Form data binding to Model class object using spring MVC supplied jsp tags
==

=> The form designed using traditional html tags support only **one way data binding** i.e only form
submission data goes to Model class object , But Model class obj data does not go to form comps as
initial values of form comps on the launch of form page.

=> To solve this Problem,design form page by using spring mvc supplied jsp form tags which supports **two way**
binding form submission to Model class obj and Model class object data to form comps.
i.e

jsp tag library is a library where jsp tags are available

Spring MVC /Spring Boot MVC gives two jsp taglibaries

a) Generic Jsp Tag **Library**
    (For general purpose operations like reading values from properties file, reading scoped data and etc..)
b) **Form Jsp tag Library**
    {For Form designing with the support of two way binding}

note:: Both these jsp tag libraries are part of spring mvc /spring boot mvc module.

=>Every Jsp taglibrary is identified with its taglib url

Spring's Generic Taglibrary url is :: *http://www.springframework.org/tags*

To use the tags of any jsp tag library , we need to import the jsp taglibrary by specifying its taglibrary uri and its prefix <%@taglib uri=".......," prefix=".....,"/>

Spring's Generic Taglibrary tags are ::

sp here is the custom prefix this can be changed to any thing

Spring's Form taglibrary url is :: *http://www.springframework.org/tags/form*

spring's form taglibrary tags are ::

frm here is the custom prefix

=> To use the jsp tags of jsp tag libraries in the jsp pages we need to
import jsp tag library by specifying its taglib uri using
<%@taglib uri="......." prefix=".....,"/>

if the form page is designed by using spring supplied jsp form tags ... we can perform two way
binding operations
    a) Model class obj data to Form comps as initial values on the launch of form page
    b) Form comps values to Model class object values on the submission of form page

welcome.jsp

RequestMappingHandlerMapping (c1) (t1)
(e1) (v1)

DispatcherServlet (x1)

```
<h3 style="text-align:center>  (x1)
<a href="register">Student register</a></h1>
```

Student class obj @Controller
(stud) public class StudentController{
  sname:null
  sadd:hyd

student_register.jsp (n1)

```
<%@taglib uri="http://www.springframework.org/tags/frm" prefix="frm" %>
<h1 style="color:blue;text-align:center">Student Registration Form</h1>
<frm:form action="register" method="POST" modelAttribute="stud">
```
(r1)
(o1) //Model class obj
        to form comps data binding
```
<table border="0" bgcolor="cyan" align="center">
<tr>
<td>Student id:</td>
<td><frm:input path="sno"/> </td>    1001
</tr>                           null
<tr>
<td>Student name: </td>
<td><frm:input path="sname"/> </td>   raja
</tr>                           null
<tr>
<td>Student address: </td>
<td><frm:input path="sadd"/> </td>   delhi
</tr>                     hyd
<tr>
<td>Student avg:: </td>
<td><frm:input path="avg"/> </td>
</tr>                    0.0    90.00
<tr>
<td><input type="submit" value="register"></td>
<td><input type="reset" value="cancel"> </td>
</tr>

</table>

</frm:form>
```

InternalResourceViewResolver
|-->prefix:/WEB-INF/pages/
|-->suffix: .jsp

/WEB-INF/pages/student_register.jsp

/WEB-INF/pages/show_result.jsp

(p1) fill the form comps

```
@GetMapping("/register")  //to launch form page
public String showForm(@ModelAttribute("stud") Student xt) {
    System.out.println("StudentOperationsController.showForm()");
    //System.out.println("Model class object data ::"+st);
    //return LVN
    return "student_register";  (l1)
}

@PostMapping("/register")  //for form submission  (u1?)
public String processStudent(Map<String,Object> map,
                 @ModelAttribute("stud") Student xt) {
    System.out.println("StudentOperationsController.processStudent()");
    System.out.println("Model class object data ::"+st);
    // generate result
    String grade=null;
    if(st.getAvg()>=75)
        grade="First Class with Dist";
    else if(st.getAvg()>=60)
        grade="First class";
    else if(st.getAvg()>=50)
        grade="Scond class";
    else if(st.getAvg()>=35)
        grade="Third class";
    else
        grade="Fail";
    //keep results in Model attribute
    map.put("result",grade);  (z1)
    //return LVN
    return "show_result";  (a2)
}
}
```

show_result.jsp (f2)

```
<%@ page isELIgnored="false" %>

<h1>Student Grade :: ${result}</h1> <br>

<a href="./">home</a>
```
(g2)

