

## **JSP in Core JAVA**

**Create a registration form using JSP**



## Overview of JSP in Core JAVA

There is one more technology in Core JAVA named JSP technology which can be used for creation of web application like Servlet technology. It can be considered as an extension to Servlet Technology because it provides many number of functionality to developer than servlet (like, expression language, JSTL, etc.)

In a JSP page, we consists of some HTML and JSP tags. Actually the maintenance of a JSP page is easier to maintain a Servlet because developer can separate designing and development in JSP. JSP pages provide some additional features (like, Expression Language, Custom Tags, etc.)

## Advantages of JSP over Servlet in Core JAVA

There are many advantages of JSP and the Servlet. Some are as followings:

### 1- **Extension of Servlet**

We have already discussed that JSP is a technology which is the extension of Servlet technology in JAVA. Although developer can use all features of the Servlet in JSP technology. In addition, developer can use predefined tags, implicit objects, Custom tags and expression language in JSP technology that makes easy development of JSP pages.

### 2- **Easy maintenance**

Any developer can separate our business logic along with presentation logic in easy way so we can say that JSP can be easily managed than Servlet. In case of Servlet, developer mix his/her business logic along with the presentation logic.

### 3- **Develop fast**

There is no need to re-compile and re-deploy if developer modifies the existing JSP page. But in case of Servlet, developer must need to re-compiled and re-deploy if he have changed the feel and look of the application.

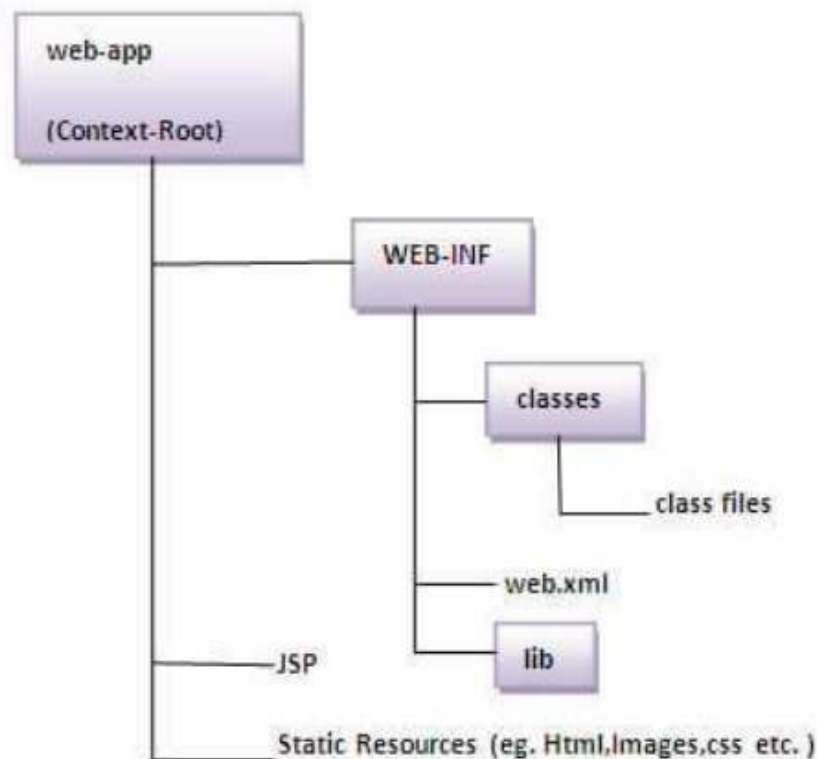
### 4- **Require less code**

In JSP technology, developer can use many tags (like: JSTL, action tags, custom tags, etc.) These tags reduce the lines of code.

