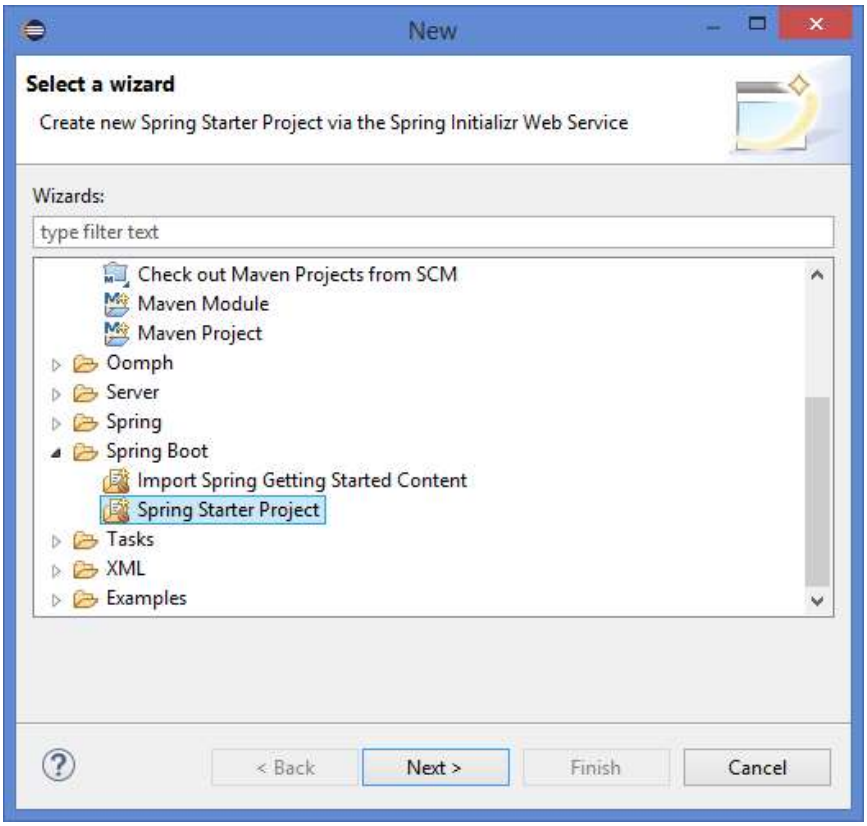
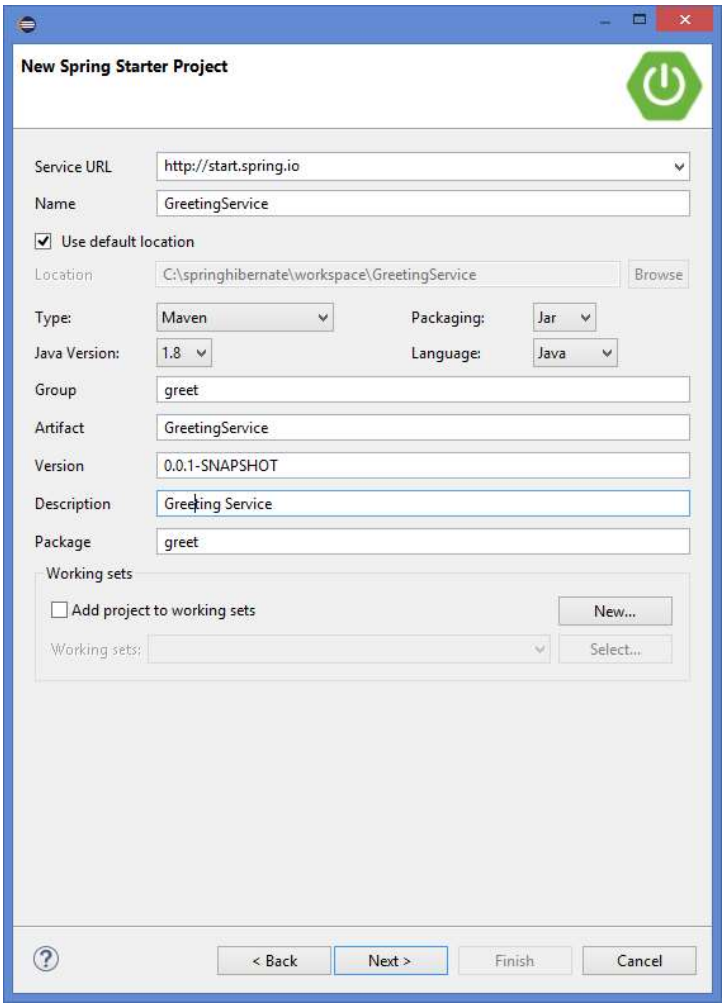
Lab assignment 2: Application architecture