Student.java
=========== @Data
public class Student{
    private Integer sno;
    private String sname;
    private String sadd="hyd";
    private float avg;
}

o1 :: performs Model class obj to form comps data binding (Performing two way binding here)
x1 :: performs Form comps to Model class obj data binding

note:: if we do not give name to model attribute in @ModelAttribute(-) annotation w.r.t Model class obj them the model class name having first letter in lowercase takes as the model attribute name

==>In the form page ,that is designed by using spring mvc supplied jsp tags
    a) the default request mode /method is POST
    b) Giving model attribute is mandatory
    c) if the request path of form launching handler method is matching with
       Form submission handler method then placing action url in the
       "action" attribute of <frm:form> tag is optional becoz the form launching
       uri itself will be used as the form submission url   (action url)

In <frm:form> tag the attribute "modelAttribute" is the mandatory attribute and the attributes "action", method are the optional attributes having default values

```
<%@taglib uri="http://www.springframework.org/tags/form" prefix="frm" %>
<h1 style="color:blue;text-align:center">Student Registration Form </h1>
<frm:form modelAttribute="stud">
```
                no action, method attributes   equal to

```
<form id="cust" action="/BootMVCProj06-FormDataBinding-
HtmlTags/register" method="post">
```

```
<table border="0" bgcolor="cyan" align="center">
<tr>
<td>Student id:</td>
<td><frm:input path="sno"/> </td>
</tr>
<tr>
<td>Student name: </td>
<td><frm:input path="sname"/> </td>
</tr>
<tr>
<td>Student address: </td>
<td><frm:input path="sadd"/> </td>
</tr>
<tr>
<td>Student avg:: </td>
<td><frm:input path="avg"/> </td>
</tr>
<tr>
<td><input type="submit" value="register"></td>
<td><input type="reset" value="cancel"> </td>
</tr>

</table>

</frm:form>
```

note:: if we take <a>tag with out href attribute and <form> tag with out "action" attribute the request url using which the web page is launched automatically becomes href url in <a> tag and action url in the <form> tag   (this is applicable both in traditional html tags and in MVC supplied jsp tag)

traditional

between
what is the difference desing form pages using html form tags and spring mvc supplied jsp form tags?

Form data binding to Model class object using spring MVC supplied jsp tags

====

=> The form designed using traditional html tags support only one way data binding i.e only form submission data goes to Model class object, But Model class obj data does not go to form comps as initial values of form comps on the launch of form page.

=> To solve this Problem design form page by using spring mvc supplied jsp form tags which supports two way binding form submission to Model class obj and Model class object data to form comps.

1.e

Spring MVC/Spring Boot MVC gives two jsp taglibaries

a) Generic Jsp Tag Library

jsp tag library is a library where jsp tags are available

(For general purpose operations like reading values from properties file, reading scoped data and etc..) b) Form Jsp tag Library

(For Form designing with the support of two way binding)

note:: Both these jsp tag libraries are part of spring mvc/Spring boot mvc module.

