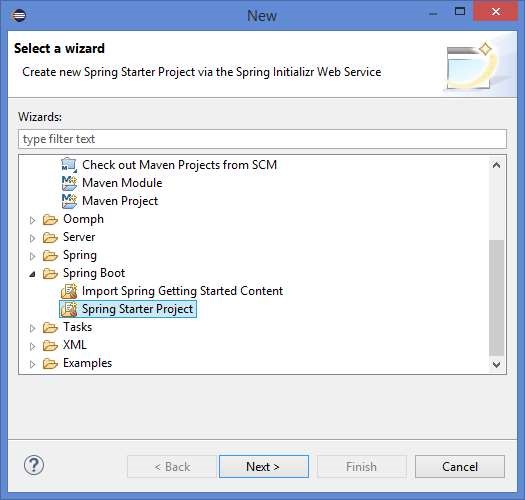
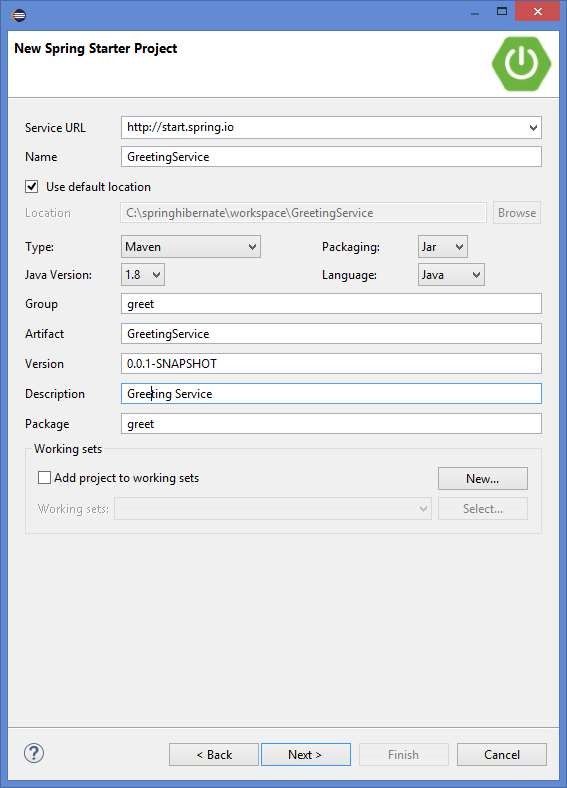
Lab assignment 2: Application architecture

# Exercise 1: Spring Boot REST

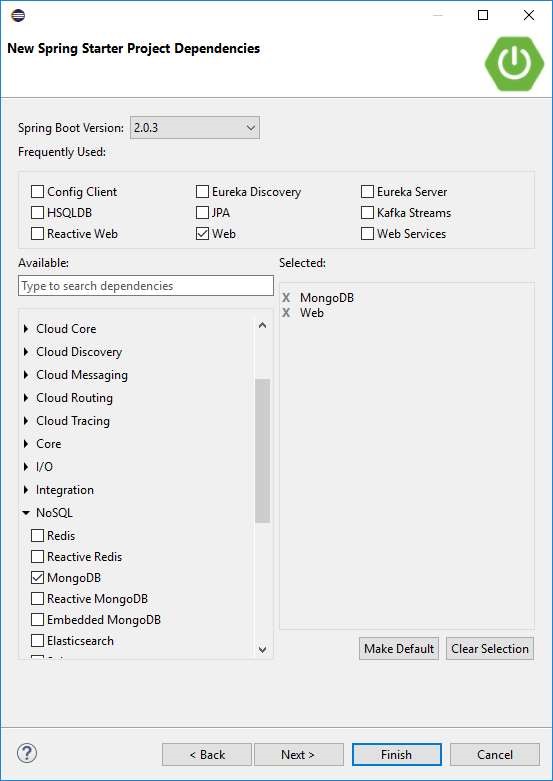
In Eclipse select the menu items File->New->Other and select Spring Boot->Spring Starter Project.



Click Next.



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



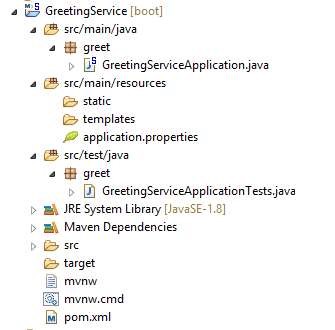
Select the checkbox for Web and the checkbox for Mongo.

Also select Spring Boot version 2.0.3.