Exercise 1: Spring Boot REST

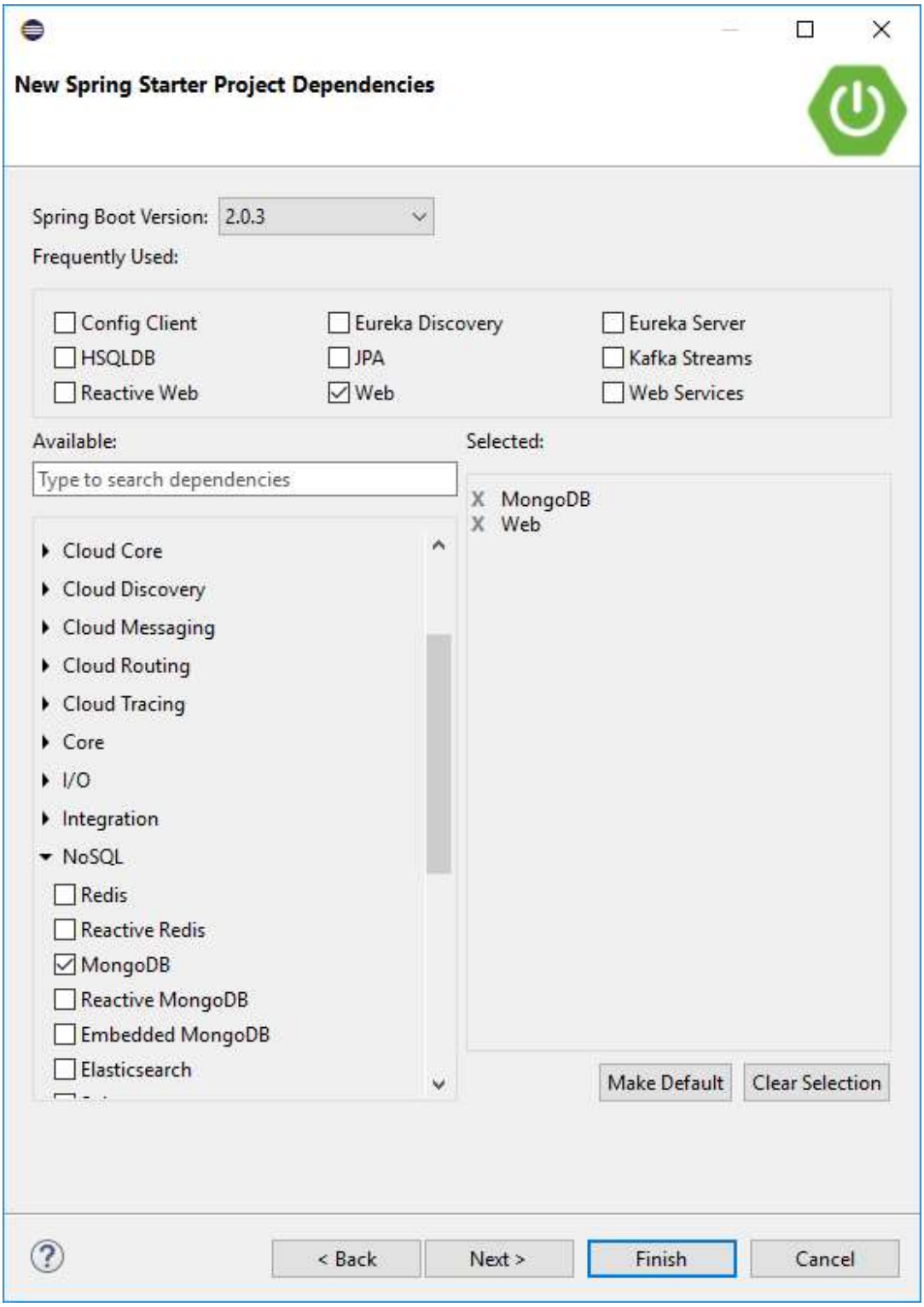
In Eclipse select the menu items **File->New->Other** and select **Spring Boot->Spring Starter Project**.



