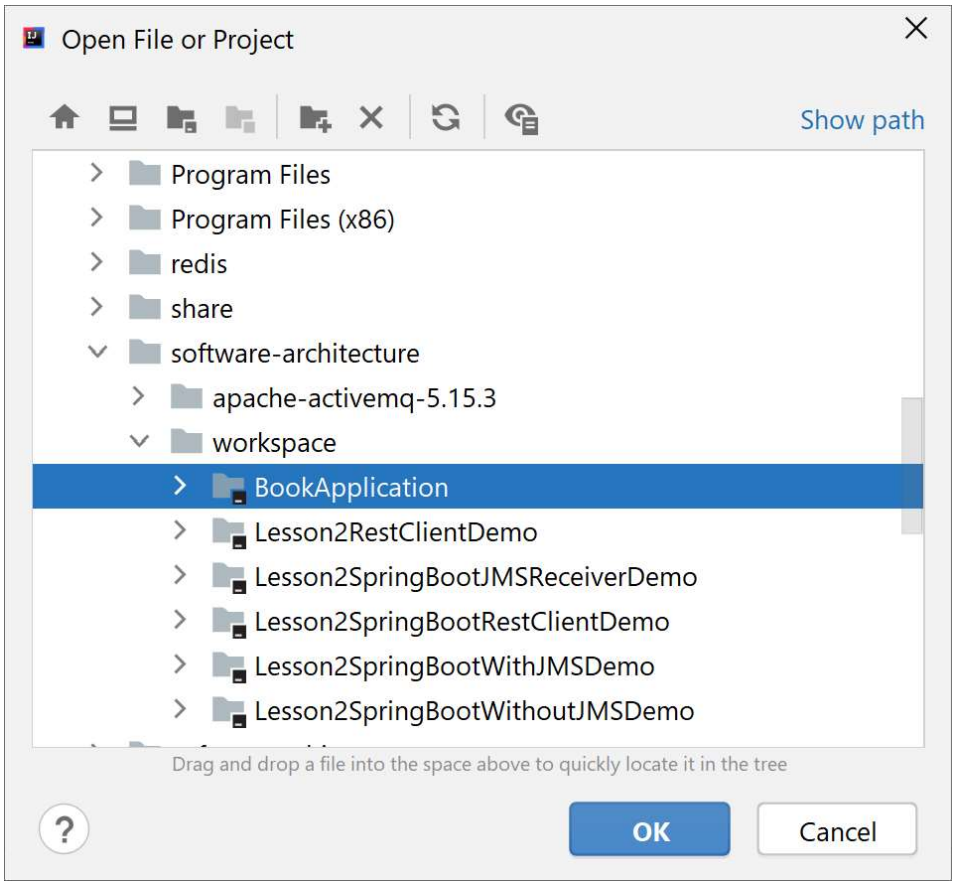
To get the web dependency, click the **Add Dependencies** button and select **Spring Web**.



Click the **Generate** button and see that **BookApplication.zip** is now downloaded to your computer.

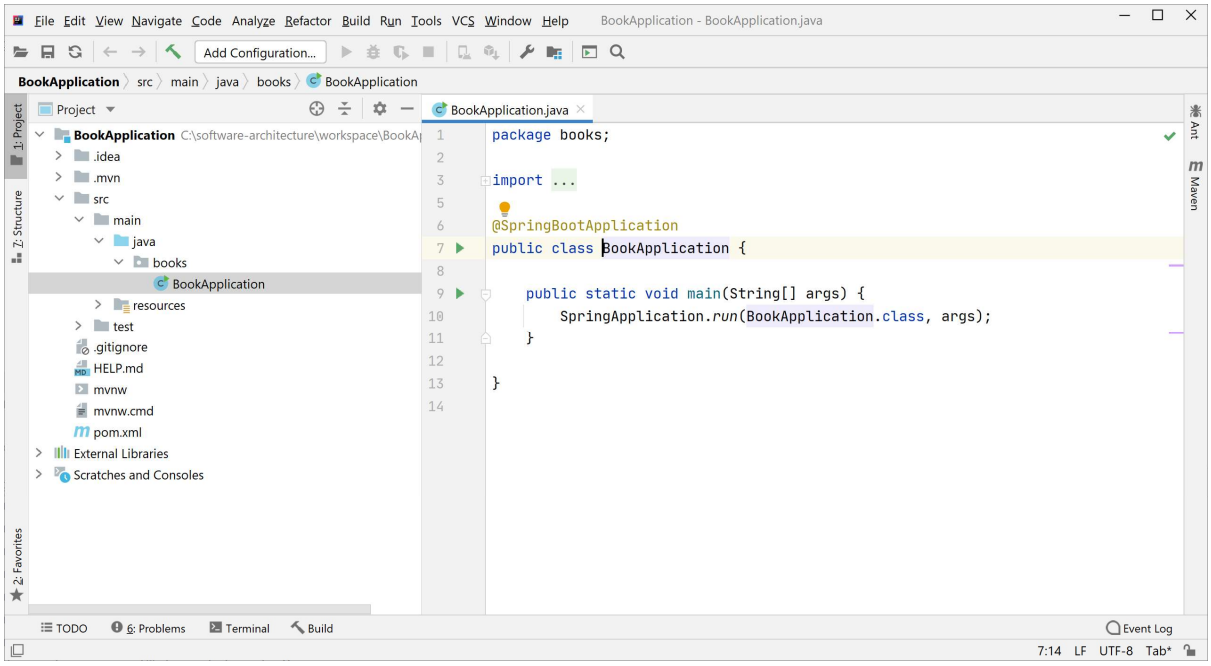
Unzip the content of this file, for example to c:\software-architecture\workspace

