

Rai Studies

Quick Start Site for JAVA Developers

[Home](#)
[Java](#)
[Struts 1](#)
[Spring](#)
[MyBatis\(iBatis\)](#)
[Tomcat](#)
[About Me](#)

Ajax with Spring MVC 3 using Annotations and JQuery

Posted by Rahul

[Go to comments](#)[Leave a comment](#)

Like  0  16  9

Its always been fun for me to work with Ajax! Is not it ? I will make it easy for you to use Ajax with **Spring MVC 3 and JQuery**. This post will illustrate you how to use Ajax in real life practices of industrial coding. As usual, we will take an practical example of Ajax in Spring MVC 3 framework and will implement it and I will make the implementation easy by make you understand the topic.

Let us see what is our example's requirement and how Spring MVC 3 Ajax facility will fulfill it :

In our example, we will make a list of students with name and highest education level, to send the list to the placement office so that the students can get chance. We will make the "Add Student Form" available to the student online so that they can submit their name online and get registered. As a lot of students will use the system, so the performance of the system may very much low. To increase to performance of the web application we will use Ajax with Spring MVC 3 Framework and JQuery.

Following steps we have to go through to implement our example :

1. First of all, we will create a domain class (User.java) that will hold the value of student information.
2. After that we will create our controller class (UserController.java) to handle HTTP request. Our controller will handle three types of requests. First, to show the "Add Student Form", second to handle Ajax request came from "Add Student Form" and add the students to a list, third to show the student information as a list.
3. Then, we will create jsp page (AddUser.jsp) to show "Add Student Form" that will use JQuery to send Ajax request to the Spring MVC Controller. The jsp will also confirm to the user that Student has been added to the list.
4. Then, we will create a jsp (ShowUsers.jsp) that will list all users in the list.

User.java

User.java has two properties name and education to store the student information. Following is the code of User.java :

```

1 package com.raistudies.domain;
2
3 public class User {
4
5     private String name = null;
6     private String education = null;
7
8     // Getter and Setter are omitted for making the code short
9 }

```

UserController.java

Controllers has three method to handle three request urls. "showForm" method handle the request for showing the form to the user. Bellow code shows the UserController.java :

```

1 package com.raistudies.controllers;
2
3 import java.util.ArrayList;
4 import java.util.List;
5
6 import org.springframework.stereotype.Controller;
7 import org.springframework.ui.ModelMap;
8 import org.springframework.validation.BindingResult;
9 import org.springframework.web.bind.annotation.ModelAttribute;
10 import org.springframework.web.bind.annotation.RequestMapping;
11 import org.springframework.web.bind.annotation.RequestMethod;
12 import org.springframework.web.bind.annotation.ResponseBody;
13
14 import com.raistudies.domain.User;

```

 Search: type, hit enter

Tutorials

[C Jstl Tutorial](#)
[Java Tutorial](#)
[MyBatis 3 Tutorial](#)
[Spring IoC Tutorials](#)
[Spring MVC Tutorial](#)
[Spring Security Tutorial](#)
[Struts Tutorial](#)

Most Popular Tutorials

[Ajax Form Validation using Spring MVC and JQuery](#)
[Ajax with Spring MVC 3 using Annotations and JQuery](#)
[Ajax with Struts Example](#)
[DispatchAction in struts example](#)
[DynaActionForm Example With Struts](#)
[File upload using Spring MVC and annotation configuration](#)
[Form Processing in Spring MVC 3 using Annotation](#)
[Simple Form Validation with Struts 1 Example](#)

Spring MVC

[Hello World Example with annotation driven Spring 3 MVC in Eclipse](#)
[Hello World Example with xml configuration Spring MVC in Eclipse](#)
[Form Processing in Spring MVC 3 using Annotation](#)
[Form Processing in Spring MVC 3 using xml configuration](#)
[Form Validation in Spring MVC 3 using XML Configuration](#)
[Form Validation in Spring MVC 3 using Hibernate Validator \(JSR 303\)](#)
[Spring MVC and Json with JQuery](#)
[File upload using Spring MVC and annotation configuration](#)
[Spring MVC BeanNameUrlHandlerMapping Example](#)
[Spring MVC SimpleUrlHandlerMapping example](#)

```

15
16 @Controller
17 public class UserListController {
18     private List<User> userList = new ArrayList<User>();
19
20     @RequestMapping(value="/AddUser.htm",method=RequestMethod.GET)
21     public String showForm(){
22         return "AddUser";
23     }
24
25     @RequestMapping(value="/AddUser.htm",method=RequestMethod.POST)
26     public @ResponseBody String addUser(@ModelAttribute(value="user") User user)
27     {
28         String returnText;
29         if(!result.hasErrors()){
30             userList.add(user);
31             returnText = "User has been added to the list. Total number of user:
32         }else{
33             returnText = "Sorry, an error has occur. User has not been added to
34         }
35         return returnText;
36     }
37
38     @RequestMapping(value="/ShowUsers.htm")
39     public String showUsers(ModelMap model){
40         model.addAttribute("Users", userList);
41         return "ShowUsers";
42     }