## The lifecycle of a JSP page

- Translation process of JSP Page
- Compilation process of JSP Page
- Process of class loading (classloader loads all class files)
- Process of instantiation (generat the object of newly created servlet is)
- Process of initialization (invokes jspInit() method by container)
- Process of request processing (invokes \_jspService() method by container)

## The Directory structure of JSP Application



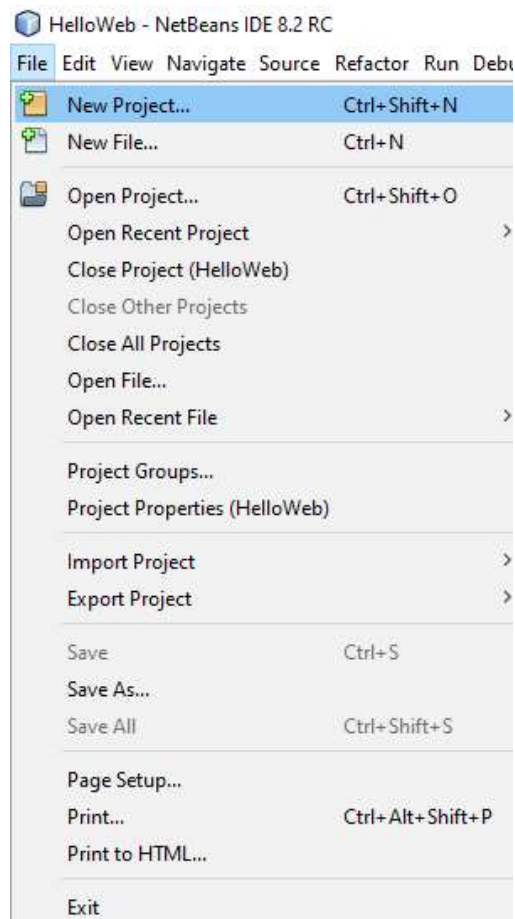
# Create a simple registration application form using JSP in Core JAVA

## Software Requirements

NetBeans JAVA IDE, Apache Tomcat Server, JAVA 8

### Step 1: Open\_NetBeans JAVA IDE

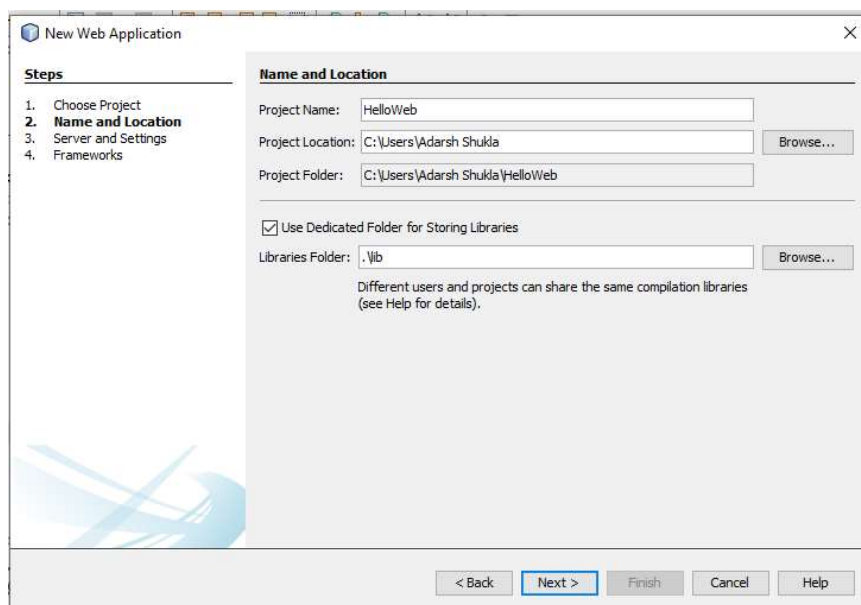
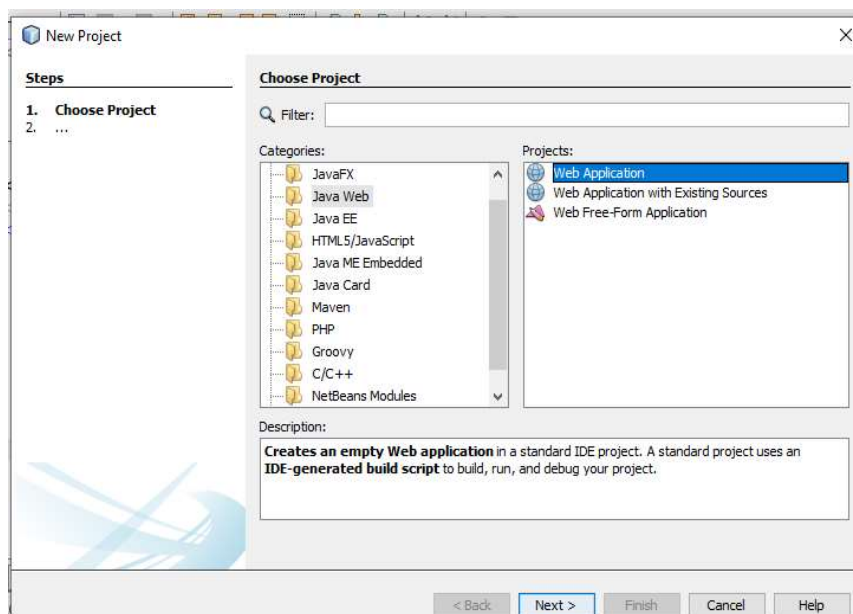
### Step 2: Go to File menu → New Project

