

# JSP Exception Handling with Error Pages

Sometime back I wrote a post about [Servlet Exception Handling](#) and why do we need it.

The same explanation is also applicable for JSP pages also and that's why Java EE provides a clear approach for exception handling in JSP pages using error pages.

To handle exceptions thrown by the JSP page, all we need is an error page and define the error page in JSP using [page directive](#).

## JSP Error Page

To create a JSP error page, we need to set page directive attribute **isErrorPage** value to true, then we can access exception [implicit object](#) in the JSP and use it to send customized error message to the client.

## JSP Error Page Configuration

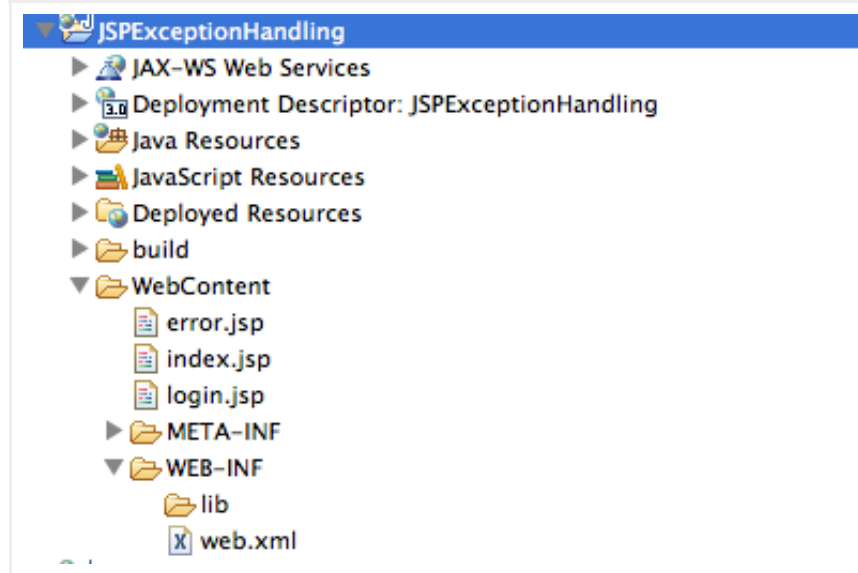
We need to set page directive **errorPage** attribute to define the JSP that will handle any exception thrown by the JSP service method. When JSP Error page is translated to servlet code, it extends `org.apache.jasper.runtime.HttpJspBase` in Tomcat.

## Error Page Deployment Descriptor Configuration

Most of the times, we have a common error page that we want to use for all the JSPs, so rather than configuring it in all the JSPs individually, we can define error page in web.xml with **error-page** element. We can configure JSP error page to handle other error codes like 404 also.

Let's see how all these fit together in a web application.

We will create a simple web application JSPExceptionHandling whose project structure will look like below image.



The entry point of the application is index.jsp whose code is given below.

index.jsp

```
1  <%@ page language="java" contentType="text/html; charset=US-ASCII"
2      pageEncoding="US-ASCII"%>
3  <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3
4  <html>
5  <head>
6  <meta http-equiv="Content-Type" content="text/html; charset=US-ASCII">
7  <title>Login Page</title>
8  </head>
9  <body>
10 <form action="login.jsp" method="post">
11 <strong>User ID</strong>:<input type="text" name="id"><br>
12 <strong>Password</strong>:<input type="password" name="password"><br>
13 <input type="submit" value="Login">
14 </form>
15 </body>
16 </html>
```

When we submit the form, request will be sent to login.jsp, code is like below.

login.jsp

```
1  <%@ page language="java" contentType="text/html; charset=US-ASCII"
2      pageEncoding="US-ASCII" errorPage="error.jsp"%>
3  <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3
4  <html>
5  <head>
6  <meta http-equiv="Content-Type" content="text/html; charset=US-ASCII">
7  <title>User Home Page</title>
8  </head>
9  <body>
10 <%
11     String user = request.getParameter("id");
12     String pwd = request.getParameter("password");
13
14     if(user == null || "".equals(user) || pwd == null || "".equals(pwd)){
15         throw new ServletException("Mandatory Parameter missing");
16     }
17
18 %>
```

```

20 <!-- do some DB processing, not doing anything for simplicity --%>
21 Hi <%=user %>
22 </body>
23 </html>

```

Notice that if input parameters are null or empty, its throwing ServletException with a proper message and errorPage is defined as error.jsp whose code is like below.

error.jsp

```

1 <%@ page language="java" contentType="text/html; charset=US-ASCII"
2   pageEncoding="US-ASCII" isErrorPage="true"%>
3 <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3
4 <html>
5 <head>
6 <meta http-equiv="Content-Type" content="text/html; charset=US-ASCII">
7 <title>Error Page</title>
8 </head>
9 <body>
10 <% if(response.getStatus() == 500){ %>
11 <font color="red">Error: <%=exception.getMessage() %></font><br>
12
13 <!-- include login page --%>
14 <%@ include file="index.jsp"%>
15 <%}else {%>
16 Hi There, error code is <%=response.getStatus() %><br>
17 Please go to <a href="/index.jsp">home page</a>
18 <%} %>
19 </body>
20 </html>

```

Notice the `isErrorPage` page directive attribute value is `true`. When application resources throw exceptions, the error code is 500, the code is written to handle both application level exceptions and errors such as 404 – page not found.