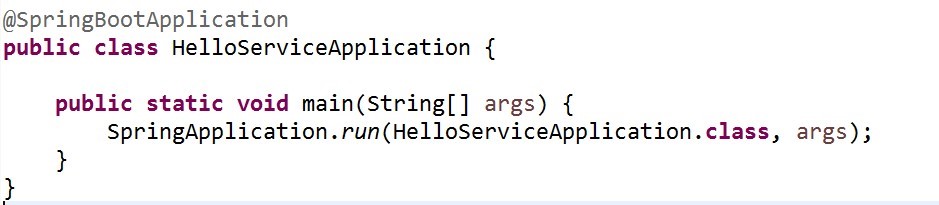
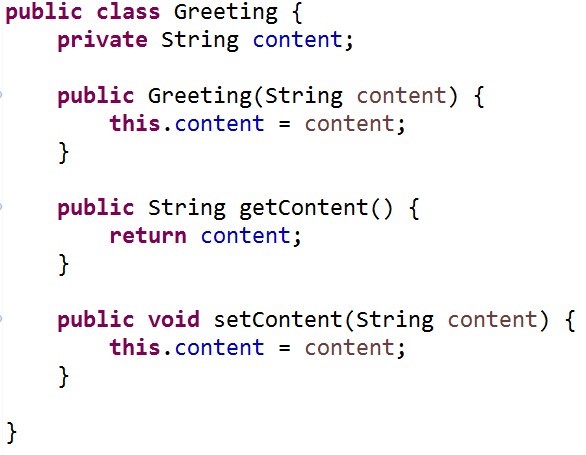
Then click Finish.

Wait till the GreetingService project is created.

The following GreetingService project is created:



In the created GreetingService project, write the following classses:



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

You should see the following output:

. \_\_\_\_ \_ \_\_ \_ \_

/\\ / \_\_\_'\_ \_\_ \_ \_(\_)\_ \_\_ \_\_ \_ \ \ \ \

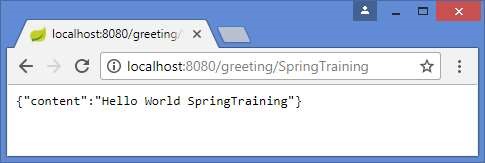
( ( )\\_\_\_ | '\_ | '\_| | '\_ \/ \_` | \ \ \ \

\\/ \_\_\_)| |\_)| | | | | || (\_| | ) ) ) )

' |\_\_\_\_| .\_\_|\_| |\_|\_| |\_\\_\_, | / / / /

=========|\_|==============|\_\_\_/=/\_/\_/\_/

:: 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.

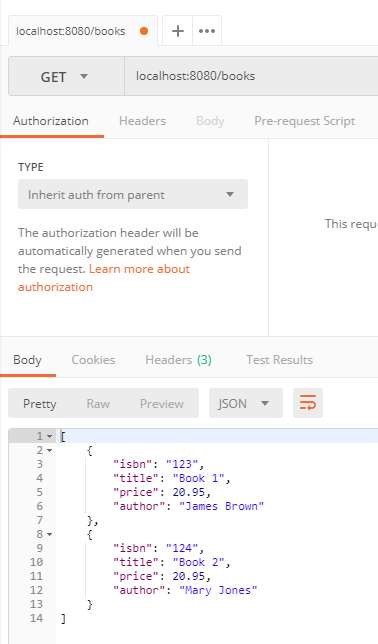
Now write a new BookService with the following functionality that we want to call with REST:

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

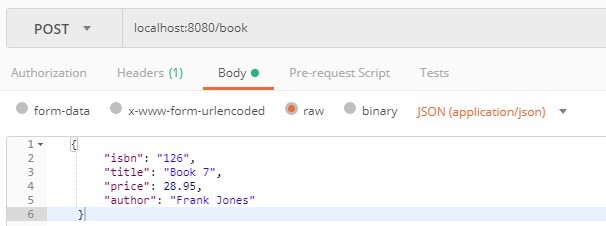
The Book class has the following properties: isbn, author, title, price

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

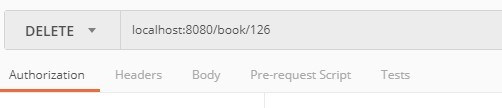
Get all the books



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.

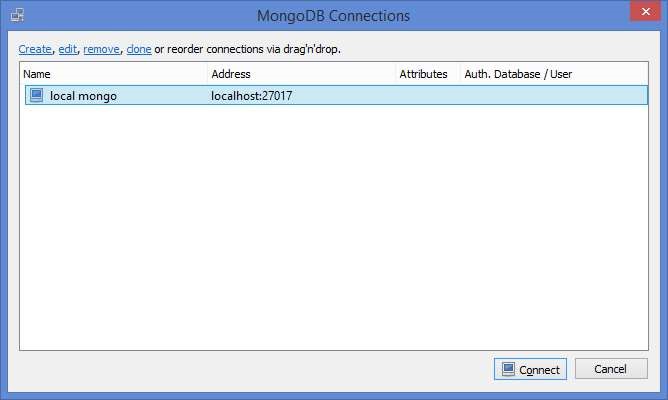
Create a BookRepository interface

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:

void addProduct(productnumber, description, price) Product getProduct(productnumber)

void setStock(quantity, 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:

void addToCart(productnumber, quantity)

ShoppingCart getCart()

The shoppingcart is stored in the mongo database.

Use REST to call the methods on the ShoppingService