In IntelliJ, open now the folder **BookApplication** from the location where you unzipped the file.



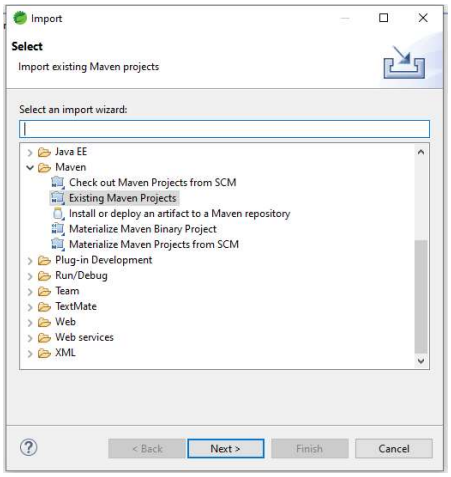
Click **OK**.

You see now a basic Spring boot application.

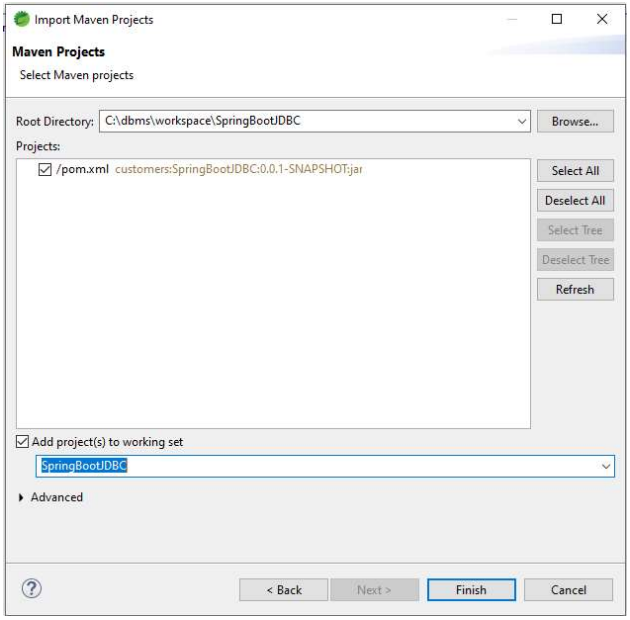


If you want to use Eclipse instead of IntelliJ, do the following:

In Eclipse select **File-> Import**



Choose **Maven-> Existing Maven Project** and click **Next**



Select the location of you project and check the **Add project to working set** checkbox.

Then click **Finish** and you are done.