Also notice the use of include directive to present user with login page incase of any exception.

Here is the web.xml where we are defining the error page for the application.

web.xml

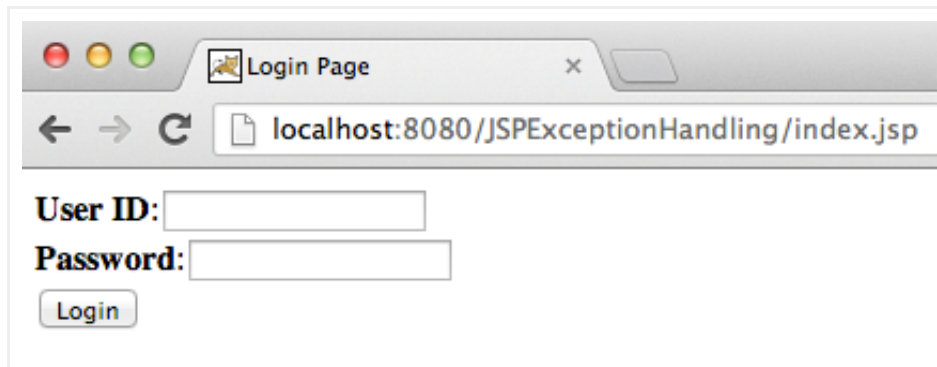
```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://
3   <display-name>JSPExceptionHandling</display-name>
4   <welcome-file-list>
5     <welcome-file>index.jsp</welcome-file>
6   </welcome-file-list>
7
8   <error-page>
9     <error-code>404</error-code>
10    <location>/error.jsp</location>
11  </error-page>
12
13  <error-page>
14    <exception-type>java.lang.Throwable</exception-type>
15    <location>/error.jsp</location>
16  </error-page>
17

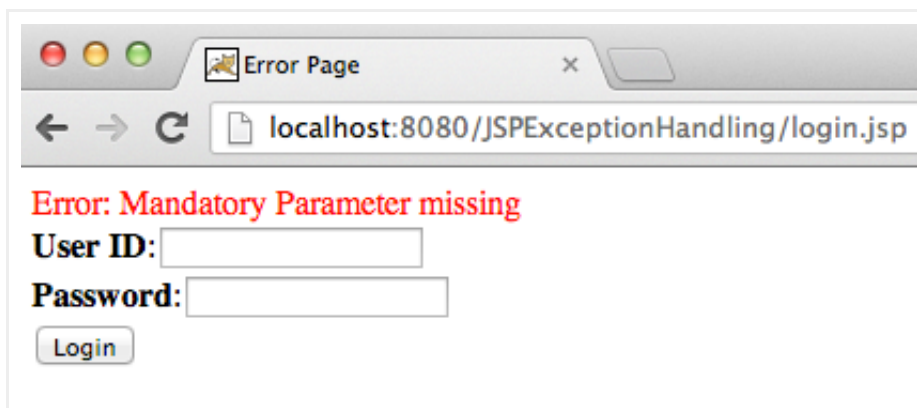
```

Now when we run above application, we get following pages as response.

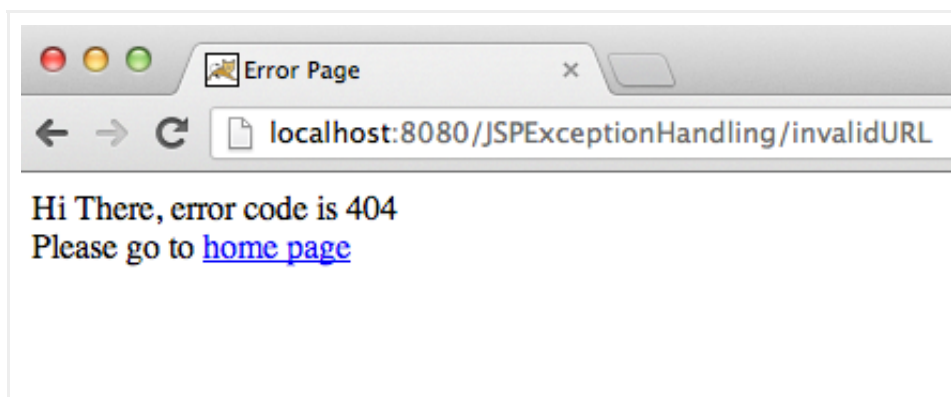
### Login Page



### JSP Error Page for Exception



### JSP Error Page for 404 Error Code



Thats all for exception handling in JSP pages, its very easy to implement and we should use it to make sure we handle all the exceptions and error codes and send a useful response to client rather than container default error page.

---