```

Recent Tutorials

[JQuery Selectors – by id, name or tag](#)
[Spring MVC 3 CRUD example with MyBatis 3](#)
[Spring MVC 3 controller for performing CRUD operation using MyBatis\(iBatis\) 3](#)
[Ajax Form Validation using Spring MVC and JQuery](#)
[Integrating MyBatis 3 and Spring frameworks – Part 2](#)

“addUsers” is same as the controller method that handle form expect that it also contain annotation **@ResponseBody**, which tells **Spring MVC** that the String returned by the method is the response to the request, it does not have to find view for this string. So the returning String will be send back to the browser as response and hence the Ajax request will work. “showUsers” method is used to show the list of the students to the user.

AddUser.jsp

AddUser.jsp contain a simple form to collect information about the student and uses JQuery JavaScript framework to generate Ajax request to the server. Following is the code in AddUser.jsp :

```

1 <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2 pageEncoding="ISO-8859-1"%>
3 <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.o
4 <html>
5 <head>
6 <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1"
7 <title>Add Users using ajax</title>
8 <script src="/AjaxWithSpringMVC2Annotations/js/jquery.js"></script>
9 <script type="text/javascript">
10 function doAjaxPost() {
11     // get the form values
12     var name = $('#name').val();
13     var education = $('#education').val();
14
15     $.ajax({
16         type: "POST",
17         url: "/AjaxWithSpringMVC2Annotations/AddUser.htm",
18         data: "name=" + name + "&education=" + education,
19         success: function(response){
20             // we have the response
21             $('#info').html(response);
22             $('#name').val('');
23             $('#education').val('');
24         },
25         error: function(e){
26             alert('Error: ' + e);
27         }
28     });
29 }
30 </script>
31 </head>
32 <body>
33 <h1>Add Users using Ajax .....</h1>
34 <table>
35 <tr><td>Enter your name : </td><td> <input type="text" id="name"><b
36 <tr><td>Education : </td><td> <input type="text" id="education"><br
37 <tr><td colspan="2"><input type="button" value="Add Users" onclick=
38 <tr><td colspan="2"><div id="info" style="color: green;"></div></td>
39 </table>
40 <a href="/AjaxWithSpringMVC2Annotations/ShowUsers.htm">Show All Users</a>
41 </body>
42 </html>

```

You may be little bit confused if you are not aware of JQuery. Here is the explanation of the JQuery code :

1. **var name = \$('#name').val();** :- here the \$ is JQuery selector that uses to select any node in HTML whose identifier is passed as argument. If the identifier is a prefix with # that means it is a id of the HTML node. Here, **\$('#name').val()** contains the value of the HTML node whose is "name". The text box in which user will enter her/his name is with id as name. so java script variable name will contain the name of the user.
2. **\$.ajax()** :- It is the method in \$ variable of JQuery to call Ajax. It has five arguments here. First of all **"type"** which indicated the request type of Ajax. It can be POST or GET. Then, **"url"** which indicates the url to be hit of Ajax submission. **"data"** will contain the raw data to be sent to the server. **"success"** will contain the function code that has to be call if the request get success and server sends an response to the browser. **"error"** will contain the function code that has to be call if the request get any error.
3. **\$('#info').html(response);** :- will set the response of the server in to the div. In this way "Hello" + name will be shown in the div whose id is "info".

ShowUsers.jsp

Following are the code in ShowUsers.jsp to print all student information from a ArrayList to jsp page :

```

1 <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2   pageEncoding="ISO-8859-1"%>
3 <%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
4 <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.o
5 <html>
6   <head>
7     <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1"
8     <title>Users Added using Ajax</title>
9   </head>
10  <body style="color: green;">
11    The following are the users added in the list :<br>
12    <ul>
13      <c:forEach items="${Users}" var="user">
14        <li>Name : <c:out value="${user.name}" />; Education : <c:out v
15      </c:forEach>
16    </ul>
17  </body>
18 </html>

```

Here, we have used JSTL core taglib to iterate through the ArrayList and show every value in browser.