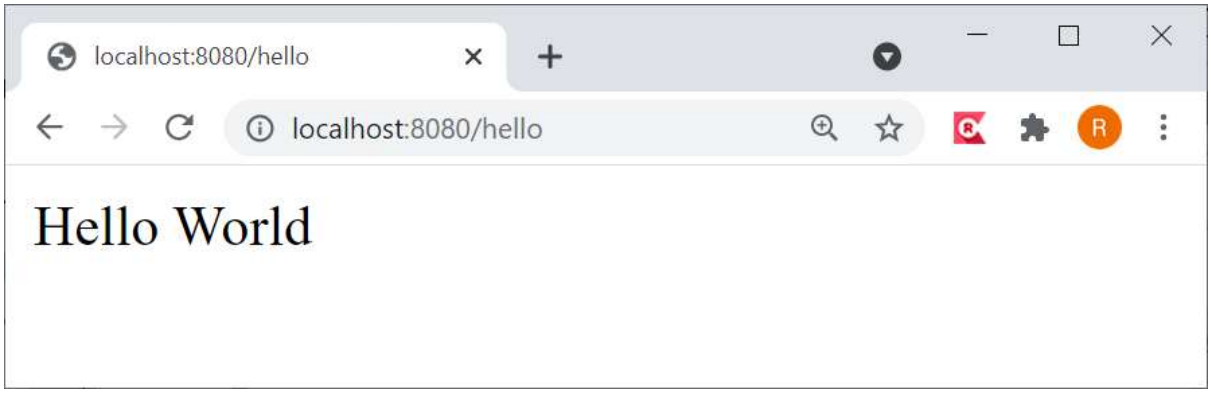
Now add the following class to the BookApplication project:

```
package books;

import org.springframework.web.bind.annotation.RestController;

@RestController
public class GreetingController {
    public String sayHello(){
        return "Hello World";
    }
}
```

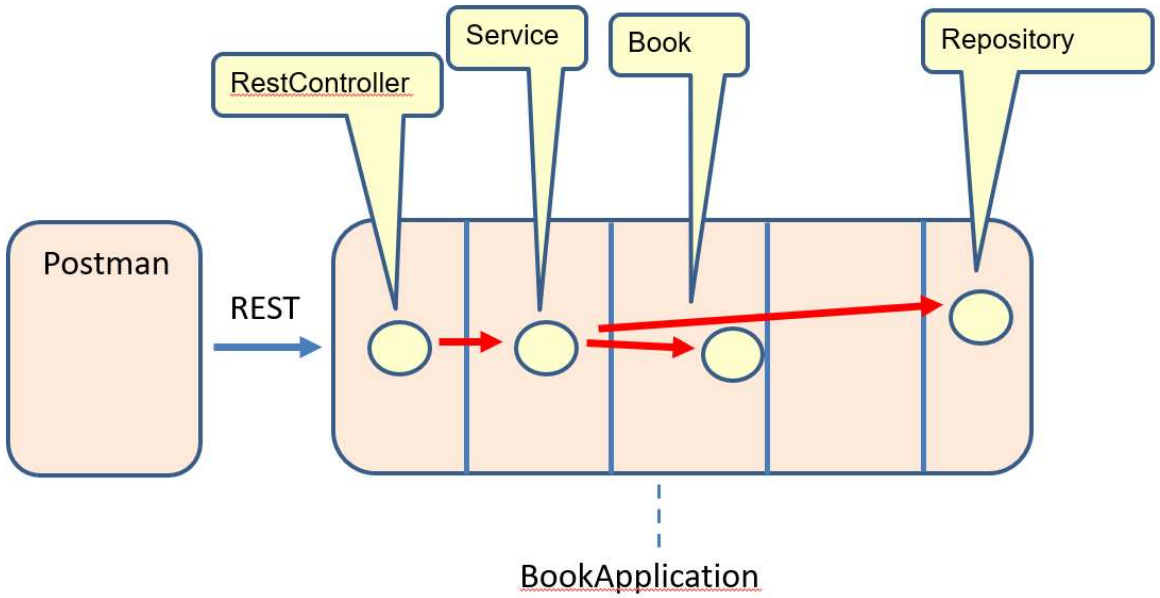
Then run the file BookApplication.java
Op the browser with the URL: **http://localhost:8080/hello**



Now write a Book application using REST with the following functionality:

addBook(Book book);
updateBook(Book book);
deleteBook(String isbn);
getBook(String isbn);
getAllBooks();

The Book class has the following properties: isbn, author, title, price
The application should have a controller class, a service class and a repository class.

