1.Started with yesterday’s project .

2.Added four jsp and designed them using bootstram form and table components :

* Login.jsp
* Register.jsp
* Products.jsp
* ProductDetails.jsp

3.Linked all with the links on navbar .

4.Downloaded spring 3.6.2.RELEASE with URL :

<http://repo.spring.io/release/org/springframework/spring/>

5.Then we added spring mvc , by adding dependency in pom.xml ,as follows:

<!-- Property List above dependencies-->

<properties>

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

</properties>

<!-- Spring MVC Dependency inside dependencies-->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-webmvc</artifactId>

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

</dependency>

NOTE : if HTTPServlet error persists, then add following dependency too:

<!-- Dependency for javaee(for HTTPServlet error) -->

<dependency>

<groupId>javax</groupId>

<artifactId>javaee-api</artifactId>

<version>7.0</version>

<scope>provided</scope>

</dependency>

Creating Dispatcher Servlet

-----------------------------

6.Add dispatcher-servlet.xml inside WEB-INF folder.

7.Open web.xml file , and add the following code inside web-app tag:

<servlet>

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

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

</servlet>

<servlet-mapping>

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

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

</servlet-mapping>

8.In the dispatcher-servlet,add the following 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-3.0.xsd*

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

*http://www.springframework.org/schema/context/spring-context-3.0.xsd*

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

[*http://www.springframework.org/schema/mvc/spring-mvc-3.0.xsd*](http://www.springframework.org/schema/mvc/spring-mvc-3.0.xsd)*"*>

<!-- Load all components -->

<context:component-scan base-package=*"com.sports.yensports.controller"* />

<!-- Linking physical file to logical -->

<bean id=*"ViewResolver"*

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

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

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

</bean>

</beans>

Implementation of navigation between login,index,register,product with spring with url

9.Create folder “java” inside main.

10.Add a package inside the java folder named :

com.sports.yensports.controller

11.Add a java class as controller inside the package with the name YenSportsController , add the following code :

@Controller

**public** **class** YenSportsController

{

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

**public** ModelAndView index(){

ModelAndView model=**new** ModelAndView("index");

**return** model;

}

@RequestMapping(value={"/login"})

**public** ModelAndView login(){

ModelAndView model=**new** ModelAndView("login");

**return** model;

}

@RequestMapping(value={"/register"})

**public** ModelAndView register(){

ModelAndView model=**new** ModelAndView("Register");

**return** model;

}

@RequestMapping(value={"/product"})

**public** ModelAndView product(){

ModelAndView model=**new** ModelAndView("Products");

**return** model;

}

@RequestMapping(value={"/productDetails"})

**public** ModelAndView productDetails(){

ModelAndView model=**new** ModelAndView("ProductDetails");

**return** model;

}

}

NOTE : Do ctrl+Shft+o for import all the required packages.

12.Create a folder named views inside WEB-INF,and add all the .jsp files inside it.

Loding Static Resources

------------------------

13.Open dispatcher servlet , and add the following :

<mvc:annotation-driven />

<mvc:resources location=*"/assets/"* mapping=*"/resources/\*\*"* />

14.In all the .jsp pages , after the page tag , add the following :

<%@taglib prefix="s" uri="http://www.springframework.org/tags"%>

Note:add isELIgnored="false" inside the page directive

15.Inside the head of the .jsp pages ,add the following :

<s:url value=*"/resources/images"* var=*"images"* />

<s:url value=*"/resources/js"* var=*"js"* />

<s:url value=*"/resources/css"* var=*"css"* />

Note:These are references to ${images},${css} and ${js} in place of folder locations.

Eg(Head):

<link rel=*"stylesheet"* href=*"*${css}*/bootstrap.min.css"* />

<link rel=*"stylesheet"* href=*"*${css}*/style.css"* />

<script src=*"*${js}*/jquery.js"*></script>

<script src=*"*${js}*/bootstrap.min.js"*></script>

Eg(Body):

<img src="${images}/cycles.jpg" alt="Cycles">

Fetching context path to set navigations to all pages through links,spring

16.In place of referring to pages by their url , write as follows:

<a href=*"*${pageContext.request.contextPath}*/register"*><span

class=*"glyphicon glyphicon-plus"*></span> Register</a>

17.Follow step 14,15 and 16 for all the .jsp pages.