

# B.Sc. In Software Development. Year 4.

## Semester I. Enterprise Development.

### JSP Error Pages and Directives.

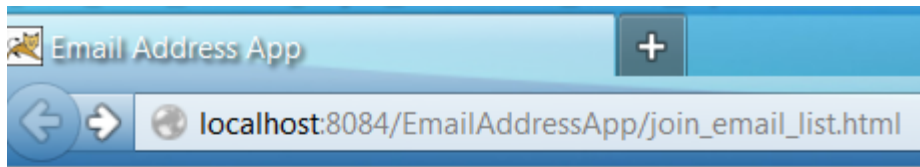


**LIMERICK INSTITUTE  
OF TECHNOLOGY**  
**SCHOOL OF SCIENCE,  
ENGINEERING & I.T.**

*Department of Information Technology*

# Working with Errors

To demonstrate how JSP applications can easily work with error pages consider the following enhancement to the email address application.



## Join Our Email List

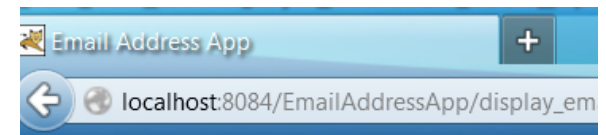
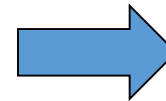
To join our email list, enter your name and email address below,

First name:

Last name:

Email address:

Age:



## Thanks for joining our email list

Here is the information you entered:

First name:	Dave
Last name:	Hopkins
Email address:	dave@lit.ie
Age:	29

To enter another email address, click on the Back button in your browser or the Return button shown below.

# Enhanced User class

The User class was augmented with an extra instance variable “age” and this was also reflected in the constructor. A getter and setter for age was also included in the class (not shown below).

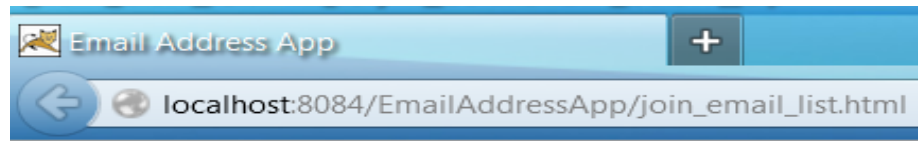
```
4  public class User {
5
6      private String firstName;
7      private String lastName;
8      private String emailAddress;
9      private int age;
10
11     public User() {
12     }
13
14     public User(String firstName, String lastName, String emailAddress, int age) {
15         this.firstName = firstName;
16         this.lastName = lastName;
17         this.emailAddress = emailAddress;
18         this.age = age;
19     }
20
21     /**
22      * @return the firstName
23      */
24     public String getFirstName() {
```

# Enhanced JSP

```
9      </head>
10     <body>
11         <%@ page import="business.User, data.UserIO" %>
12
13         <%
14
15             String firstName = request.getParameter("firstName");
16             String lastName = request.getParameter("lastName");
17             String emailAddress = request.getParameter("emailAddress");
18             String ageInput = request.getParameter("age");
19
20             int age = Integer.parseInt(ageInput);
21
22             ServletContext sc = this.getServletContext();
23
24             String path = sc.getRealPath("/WEB-INF/EmailList.txt");
25
26             User user = new User(firstName, lastName, emailAddress, age);
27
```

# Working with Errors

Consider the following sample run of the web app.