=>Every Jsp taglibrary is identified with its taglib uri

Spring's Generic taglibrary uri is :: http://www.springframework.org/tags

Spring's Generic Taglibrary tags are ::

Spring's Form taglibrary uri is ::

spring's form taglibrary tags are ::

→sp:argument

→ sp:bind

→ sp:escapeBody

→ sp:eval

sp:hasBindErrors →sp:htmlEscape

→ sp:message

→ sp:nestedPath

→ sp:param

→ sp:theme

→ sp:transform

→ sp:url

To use the tags of any jsp tag library, we need to import the jsp taglibrary by specifying its taglibrary uri and its prefix

<%@taglib uri="......" prefix="....."/>

sp here is the custom prfefix

this can be changed to any thing

http://www.springframework.org/tags/form

→ frm:button

→ frm:checkbox

◆ frm:checkboxes

frm:errors

→ frm:form

frm:hidden

→ frm:input

♦ frm:label

→ frm:option

→ frm:options

→ frm:password

frm here is the custom prefix

=> To use the jsp tags of jsp tag libraries in the jsp pages we need to

import jsp tag library by specifying its taglib uri using

<%@taglib uri="......." prefix="...."/>

frm:radiobutton

frm:radiobuttons frm:select frm:textarea

mvc

if the form page is designed by using spring supplied jsp form tags.. we can perform two way binding operations

welcome.jsp

a) Model class obj data to Form comps as initial values on the launch of form page

b) Form comps values to Model class object values on the submission of form page

(c1) (t1) RequestMappingHandlerMapping]

`(el) (v1)`

DispatcherServlet

(x1)

student_register.jsp

(n1)

<h3 style="text-align:center"> (a1)

Student register</h1>

BindingAware Mode Map obj result: first

class with

(b1) (f1)

(j1) (m1) (s1) (w1)

Student class obj no:n (stud) sname:ull sadd:hy

dist (x1

Student obj Ng:90/stud) sho:1001 shame raja sadd delhi

(b2)

ag:0 of

 (e2)

(g1)

@Controller

public class StudentController{

@GetMapping("/register") //To launch form page

(d1?)

public String showForm(@ModelAttribute("stud") Student st) {

System.out.println("StudentOperationsController.showForm()"); System.out.println("Model class object data ::"+st); //return LVN

return "student_register"; (i1)

@PostMapping("/register") //for form submission

(u1?)

@ModelAttribute("stud") Student st) {

public String processStudent(Map<String, Object> map,

(y1)

System.out.println("StudentOperationsController.processStudent()");

System.out.println("Model class object data ::"+st);

<%@taglib uri="http://www.springframework.org/tags/form" prefix="frm" %> <h1 style="color:blue;text-align:center">Student Registration Form </h1> <frm:form action="register" method="POST" modelAttribute="stud">

(r1)

<table border="0" bgcolor="cyan" align="center">

<tr>

<td>Student id::</td>

<td><frm:input path="sno"/> </td> 1001

</tr>

<tr>

null

<td>Student name::</td>

<td><frm:input path="sname"/> </td>

</tr>

<tr>

null

raja

}

(01) (Model class obj

to form comps data binding) (k1) (c2)

InternalResourceViewResolver

|-->suffix: .jsp

(11) /WEB-INF/pages/student_register.jsp

|-->prefix:/WEB-INF/pages/

<td>Student address::</td>

<td><frm:input path="sadd"/> </td>

</tr>

<tr>

hyd delhi (p1) fill the form comps

<td>Student avg:: </td>

<td><frm:input path="avg"/></td>

</tr>

<tr>

0.0

90.00

<td><input type="submit" value="register"></td> <td><input type="reset" value="cancel"> </td>

</tr>

</table>

</frm:form>

show_result.jsp

(f2)

<%@ page isELIgnored="false" %>

**<h1> Student Grade :: ${result}</h1>
**

home

(g2)

01:: performs Model class obj to form comps data binding x1 :: performs Form comps to Model class obj data binding

// generate result

(d2) /WEB-INF/pages/show_result.jsp

String grade=null;

if(st.getAvg()>=75)

grade="First Class with Dist";

else if(st.getAvg()>=60) grade="First class";

else if(st.getAvg()>=50) grade="Scond class"; else if(st.getAvg()>=35) grade="Third class";

else

grade="Fail";

