1. Open Existing HealthMedicineQuest Project in Eclipse.
2. Added 4 jsp pages named login.jsp, register.jsp, productDetail.jsp and product.jsp.
3. Designed login.jsp and register.jsp using bootstarp form component.
4. Designed productDetail.jsp using datatable component of bootstrap.
5. Linked all pages and mapped.
6. Downloaded spring 4.3.6 release from <http://repo.spring.io/release/org/springframework/spring/> url.
7. Add dependencies for implementing Spring in pom.xml:
   1. Open pom.xml and type the following code:

<!-- Property List -->

<properties>

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

</properties>

* 1. Added following code inside <dependencies> tag

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-webmvc</artifactId>

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

</dependency>

* 1. If HTTPServlet Error persists then add following dependency code:

<dependency>

<groupId>javax</groupId>

<artifactId>javaee-api</artifactId>

<version>7.0</version>

<scope>provided</scope>

</dependency>

1. Create the Dispatcher Servlet:
   1. Open web.xml file and type the following code:

<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>

* 1. Create dispatcher-servlet.xml file inside the WEB-INF folder and type 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"*

>

<!-- Load all components -->

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

<bean id=*"viewResolver"* class=*"org.springframework.web.servlet.view.InternalResourceViewResolver"*>

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

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

</bean>

</beans>

1. Implement Navigation between LandingPage, Login Page and Register Page using Spring.
   1. Add a folder inside main named “java”.
   2. Add a package inside the java folder named “com.health.HeathMedicineQuest.controller”.
   3. Add java class file as controller inside the above said package named “HealthController”.
   4. Type the following code to configure the controller:

**import** org.springframework.stereotype.Controller;

**import** org.springframework.web.bind.annotation.RequestMapping;

**import** org.springframework.web.servlet.ModelAndView;

@Controller

**public** **class** HealthController {

@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={"/allproducts"})

**public** ModelAndView allproducts(){

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

**return** model;

}

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

**public** ModelAndView product(){

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

**return** model;

}

}

NOTE: Press Ctrl+Shft+O to add required packages.

1. Create a folder inside WEB\_INF named “views” and shift all .jsp pages within this folder.
2. Fix the "image not coming after implementing Spring Controller" Issue (Loading Static Resources):
   1. Open dispatcher-servlet.xml file and add the following code:

<!-- Loading static resources -->

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

<mvc:annotation-driven/>

* 1. Open index.jsp page. Add isELIgnored="false" attribute to the page directive.
  2. Add taglib directive as following:

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

* 1. Replace the <head> tag with following code except <style and title tag>:

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

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

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

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

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

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

<!-- Latest compiled JavaScript -->

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

<!-- jQuery library -->

<script src=*"*${jquery}*/jquery-3.1.1.min.js"*></script>

<!-- Internal Style -->

* 1. All the <img> tag src value must preceded with ${images} and <a> tag href must be preceded with ${pageContext.request.contextPath}/
  2. Repeat the same for all the pages.