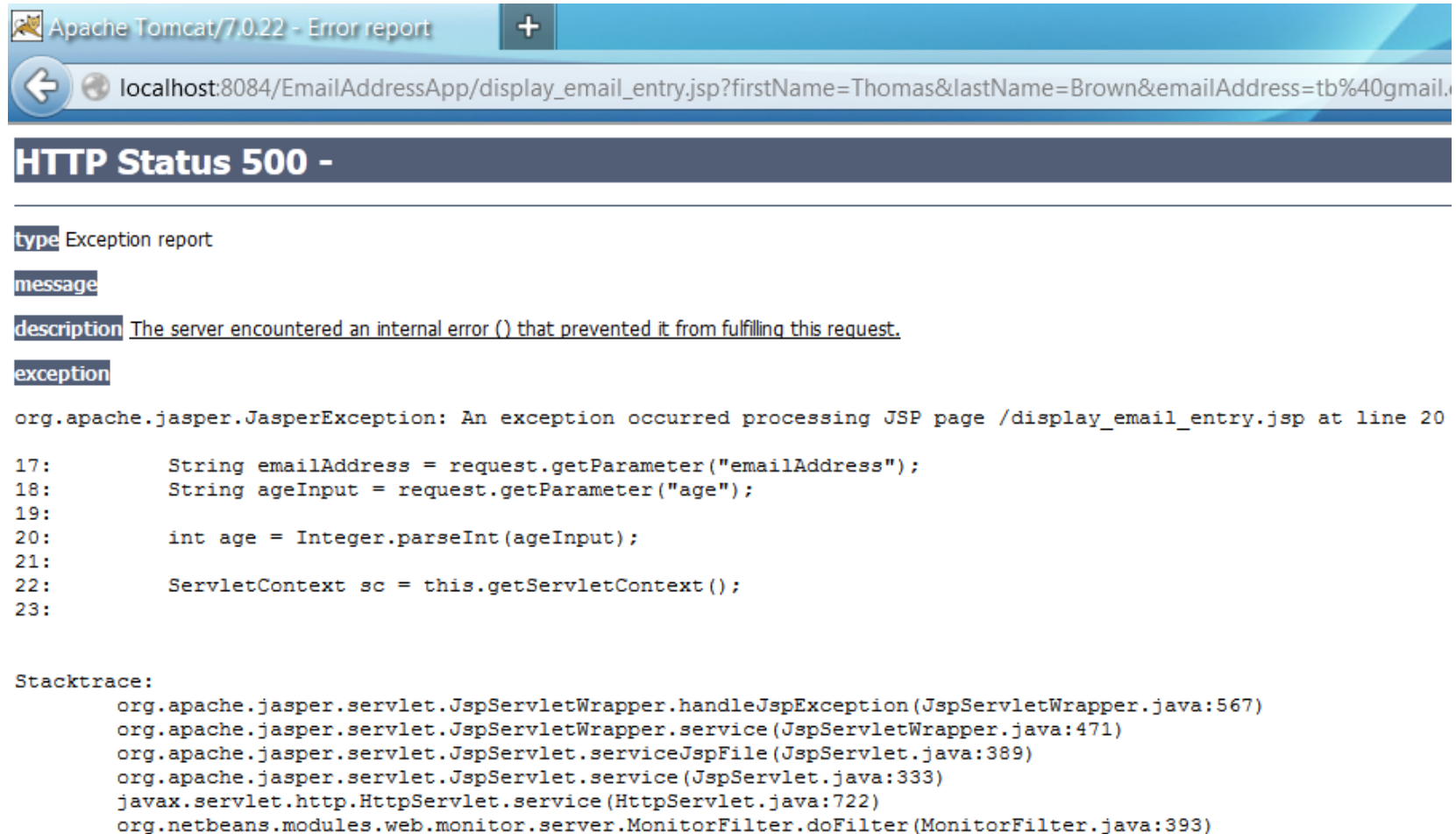
## Join Our Email List

To join our email list, enter your name and email address below,

First name:	<input type="text" value="Thomas"/>
Last name:	<input type="text" value="Brown"/>
Email address:	<input type="text" value="tb@gmail.com"/>
Age:	<input type="text" value="Twenty Two"/>
<input type="button" value="Submit"/>	

The application is expecting the user to enter a numeric value for their age so when they enter a String “Twenty Two” my application is going to crash!