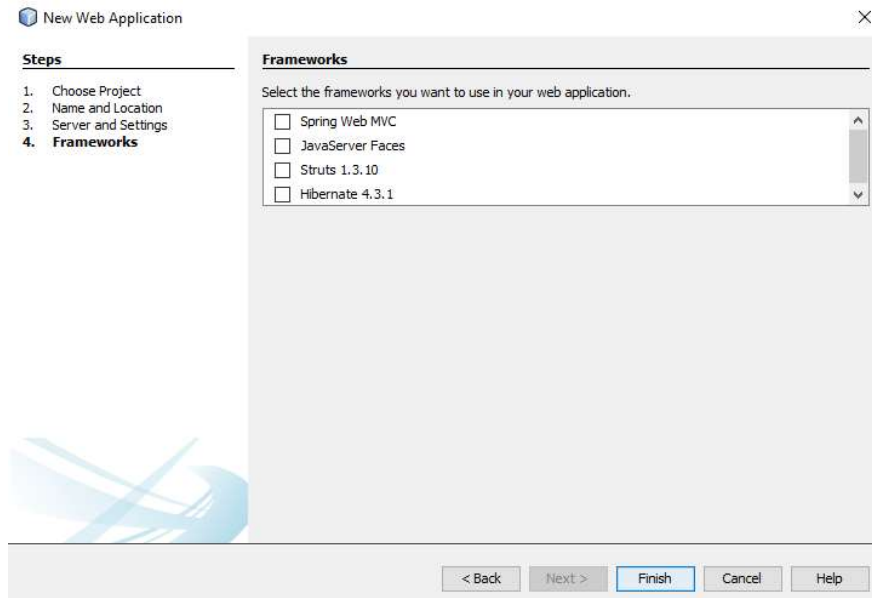
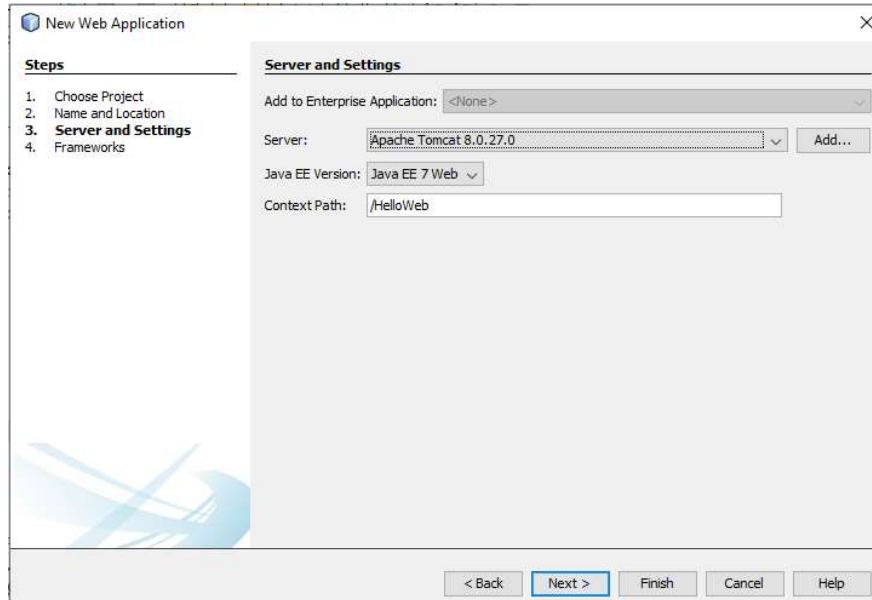


