Spring MVC Exercises and Thymeleaf – Week 1

Before we start

- Go to **Spring Initializr** and create your Maven Project
- Add the following Dependencies
- Web - Full-stack web development with Tomcat and Spring MVC
- Thymeleaf - Thymeleaf templating engine
- DevTools - Spring Boot Development Tools
- Validation – Hibernate validator
- Unpack it in your Workspace
- Load it into your IDE

Exercise 1

- Create a package name ”controllers” inside your base package
- Create a Class named ”Home” inside your controllers package
- Add ”@Controller” annotation on the class
- Create a method named ”index” that will return a String
- Add ”@GetMapping("/index")” annotation on the method
- Create ”index.html” at ”/src/main/resources/templates”
- If folder does not exist in your project, create it
- Add the necessary elements for a proper web-page
- Add the following attribute ”xmlns:th="http://www.thymeleaf.org” to the html element

Exercise 2

- In Home, Create a method named ”contact” that will return a String
- Add ”@GetMapping("/contact ")” annotation on the method
- Create ”contact.html” at ”/src/main/resources/templates”
- Add a Form that takes a text input • Method = Post
- In Home, Create a method named ”contact” that will take a String as input and return a String
- Add ”@PostMapping("/contact ")” annotation on the method
- Add ”@RequestParam” in front of the input String
- Save input String to a List of strings in the Home controller
- In Home, Create a method named ”contactList” that will take a Model as input named model
- Use ” addAttribute” on model and have your List of strings as secondary input
- Create View for this method and use Thymeleaf ”th:each” to show the List of strings content

Exercise 3

- In ”src/main/resources/static”
- Add the following folders css, js, img
- Add a ccs file and a img file in there respective folders
- In Home, Create a method named ”about” and a View for it
- Use the img file you added before in your View
- Use your css file in all your View’s

Exercise 4

- Create a Fever controller and methods with views to handle the following
- Ask the user for their body temperature
- Show the user if they have fever /normal / hyperthermia based on body temperature and treatment if needed
- Use Model
- You may not use JavaScript