# Working with Errors



The screenshot shows a web browser window with the title "Apache Tomcat/7.0.22 - Error report". The address bar shows the URL "localhost:8084/EmailAddressApp/display\_email\_entry.jsp?firstName=Thomas&lastName=Brown&emailAddress=tb%40gmail.com". The main content area has a dark blue header with the text "HTTP Status 500 -". Below this, there are sections for "type", "message", "description", and "exception". The "description" section contains the text "The server encountered an internal error () that prevented it from fulfilling this request." The "exception" section contains the following text:

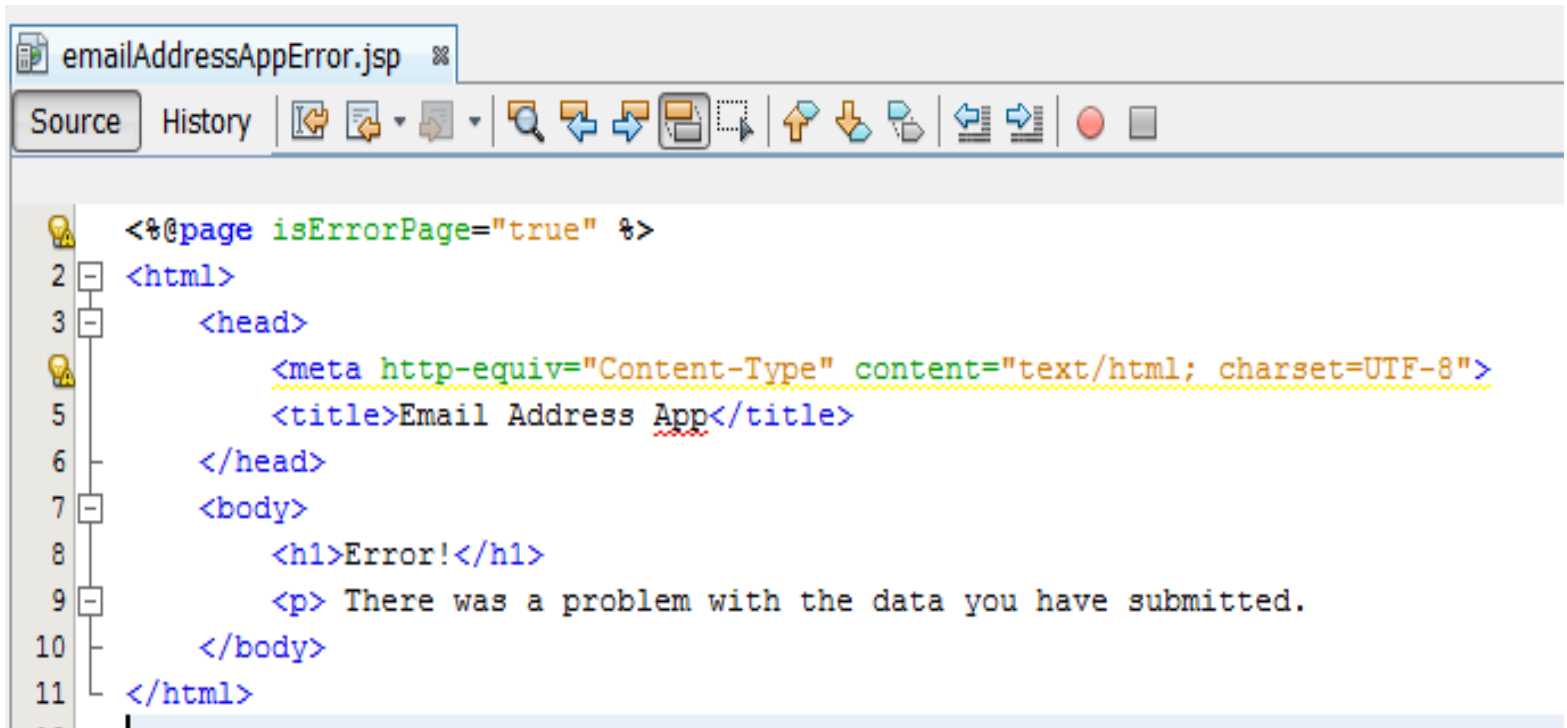
```
org.apache.jasper.JasperException: An exception occurred processing JSP page /display_email_entry.jsp at line 20

17:         String emailAddress = request.getParameter("emailAddress");
18:         String ageInput = request.getParameter("age");
19:
20:         int age = Integer.parseInt(ageInput);
21:
22:         ServletContext sc = this.getServletContext();
23:

Stacktrace:
    org.apache.jasper.servlet.JspServletWrapper.handleJspException(JspServletWrapper.java:567)
    org.apache.jasper.servlet.JspServletWrapper.service(JspServletWrapper.java:471)
    org.apache.jasper.servlet.JspServlet.serviceJspFile(JspServlet.java:389)
    org.apache.jasper.servlet.JspServlet.service(JspServlet.java:333)
    javax.servlet.http.HttpServlet.service(HttpServlet.java:722)
    org.netbeans.modules.web.monitor.server.MonitorFilter.doFilter(MonitorFilter.java:393)
```

