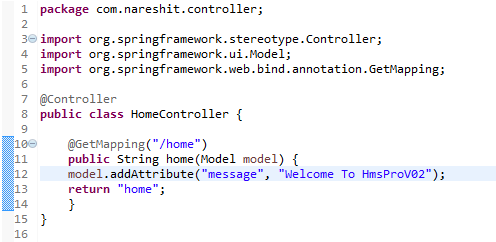
Thymeleaf is a server-side Java template engine that provides support for integrating with SpringMVC and SpringSecurity. Among the supporting view templates, Thymeleaf is the most popular one used in Spring Boot applications.

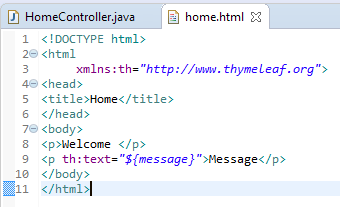
Create a Spring Boot Maven project with the spring-boot-starter-thymeleaf starter.

|  |
| --- |
| <dependency>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-starter-thymeleaf</artifactId>  </dependency> |

ThymeleafAutoConfiguration will take care of registering TemplateResolver, ThymeleafViewResolver, SpringResourceTemplateResolver, and SpringTemplateEngine. By default, **Spring Boot loads the Thymeleaf view templates from the classpath:/templates/ directory.** Now create a controller to handle the "/home" request.



This example is adding the string "Welcome To HmsProVo2" to model the key "message", which we want to render in the Thymeleaf template. Create the Thymeleaf view called home.html in the **src/main/resources/templates** directory.



This displays the model attribute "message" value in the Thymeleaf template using th:text="${message}". Now run the following entry point class.

**Working with Thymeleaf Forms**

Thymeleaf offers very good Spring integration with support for:

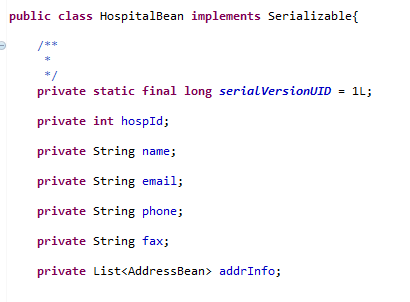
• Forms with backing beans and result/error bindings

• Use of property editors and conversion services

• Displaying internationalization (i18n) messages using ResourceBundles

• Using the Spring Expression Language (Spring EL)

Create a POJO called HospitalBean.java, as shown below



Create the addHospital.html file in the src/main/resources/templates directory, as shown below