//keep results in Model attribute

map.put("result",grade); (z1)

//return LVN

```
return "show_result"; (a2)
}
```

Student.java

===========

```
@Data
public class Student{
}
```

(Performing two way binding here)

```
private Integer sno;
private String sname;
private String sadd="hyd";
private float avg;
```

note:: if we do not give name to model attribute in @ModelAttribute(-) annotation w.r.t Model class obj them themel class name having first letter in lowercase

takes as the model attribute name

=>In the form page, that is designed by using spring mvc supplied jsp tags

a) the default request mode /method is POST

b) Giving model attribute is mandatory

c) if the request path of form launching handler method is matching with Form submission handler method then placing action url in the "action" attribute of <frm:form> tag is optional becoz the form launching url itself will be used as the form submission url (action url)

```
<%@taglib uri="http://www.springframework.org/tags/form" prefix="frm" %> <h1
style="color:blue;text-align:center">Student Registration Form </h1> <frm:form modelAttribute="stud">
```

```
<table border="0" bgcolor="cyan" align="center">
<tr>
<td>Student id::</td>
<td><frm:input path="sno"/> </td>
</tr>
<tr>
<td>Student name::</td>
<td><frm:input path="sname"/> </td>
</tr>
<tr>
<td>Student address::</td>
<td><frm:input path="sadd"/> </td>
```

</tr>

<tr>

<td>Student avg::</td>

<td><frm:input path="avg"/> </td>

</tr>

<tr>

<td><input type="submit" value="register"></td> <td><input type="reset" value="cancel"> </td>

In <frm:form> tag the attribute "modelAttribute" is the mandatory attribute and the attributes "action", method are the optional attributes having default values

equal to

no action, method attributes

<form id="cust" action="/BootMVCProj06-FormDataBinding- HtmlTags/register" method="post">

note:: if we take <a>tag with out href attribute and <form> tag with out "action" attribute the request url using which the web page

is launched automatically becomes href url in <a> tag and action url in the <form> tag

(this is applicable both in traditional html tags and in MVC supplied jsp tags)

</tr>

</table>

</frm:form>

✓MS MVCBootProj05-FormDataBindingUsing Mvc Tags [boot]

>

Deployment Descriptor: MVCBootProj05-FormDataBindin

>Spring Elements

> JAX-WS Web Services

src/main/java

✓com.nt

> MvcBootProj5FormDataBindingApplication.java

>ServletInitializer.java

com.nt.controller

> StudentOperationsController.java

com.nt.model

> Student.java

#src/main/resources

>

static

templates

application.properties

src/test/java

> JRE System Library [JavaSE-16]

>

Maven Dependencies

> C Deployed Resources

>classcontent

✓ **src**

main

> java

>

resources

✓ webapp

✓

WEB-INF

✓ `pages`

show_result.jsp student_register.jsp welcome.jsp

between

traditional

what is the differnce desing form pages using html form tags and spring mvc supplied jsp form tags?

Trandtional html form tags

(a) These tags are given by w3c

(world wide web consortium)

(b) These tags support one way data binding

when they are used Spring mvc apps (form data to Model class obj)

(c) default request method for the form page

is GET

are

(d) These tags not strictly typed

are

(e) These tags not case-sensitive

(f) can be used in html programming and

also in jsp programming (.html, .jsp files)

(g) tags are given for submit button, reset button

(h) supports the writing of place holder text

spring MVC supplied jsp form tags

`-by-`

(a) These tags are given spring MVC framework (pivotal team)

b)These tags support two way data binding when they are used'spring mvc apps

(form data to Model class obj and vice-versa)

(c) default request method for the form page is POST

are

(d) these tags strictly typed

(e) These tags are very much case-sensitive

(f) can be used only in jsp programming

(only in .jsp files)

(g) no tags for submit and reset button (here also use traditional html tags)

(h) Does not support the writing of place holder text