# Working with Errors

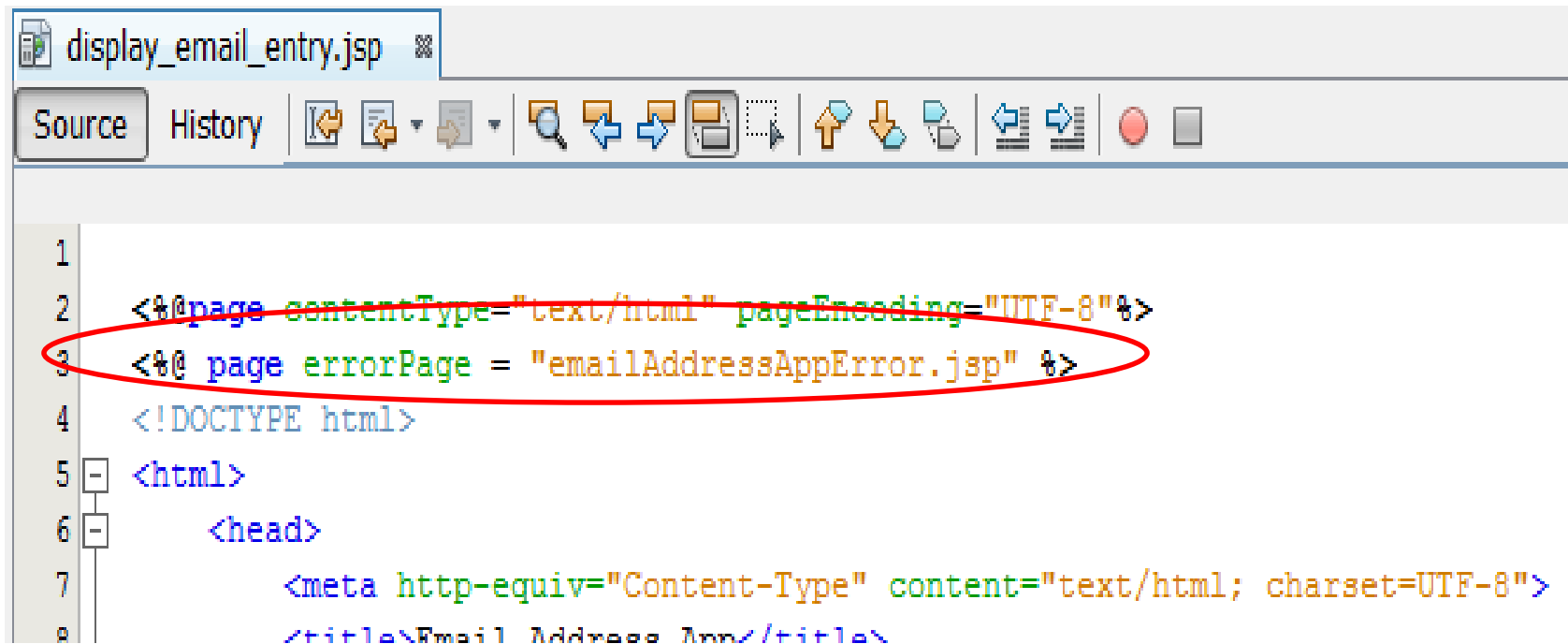
The first thing we need to do is add a JSP to the project. I've called mine "display\_email\_entry.jsp".

