Creating front end

1.Adding spring dependency to project prom file.

a.In the properties list , add:

<spring.version>4.3.6.RELEASE</spring.version>

b.In the dependencies , add:

<!-- Spring dependency -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-webmvc</artifactId>

<version>${spring.version}</version>

</dependency>

2.Configuring FrontController that will intercept all request .

a.Open the web.xml file of the project.

b.Inside the web-app tag , add the following :

<!-- Configuring front controller -->

<servlet>

<servlet-name>dispatcher</servlet-name>

<servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>

</servlet>

c.Also add the following mapping code beneath the servlet tag in web.xml:

<servlet-mapping>

<servlet-name>dispatcher</servlet-name>

<url-pattern>/</url-pattern>

</servlet-mapping>

3.Create a directory with the following structure :

--src

----main

------java

a.Inside the main folder , add a new folder with the name java.

4..Inside the package created under the java resources folder with the name src/main/java add a new package , give it the name net.sports.ZenSportsFrontEnd.controller.

5.Inside the controller folder , add a class , FrontEndPageController.java.

a.Add @Controller before the class declaration of the controller file and do Ctrl+Shft+O to resolve imports.

b.Inside the class , add the code :

@RequestMapping(value = { "/", "/home", "/index" })

public ModelAndView index() {

ModelAndView model = new ModelAndView("page");

model.addObject("greeting","Welcome to ZenSport.com");

return model;

}

Then do Ctrl+Shft+O to resolve imports.

6.Creating the page.jsp.

a.Inside the WEB-INF folder , create a new folder with the name views.

b.Inside this views folder , we have to add a jsp file with the name page.jsp.

c.Inside the body of the page.jsp , we have to add :

${greeting}

And also have to set false inside the isElIgnored attribute.And customise the title of the page.

d.Delete the index.jsp page.

7.Creating a spring bean configuration file.

a.Add an xml file inside the WEB-INF folder with the name dispatcher-servlet.xml.

b.Inside it , add the code :

<beans xmlns="http://www.springframework.org/schema/beans"

xmlns:context="http://www.springframework.org/schema/context"

xmlns:mvc="http://www.springframework.org/schema/mvc"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="

http://www.springframework.org/schema/beans

http://www.springframework.org/schema/beans/spring-beans.xsd

http://www.springframework.org/schema/context

http://www.springframework.org/schema/context/spring-context.xsd

http://www.springframework.org/schema/mvc

http://www.springframework.org/schema/mvc/spring-mvc.xsd

">

<context:component-scan base-package="net.sports.ZenSportsFrontEnd.controller" />

<!-- View Resolver -->

<bean id="viewResolver"

class="org.springframework.web.servlet.view.InternalResourceViewResolver">

<property name="prefix" value="/WEB-INF/views/" />

<property name="suffix" value=".jsp"/>

</bean>

</beans>

8.Add a server to the application.

a.Click on windows->add view->server or properties->project facets->runtime,then add Tomcatv8.0 server.

b.If it is not there available for adding , then click on add , and then browse to the installed directory for tomcat and then add it .

9.Start the server , and then type the following url in the browser :

<http://localhost:8080/ZenSportsFrontEnd/>

10.Adding dependency for JSTL in pom.xml.

a.Add the property in the property list in pom.xml:

<jstl.version>1.2</jstl.version>

b.Add the dependency in the dependencies tag:

<!-- JSTL dependency -->

<dependency>

<groupId>jstl</groupId>

<artifactId>jstl</artifactId>

<version>${jstl.version}</version>

</dependency>

11.Adding code in page.jsp.

a.In the page.jsp,add the core taglib after the page tag :

[%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%](mailto:%25@taglib%20prefix=%22c%22%20uri=%22http://java.sun.com/jsp/jstl/core%22%25)

b.Add the c:set after the taglib line:

<c:set var="contextRoot" value="${pageContext.request.contextPath}"/>

c.Inside the body of the page.jsp,add the following :

${contextRoot} says ${greeting}