**Step 3:** Now you have to set 4 things:

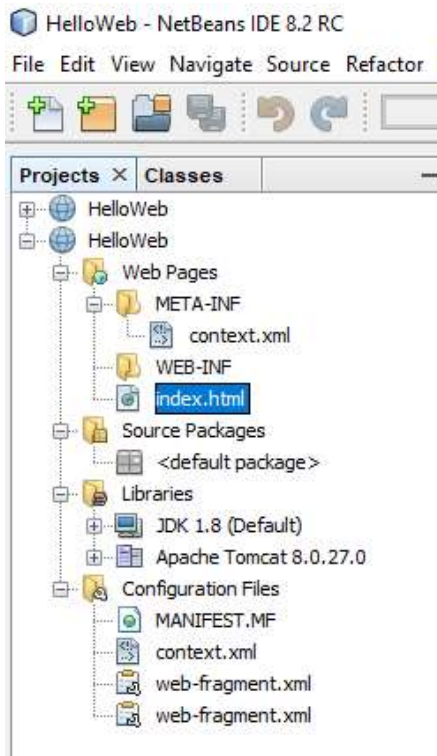
- a- Choice of project
- b- Project name and location
- c- Server and settings
- d- Framework

Click on Java Web → Web Application → Next → Type Project Name → Select Location → Next → Next → Finish (Refer below screenshots)