Click **Next**.



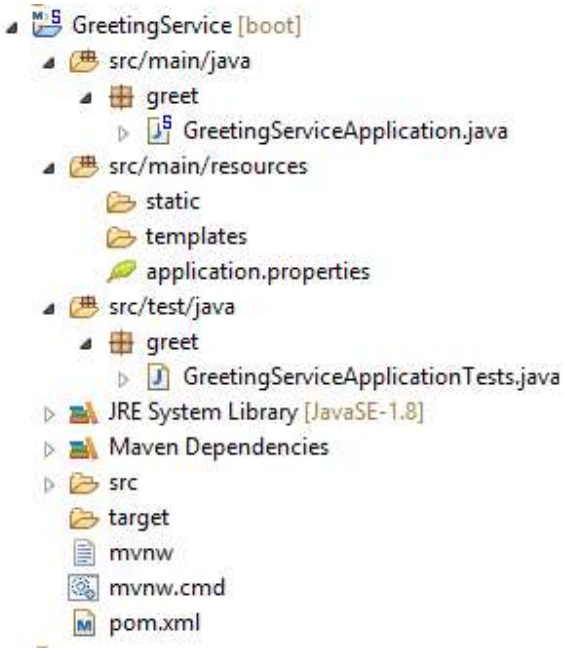
Fill in the details as given in the picture above and click **Next**.



Select the checkbox for **Web** and the checkbox for **Mongo**.
Also select Spring Boot version **2.0.3**.

Then click **Finish**.
Wait till the **GreetingService** project is created.

The following GreetingService project is created:



In the created GreetingService project, write the following classses:

```
@RestController
public class GreetingController {

    @RequestMapping("/greeting/{message}")
    public ResponseEntity<?> getGreeting(@PathVariable("message") String message) {
        Greeting greeting = new Greeting("");
        greeting.setContent("Hello World "+message);
        return new ResponseEntity<Greeting>(greeting, HttpStatus.OK);
    }
}
```

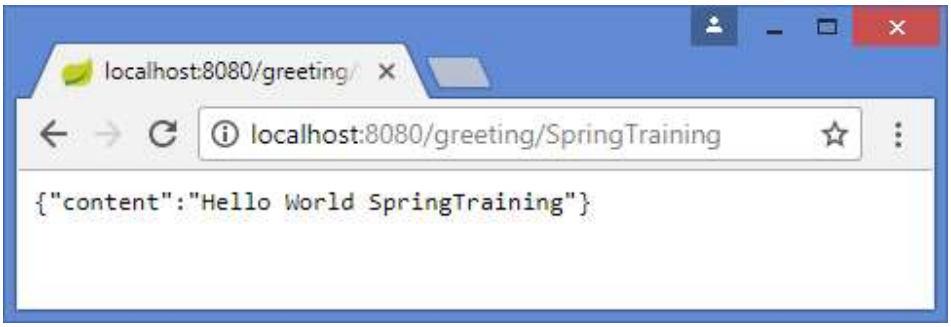
```java
public class Greeting {
    private String content;

    public Greeting(String content) {
        this.content = content;
    }

    public String getContent() {
        return content;
    }

    public void setContent(String content) {
        this.content = content;
    }

}
```

```java
@SpringBootApplication
public class HelloServiceApplication {

    public static void main(String[] args) {
        SpringApplication.run(HelloServiceApplication.class, args);
    }
}
```

Right-click the **HelloServiceApplication.java** file and select **Run as->Spring Boot App**.

