LoginController Routing :- searchable url

// Mapping 🡪 approviate view

Register

//Mapping 🡪 appropriate view

SignUp

Forgotpassword

Rememberme

SPRING MVC Design Pattern

1. We will be creating Spring Web Application
2. Developing the web application using Spring MVC Design pattern
   1. Model :- It represents the data . It is associated with the Business logic of the application for accessing and manipulating of the data.
   2. View :- UI
   3. Controller :- It is responsible for taking the user request and passes it to the model for appropriate action. Based on the result of the action, the controller directs the appropriate view to the user.
3. A Spring MVC is sub framework of Spring framework which is used to build a web application following MVC design pattern, that allows us to represent the UI flow of a Web application into individual controllers and views.
   1. Features
      1. Pluggable view technology :- Jsp, Jsf, Tiles, Velocity,Thymeleaf, FreeMaker.
      2. Injection of services into controllers .
      3. Integration support.
4. Why Spring MVC?
   1. Scalability
   2. It separates each role (Model, View, Controller
      1. Segregating business concerns.
   3. Loosely coupled application
   4. Rapid Application Development

Is DS or Spring is a magic?

==== Digram

1. DispatcherServlet -> FrontController
2. ViewResolver 🡪 Helps in Resolving the views.
   * 1. Prefix (name)
     2. Suffix (.jsp)

Maven Spring MVC

HTML

Servlet :- Server side application

Business logics

Is also for presentation (it is embedded with html)

MVC application Steps

1. Configure the dispatcher servlet in web.xml (FrontController)
2. Create Spring configuration file (dispatcher-servlet.xml)
3. Configure ViewResolver
4. Create Controller
5. Create a view to show (.jsp)
6. Create a maven project
7. Archetype-webapp
8. Groupid :- com.spring.mvc
9. Artifact :- springmvc

Things to Ensure

Steps and configuration for Implementing Spring MVC Web Application using ViewResolver

Create Maven – webapp project

1. Window 🡪 Preferences 🡪 Server 🡪 Runtime environment
2. Add Library of Tomcat server in Build Path of the project
3. Check the Project Facets 🡪 Runtime 🡪 Tomcat server must be enabled.

Configurations

For Web application

web.xml 🡪 webapp🡪WEB-INF

The front controller to understand the request and forwards it to the appropriate controller with the help of handler mappings, is called as dispatcher servlet in spring

<!DOCTYPE web-app PUBLIC

"-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"

"http://java.sun.com/dtd/web-app\_2\_3.dtd" >

<web-app>

<display-name>Archetype Created Web Application</display-name>

<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>

</web-app>

Configuartion of Spring

IOC :- Registering the components(controller)

The name of the xml file must be same as the servlet-name in web.xml file

dispatcher-servlet.xml

<?xml version = *"1.0"* encoding = *"UTF-8"*?>

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

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

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

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

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.xsd"*>

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

<mvc:annotation-driven/>

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

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

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

</bean>

</beans>

1. POM.xml

Spring-webmvc dependency

Business Logic

Package :- com.example.controller

@Controller

**public** **class** HomeController {

@RequestMapping("/home")

**public** String overview() {

System.***out***.println("inside home handler");

**return** "home";

}

@RequestMapping("/about")

**public** String about() {

System.***out***.println("inside about handler");

**return** "about";

}

}

Views

Src/main/webapp/WEB-INF/jsp

about.jsp home.jsp

Exception :- classnotfoundexception – dispatcher servlet –solution