- **<c:forEach items="\${Users}" var="user">** : tag is used to iterate through the ArrayList. Property "items" is used to define the bean on which the List object has been stored, so **items="\${Users}"** says that the users list is present in "Users" bean. "var" attribute says the name of the variable in which each user will be stored.
- **<c:out value="\${user.name}" />** : As, a single user will be stored in variable name "user" so to print the name property in User object we use **#{user.name}**.

app-config.xml

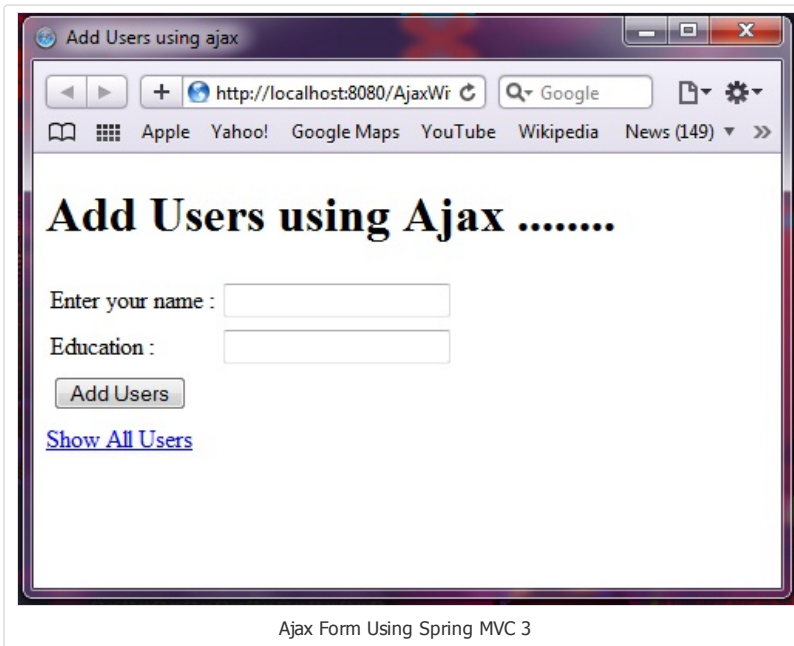
Our Spring MVC configuration file should be able to handle annotation driven controllers. The configuration are as follows :

```

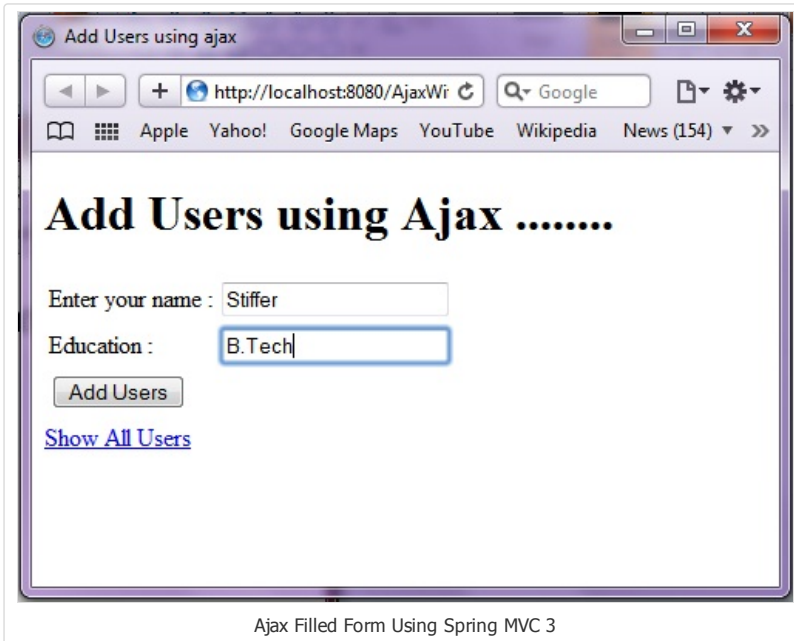
1 <?xml version="1.0" encoding="UTF-8"?>
2 <beans xmlns="http://www.springframework.org/schema/beans"
3   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4   xmlns:context="http://www.springframework.org/schema/context"
5   xmlns:mvc="http://www.springframework.org/schema/mvc"
6   xsi:schemaLocation="
7     http://www.springframework.org/schema/beans http://www.springframework.org/s
8     http://www.springframework.org/schema/context http://www.springframework.org/sc
9     http://www.springframework.org/schema/mvc http://www.springframework.org/schema
10
11   <!-- Scans the classpath of this application for @Components to deploy as b
12   <context:component-scan base-package="com.raistudies" />
13
14   <!-- Configures the @Controller programming model -->
15   <mvc:annotation-driven />
16
17   <!-- Resolves view names to protected .jsp resources within the /WEB-INF/vi
18   <bean id="viewResolver" class="org.springframework.web.servlet.view.Interna
19     <property name="prefix" value="/WEB-INF/jsp/" />
20     <property name="suffix" value=".jsp" />
21   </bean>
22
23 </beans>