You should see the following output:

```
  .   ____          _            __ _ _
 /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
 \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
  '  |____| .__|_| |_|_| |_\__, | / / / /
 =========|_|==============|___/=/_/_/_/
 :: Spring Boot ::             (v2.0.0.M4)

...
```



Open now the URL http://localhost:8080/greeting/SpringTraining in the browser and you should see the content given above.

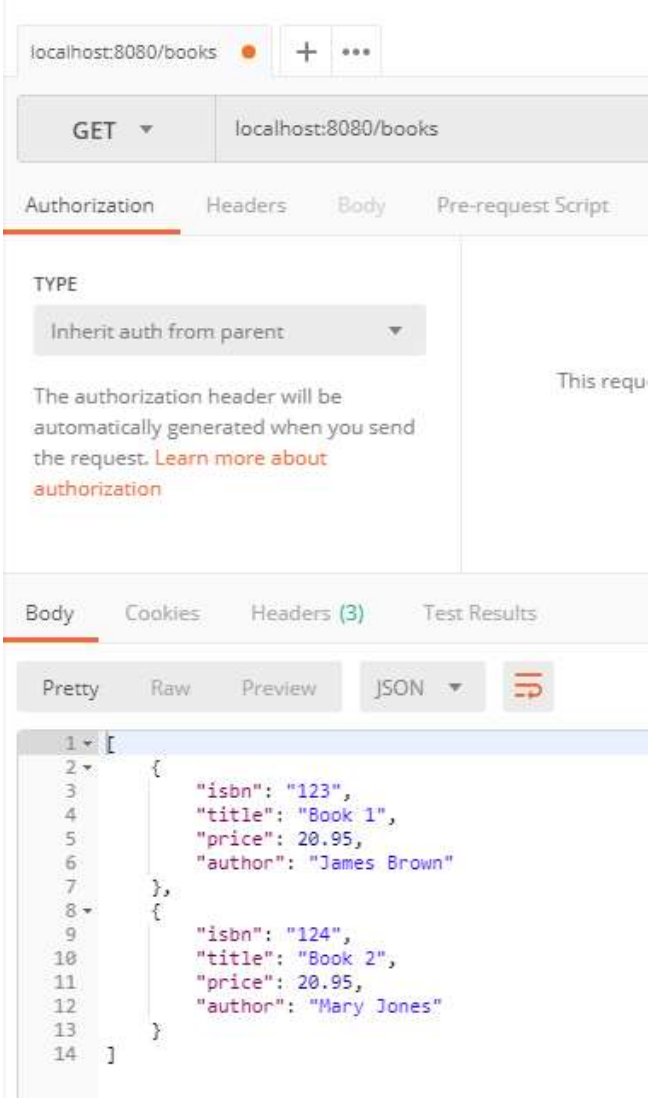
Now write a new BookService with the following functionality that we want to call with REST:

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

The Book class has the following properties: isbn, author, title, price

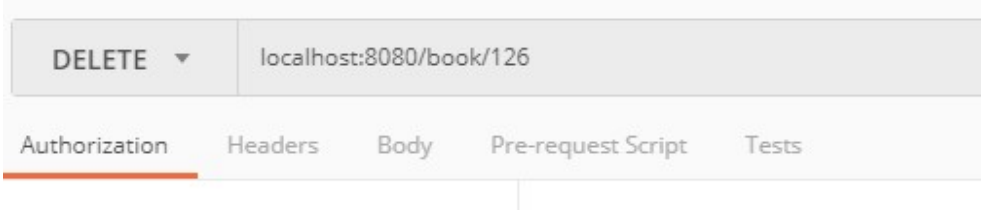
To test the BookService, you first have to install **Postman**. Download Postman from https://www.getpostman.com/apps and install Postman.
Test the BookService with Postman:

Get all the books



Posting a new book:

Deleting an existing book:



Exercise 2: Spring Mongo

Now we want to store the products of exercise 2 to the Mongo database.