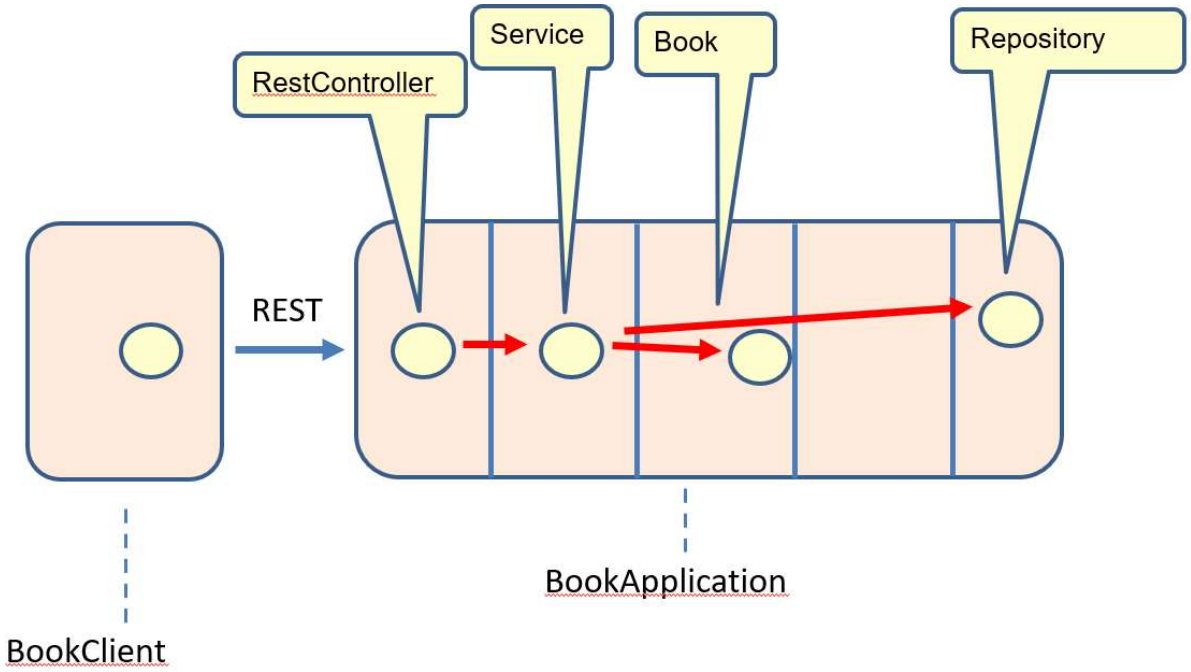


Download and install **postman** and check if your application works correctly

Part 2

Copy and paste the given **Lesson2SpringBootRestClientDemo** to a new project with the name BookClient.

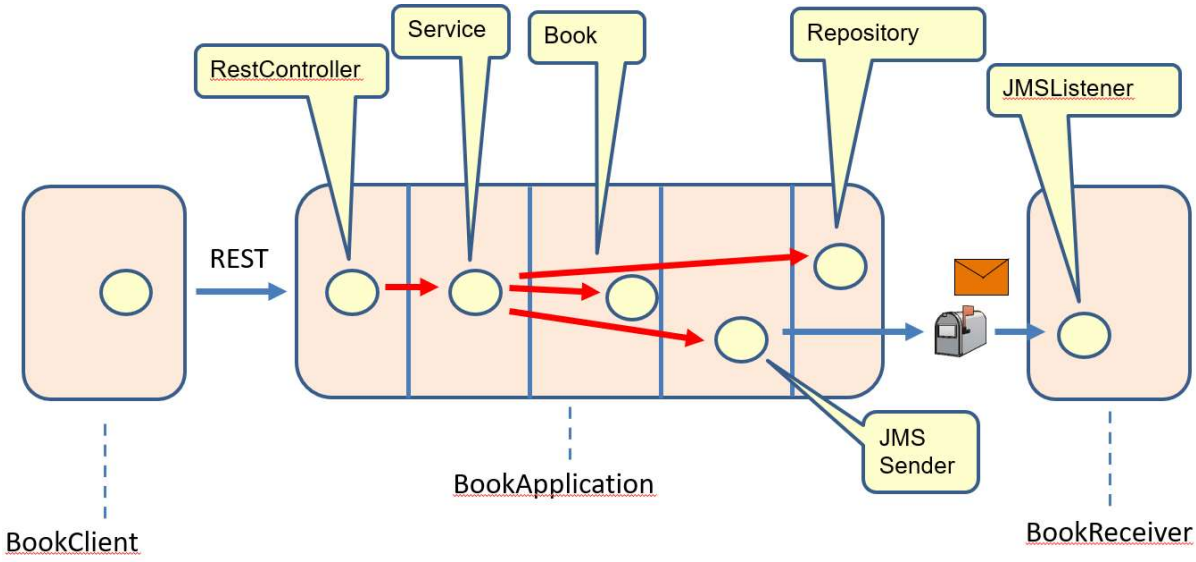
Modify the application so that this client application calls the REST interface of the BookApplication using a RestTemplate.