```

Deploy the war file to tomcat 6 and hit the url in browser, you will get following page displayed :



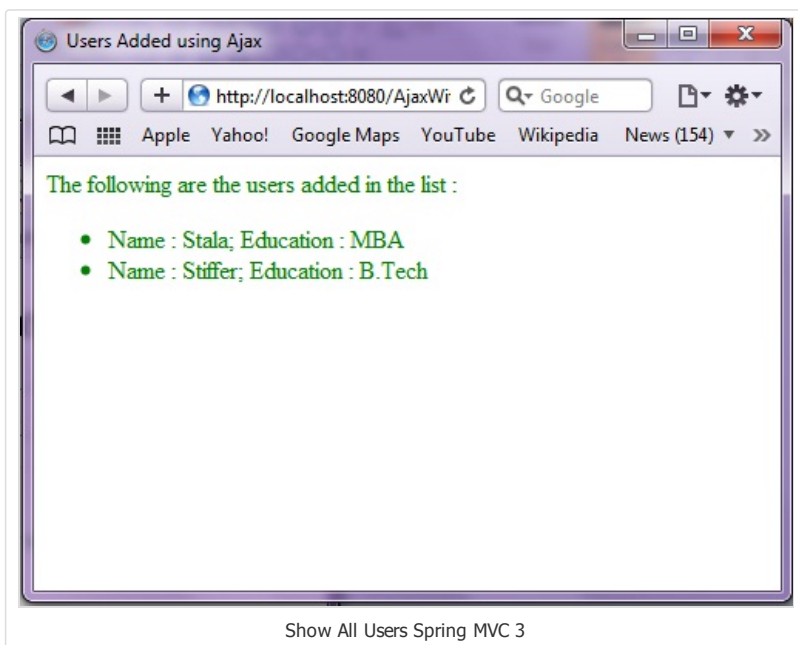
Fill student information :



After that click on "Add Users" button, you will get message that user has been added to the list :



To show all student added to the list click on the button "Show All Users", you will get following page :



That is all from Ajax using Spring MVC 3 and JQuery. You can download source from bellow link.

Source : [Download](#)

War : [Download](#)

- [Post to Delicious](#)
- [Post to Facebook](#)

23

Like  16

Related Posts:

- [Ajax Form Validation using Spring MVC and JQuery](#)
- [Spring MVC and Json with JQuery](#)
- [Spring MVC 3 CRUD example with MyBatis 3](#)
- [Spring MVC 3 controller for performing CRUD operation using MyBatis\(iBatis\) 3](#)
- [Theme Resolver Example in Spring MVC with annotation configuration \(Spring 3.x\)](#)

Ajax, MVC, Spring

[Leave a comment ?](#)[21 Comments.](#)**tuna** December 15, 2011 at 2:51 am[Reply](#)

Really good tutorial regarding using JQuery and Spring together
Thanks

**Easy SEO** January 10, 2012 at 11:53 am[Reply](#)

Awesome things here. I am very glad to peer your post. Thank you so much and I'm
having a look forward to touch you. Will you kindly drop me a e-mail?

**Poteau Oklahoma** January 10, 2012 at 5:40 pm[Reply](#)

I will right away grab your rss as I can't in finding your email subscription link or e-
newsletter service. Do you've any? Kindly allow me realize so that I may subscribe. Thanks.

**Marner Hillary** January 13, 2012 at 11:27 pm[Reply](#)

It's really a nice and helpful piece of information. I'm satisfied that you just shared this
useful info with us. Please keep us up to date like this. Thanks for sharing.

**maschinenfick** January 14, 2012 at 5:16 pm[Reply](#)

I do consider all the ideas you've offered on your post. They are very convincing and will
certainly work. Nonetheless, the posts are too quick for starters. May just you please
lengthen them a little from next time? Thanks for the post.