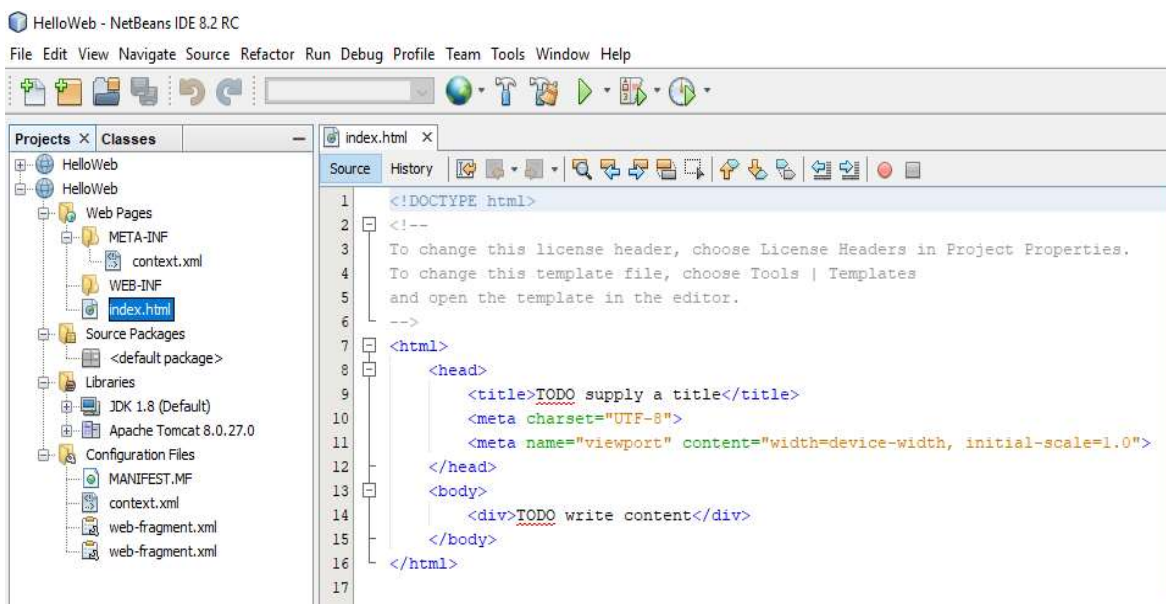




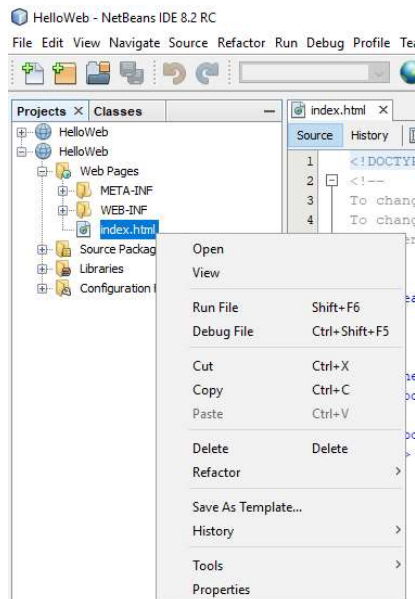
**Step 4:** After click on 'Finish' in project section, a hierarichical directory structure will be generated



You can see the content of index.html by clicking on '*index.html*' in directory structure file which is generated by default



### **Step 5:** Delete '*index.html*' file from directory



Click on Delete.

### **Step 6:** Go to the location in disctionary i.e. HelloWeb(App) → WEB-INF (Refer hierarchical structure from step 4)

Create 2 new files here:

A- index.jsp

B- response.jsp

Content of index.jsp file:

Copy the content of this file from here

<https://github.com/Ad1106/JSP-Registration-form/blob/main/index.jsp>

Content of response.jsp file:

Copy the content of this file from here

<https://github.com/Ad1106/JSP-Registration-form/blob/main/response.jsp>



**Step 7:** Go to the location in disctionary i.e. HelloWorld(App) → Source Packages  
→ org.mypackage.hello (Refer hierarchical structure from step 4)

Create a new java files here named '*NameHandler.java*'

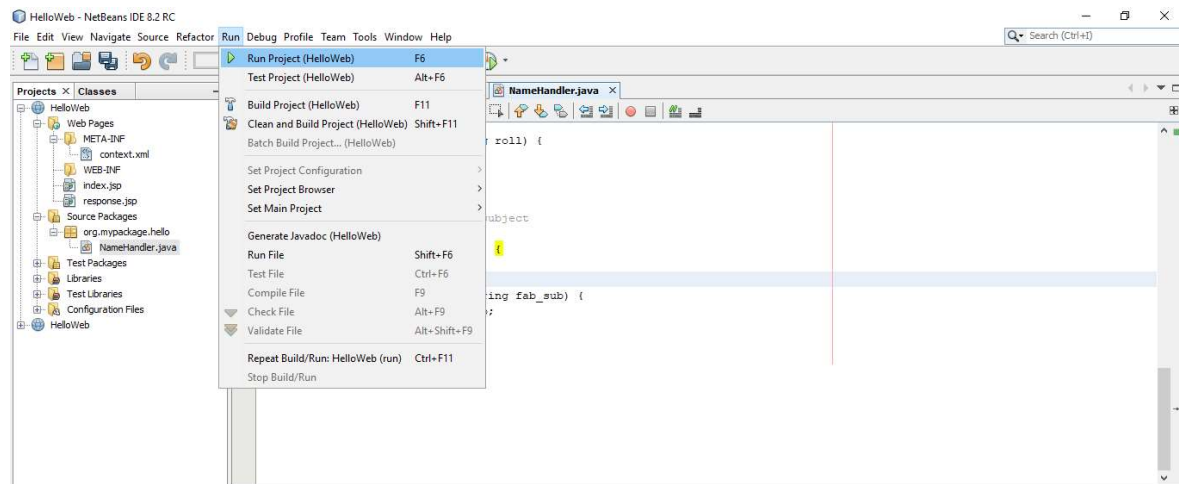
Content of '*NameHandler.java*' file:

Copy the content of this file from here

<https://github.com/Ad1106/JSP-Registration-form/blob/main/NameHandler.java>

**Step 8:** After this all configurations of our web app are set. This is the time of run our web-app.

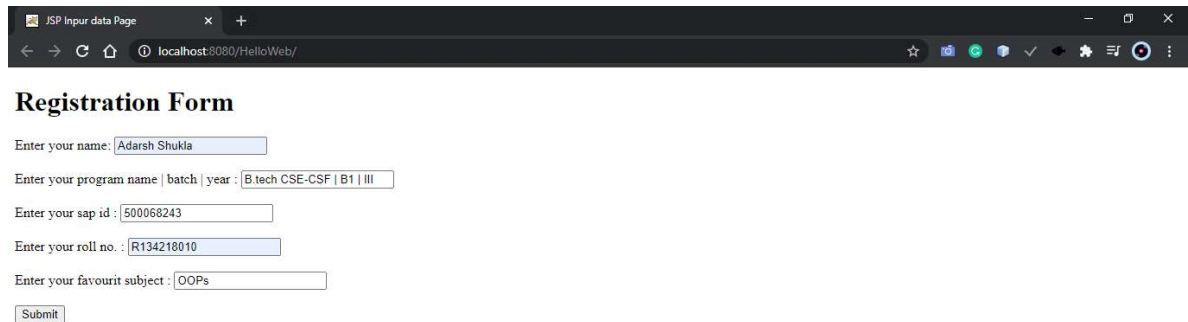
To run the project click on Run → Run Project



**Step 9:** After running, it will automatically navigate to your default browser and open following tab (Registration form)

A screenshot of a web browser window. The address bar shows 'localhost:8080/HelloWeb/'. The page title is 'JSP Input data Page'. The main content area displays a 'Registration Form' with several input fields: 'Enter your name:', 'Enter your program name | batch | year:', 'Enter your sap id:', 'Enter your roll no.:', and 'Enter your favourit subject:'. There is a 'Submit' button at the bottom left of the form.

## **Step 10:** Fill the form and click on ‘Submit’ button



The screenshot shows a web browser window with the title 'JSP Input data Page'. The address bar shows 'localhost:8080/HelloWeb/'. The page contains a registration form with the following fields and values:

- Enter your name:
- Enter your program name | batch | year :
- Enter your sap id :
- Enter your roll no. :
- Enter your favourit subject :
- 

## **Step 11:** After this, one more tab will be open. This is our response tab with respect to input data



The screenshot shows a web browser window with the title 'JSP Response viewer page'. The address bar shows 'localhost:8080/HelloWeb/response.jsp?name=Adarsh+Shukla&program=B.tech+CSE-CSF+%7C+B1+%7C+III&sap=500068243&roll=R1...'. The page displays the following information:

- Name :** Adarsh Shukla
- Program name :** B.tech CSE-CSF | B1 | III
- Sap id :** 500068243
- Roll no. :** R134218010
- Fabourite Subject :** OOPs

## **Conclusion**

In this article, First I explain some theoritical concepts of JSP in Core JAVA and second how can we create a simple registration form web – app in Core JAVA using JSP and NetBeans IDE.