Part 3

Modify the BookApplication so that every time a book is added, deleted or updated, the application sends a JMS message with the corresponding book.

Copy and paste the given **Lesson2SpringBootJMSReceiverDemo** to a new BookReceiver project. Modify this project so that it receives all messages send by the BookApplication.



For this part you need to install ActiveMQ.

Windows users can just download it from

https://drive.google.com/file/d/1zXvVxWHXP4e6diCWREkqwhvUzqjzG3yG/view?usp=sharing

Then go to the folder **\apache-activemq-5.15.3\bin** and click the file **startactivemq.bat**

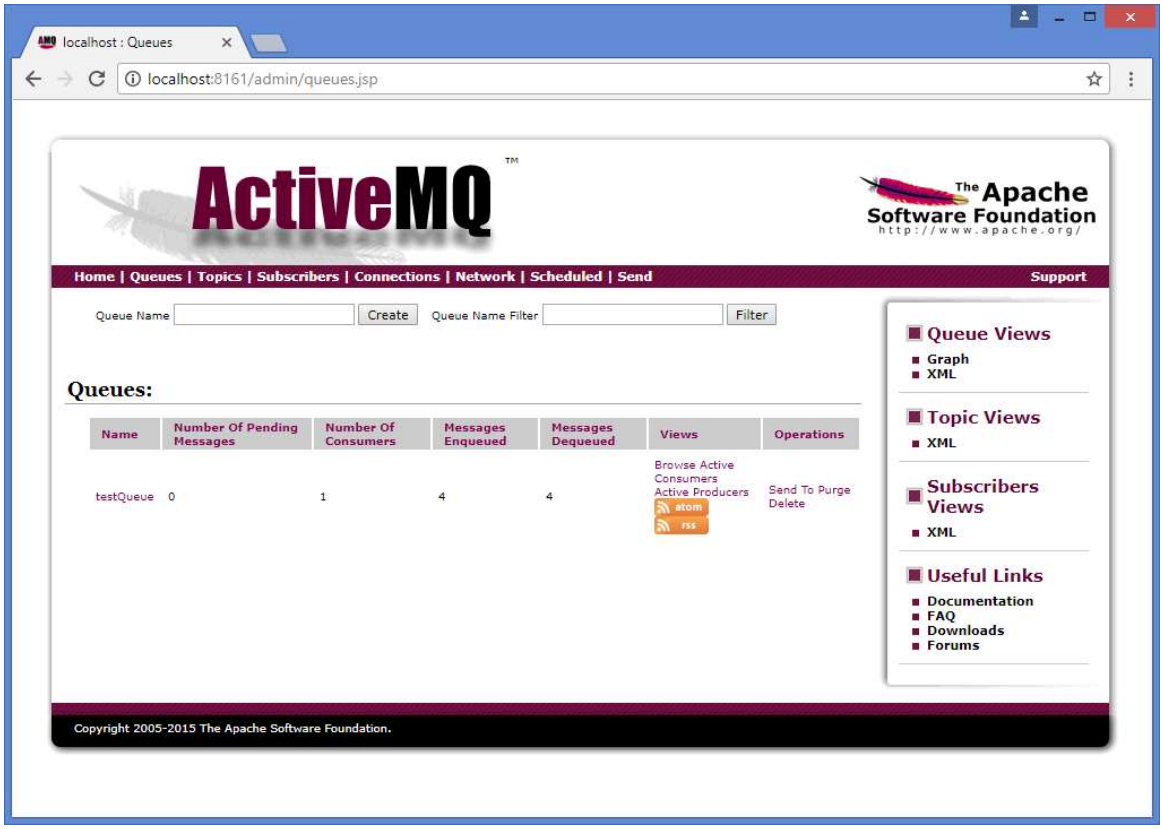
Mac users can follow the explanation in this link:

https://www.programmersought.com/article/60036401499/

Once ActiveMQ is running you can open the ActiveMQ console at http://localhost:8161/admin.

You can login with username **admin** and password **admin**

Select the Queues page from the menu:



Here you see the queues and other data.

What to hand in?

1. A separate zip file for part1, part 2 and part3
2. Write a readme.txt file with the following statement and sign with your name:

   I hereby declare that this submission is my own original work and to the best of my knowledge it contains no materials previously published or written by another person. I am aware that submitting solutions that are not my own work will result in an NC of the course.

   [your name as signature]