First we have to start the mongo database by running
C:\architecturetraining\mongodb\bin\startmongo.bat.

Modify the application of Exercise 1 as such that the Books are stored in the Mongo database.

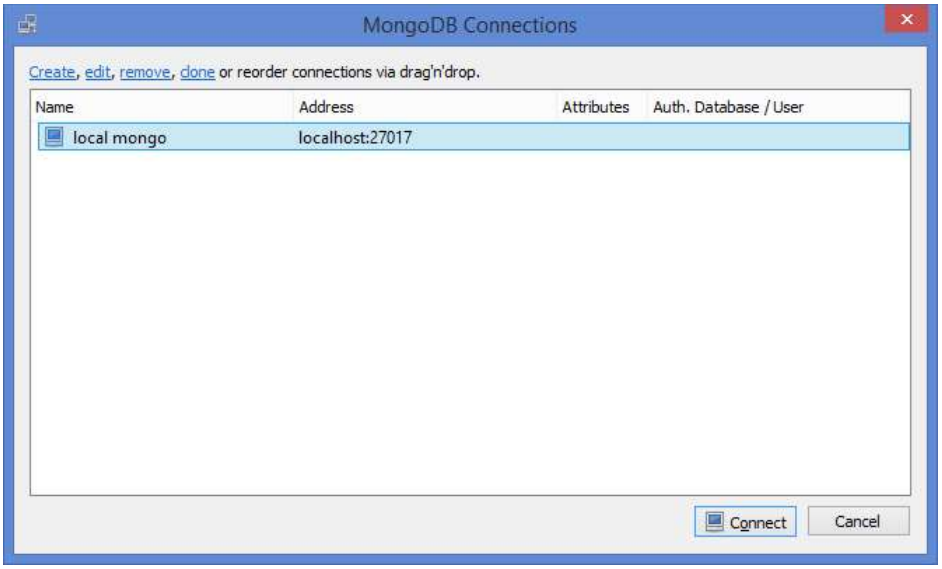
Create a BookRepository interface

Add the following properties to application.properties:

spring.data.mongodb.host=localhost
spring.data.mongodb.port=27017
spring.data.mongodb.database=testdb

Modify the BookService class so that the books are stored in the database.

Now run **C:\ architecturetraining\robo3t\ robo3t.exe.** This application is a Client application that allows you to look into the mongo database.



Click **Connect.**

In the **testdb** you should see now all the books.

Add some new books and check if they are stored in the database.

Exercise 3: Webshop implementation

Implement the ProductCatalogService from the webshop design of Lab 2 with Spring Boot. This ProductCatalogService has the following (subset) of methods:

public void addProduct(String productnumber, String description, double price)
public Product getProduct(String productnumber)
public void setStock(String productnumber, int quantity, String locationcode)

The products are stored in the mongo database.
Use REST to call the methods on the ProductCatalogService

In the same application, implement the ShoppingService from the webshop design of Lab 2 with Spring Boot with the following (subset) of methods:

public void addToCart(String cartId, String productnumber, int quantity)
public ShoppingCart getCart(String cartId)

The shoppingcart is stored in the mongo database.
Use REST to call the methods on the ShoppingService

IMPORTANT: You only learn from this lab if you do this lab yourself. If you copy a solution from someone else, then this is in violation with the academic honesty policy of the university and the penalty will be a NC for the course.

What to hand in?

You can submit a zip file with your solutions in sakai. All labs that you submit in sakai should contain a **readme.txt** file with the following statement:

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.

[your name as signature]

If you do not have this readme.txt file, you don't get credit for your lab submission.