A screenshot of an IDE window titled "emailAddressAppError.jsp". The window has a toolbar with icons for source, history, and various editing actions. The code is written in JSP syntax and is as follows:

```
<%@page isErrorPage="true" %>
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <title>Email Address App</title>
</head>
<body>
  <h1>Error!</h1>
  <p> There was a problem with the data you have submitted.
</body>
</html>
```

# Working with Errors

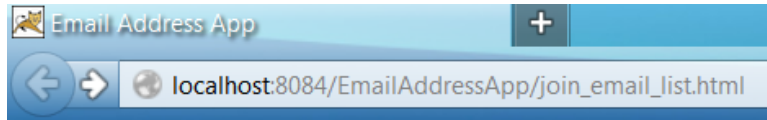
The next thing that needs to be done is to add one line to the original JSP that processes the form.



```
1
2 <%@page contentType="text/html" pageEncoding="UTF-8"%>
3 <%@ page errorPage = "emailAddressAppError.jsp" %>
4 <!DOCTYPE html>
5 <html>
6     <head>
7         <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
8         <title>Email Address App</title>
```



# Working with Errors



## Join Our Email List

To join our email list, enter your name and email address below,

First name:

Last name:

Email address:

Age:



## Error!

There was a problem with the data you have submitted.

# Working with Errors

The error page will be displayed no matter what exception is thrown during the execution of the JSP.

The error handling was made possible by using the page directive.

Page is one of three directives within JSP.

# JSP Directives

A JSP directive is a statement that gives the JSP engine information about the JSP page. For example, if your JSP page uses a Java class from a package other than the `java.lang` package, you have to use a directive to import this package. The general syntax for a JSP directive is as follows:

`<% @ directive attribute="value" %>`, or

`<% @ directive attribute1="value1"`

`attribute2="value2"`

`...`

`attributen="valuen" %>`

# JSP Directives

There are 3 possible JSP directives:

1. page.
2. include
3. taglib.

# JSP Directives: page

*page* lets you provide information for the page, such as importing classes and setting up content type.

The page directive can appear anywhere in the JSP file.

# JSP Directives: include

*include* lets you insert a file to the servlet when the page is translated to a servlet.

The include directive must be placed where you want the file to be inserted.

This is useful for inserting headers, footers, menu-bars etc into your JSP.

# JSP Directives: taglib

*taglib* lets you define custom tags.

# Attributes for the page directive: import

Specifies one or more packages to be imported for this page. For example, the directive

```
<% @ page import="java.util.*, java.text.*" %>
```

imports `java.util.*` and `java.text.*`.



# Attributes for the page directive: contentType

Specifies the MIME type for the resultant JSP page.

By default, the content type is text/html for JSP.

The default content type for servlets is text/plain.

# Attributes for the page directive: session

Specifies a boolean value to indicate whether the page is part of the session.

By default, session is true.

# Attributes for the page directive: `errorPage`

`errorPage` specifies a JSP page that is processed when an exception occurs in the current page.

For example, the directive

```
<% @ page errorPage="HandleError.jsp" %>
```

specifies that `HandleError.jsp` is processed when an exception occurs.

# Attributes for the page directive: `isErrorpage`

`isErrorPage` specifies a boolean value to indicate whether the page can be used as an error page.

By default, this attribute is false.

# References

Murach, J., (2014) *Murachs Java Servlets JSP*, 3rd edn.  
Mike Murach and Associates, Inc.

Y. Daniel Liang (2014) *Introduction to Java Programming, Comprehensive Version*, 10th edn.  
Pearson.

<http://docs.oracle.com/javaee/6/tutorial/doc/>