**pwsfrap** January 17, 2012 at 5:16 am[Reply](#)

Thanks a bunch for sharing this with all of us you really know what you're talking about!
Bookmarked.

**ultimate demon** January 17, 2012 at 11:41 am[Reply](#)

A person necessarily lend a hand to make critically posts I'd state. That is the very first time
I frequented your website page and so far? I surprised with the analysis you made to make
this actual submit incredible. Fantastic job!

**video games reviews** January 19, 2012 at 11:34 am[Reply](#)

Hello there, just changed into aware of your weblog thru Google, and located that it's truly
informative. I am going to be careful for brussels. I'll be grateful in the event you continue
this in future. Many people will likely be benefited from your writing. Cheers!

**suresh** February 2, 2012 at 3:17 pm[Reply](#)

😊
excellent article..no words to describe...kudos to you..

**Bill** March 25, 2012 at 4:08 am[Reply](#)

Great tutorial, but the download zip file was corrupted.

**Rahul** March 25, 2012 at 10:14 am[Reply](#)

Source Code file is checked and war file attached.

**ali** June 27, 2012 at 12:39 am[Reply](#)

Hi Rahul,
i want to pass utf8 data in this example but not possible for me. output of this command
in UserController class is question sign(????...).
command: System.out.print(request.getParameter("name"));

i set Unicode to utf8 in all part such in ajax function and in jsp page and also in
UserController class.
i modify this.

```
in ajax function
*****
$.ajax({
  type: "POST",
  url: contextPath + "/AddUser.htm",
  contentType: "application/x-www-form-urlencoded; charset=UTF-8",
  data: "name=" + name + "&education=" + education,
  *****
in UserController
*****
@RequestMapping(value="/AddUser.htm",method=RequestMethod.POST)
public @ResponseBody JsonResponse addUser(@ModelAttribute(value="user") User user,
BindingResult result,HttpServletRequest request, HttpServletResponse response ) throws
UnsupportedEncodingException{
  request.setCharacterEncoding("utf8");
  response.setCharacterEncoding("utf8");
  *****
and in jsp page change chrset to utf8

but not correctly run.
please help me.
```



Nikhil June 29, 2012 at 12:59 pm

Reply

😊 Thanks a lot!!!!!! much needed at this time



marketing July 24, 2012 at 6:58 am

Reply

I every time spent my half an hour to read this website's articles or reviews every day along
with a mug of coffee.



Mirek July 24, 2012 at 8:43 pm

Reply

If in real world project each developers which wants ajax catability have to write this ugly
javascript code.....

In JSF 2 ajax is co simple



Arat Kumar August 2, 2012 at 5:28 pm

Reply

Nice tutorial.I have once questions how can I define multiple controller?



Google seo August 15, 2012 at 11:05 pm

Reply

Keep this going please, great job!



hämorrhoiden symptome August 18, 2012 at 3:26 am

Reply

This blog was... how do I say it? Relevant!! Finally I have found something that helped me.
Cheers!



programmer tie December 6, 2012 at 2:27 am

Reply

I downloaded the source zip and the war zip. But I'm not able to open the zip files. I get
error "start of central directory not found; Zip file corrupt. Possible cause: file transfer
error".



Mansoor Sajjad December 18, 2012 at 3:09 pm

Reply

Simple Yet powerful 😊



seo company January 1, 2013 at 5:51 am

Reply

Thanks for writing this information within your website.

Leave a Comment

NOTE - You can use these HTML tags and attributes:

```
<a href="" title=""> <abbr title=""> <acronym title=""> <b> <blockquote cite="">
<cite> <code> <del datetime=""> <em> <i> <q cite=""> <strike> <strong>
```













- o C Jstl Tutorial
- o Java Tutorial
- o MyBatis 3 Tutorial
- o Spring IoC Tutorials
- o Spring MVC Tutorial
- o Spring Security Tutorial
- o Struts Tutorial

- o [MyBatis Official Website](#)
- o [Spring IoC Documentation](#)
- o [Spring MVC 2.5.6 Documentation](#)
- o [Spring MVC 3 Documentation](#)
- o [Spring Security Documentation](#)
- o [SpringSource Official Site](#)
- o [Struts 1 official site](#)

Ajax CRUD Database Eclipse
 File Upload Form Validation iBatis
 Insert Integration IoC JQuery Json jstl
 MVC MyBatis Spring
 Spring Security Struts 1
 Theme Tomcat

Hi Friends, I am just trying to help you and also help myself. I like to learn new things and make others learn also. If you get any error in my site, please inform me. Thanks Rahul Mondal