# JSP Elements – Declaration, Syntax & Expression

By James Hartman                                                  May 27, 2023

In this tutorial, we will be learning the basic tags of JSP and how to add comments into JSP. Along with this, we will also create a JSP and run that JSP on the server.

## JSP Declaration

- A declaration tag is a piece of Java code for declaring variables, methods and classes. If we declare a variable or method inside declaration tag it means that the declaration is made inside the servlet class but outside the service method.
- We can declare a static member, an instance variable (can declare a number or string) and methods inside the declaration tag.

**Syntax of declaration tag:**

```
<%! Dec var %>
```

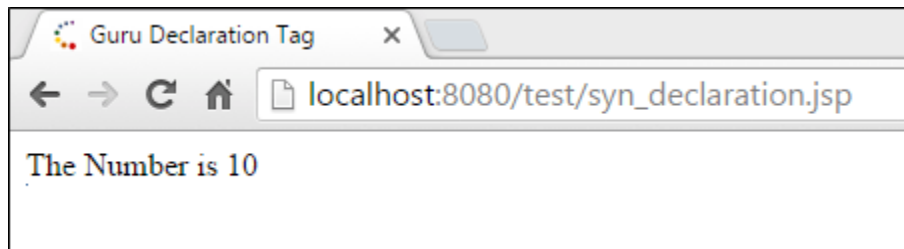Here Dec var is the method or a variable inside the declaration tag.

Example:

In this example, we are going to use the declaration tags

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=ISO-8859-1">
<title>Guru Declaration Tag</title>
</head>
<body>
<%! int count =10; %>
<% out.println("The Number is " +count); %>
</body>
</html>
```

Explanation the code:

Code Line 10: Here we are using declaration tag for initializing a variable count to 10.

When you execute the above code you get the following output:



Output:

The variable which is declared in the declaration tag is printed as output.

# JSP Scriptlet

- Scriptlet tag allows to write Java code into JSP file.
- JSP container moves statements in _jspservice() method while generating servlet from jsp.
- For each request of the client, service method of the JSP gets invoked hence the code inside the Scriptlet executes for every request.
- A Scriptlet contains java code that is executed every time JSP is invoked.

Syntax of Scriptlet tag:

```
<% java code %>
```

Here <%%> tags are scriplets tag and within it, we can place java code.

Example:

In this example, we are taking Scriptlet tags which enclose java code.
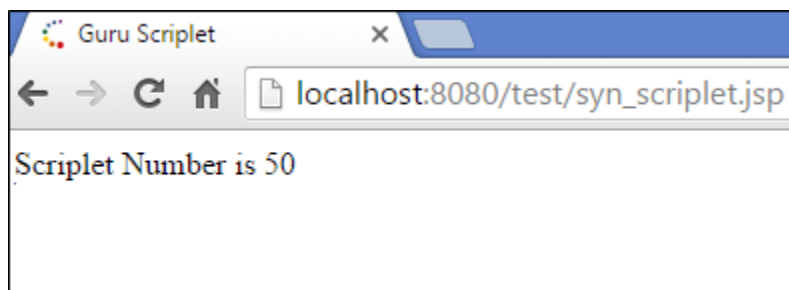
```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=ISO-8859-1">
```

```
<title>Guru Scriplet</title>
</head>
<body>
<% int num1=10;
    int num2=40;
    int num3 = num1+num2;
    out.println("Scriplet Number is " +num3);
%>
</body>
</html>
```

Explanation of the code:

Code Line 10-14: In the Scriptlet tags where we are taking two variables num1 and num2 . Third variable num3 is taken which adds up as num1 and num2.The output is num3.

When you execute the code, you get the following output:



Output:

The output for the Scriptlet Number is 50 which is addition of num1 and num2.

# JSP Expression

Expression tag evaluates the expression placed in it.
It accesses the data stored in stored application.
It allows create expressions like arithmetic and logical.
It produces scriptless JSP page.

Syntax:

```
<%= expression %>
```

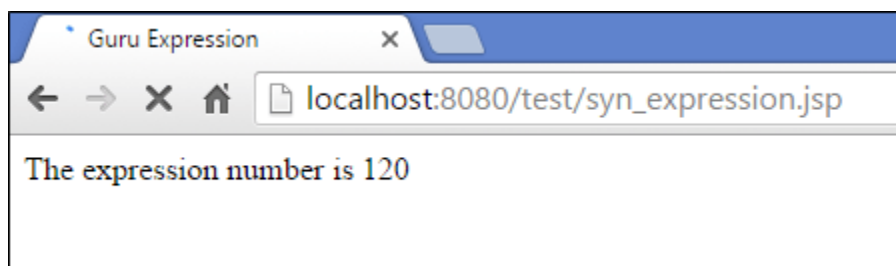Here the expression is the arithmetic or logical expression.

Example:

In this example, we are using expression tag

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=ISO-8859-1">
<title>Guru Expression</title>
</head>
<body>
<% out.println("The expression number is "); %>
<% int num1=10; int num2=10; int num3 = 20; %>
<%= num1*num2+num3 %>
</body>
</html>
```

Explanation of the code:

Code Line 12: Here we are using expression tags where we are using an expression by multiplying two numbers i.e. num1 and num 2 and then adding the third number i.e. num3.

When you execute the above code, you get the following output:



Output:

The expression number is 120 where we are multiplying two numbers num1 and num2 and adding that number with the third number.

# JSP Comments

Comments are the one when JSP container wants to ignore certain texts and statements.

When we want to hide certain content, then we can add that to the comments section.

Syntax:

```
<% -- JSP Comments %>
```

These tags are used to comment in JSP and ignored by the JSP container.

```
<!—comment –>
```

This is HTML comment which is ignored by browser

Example:
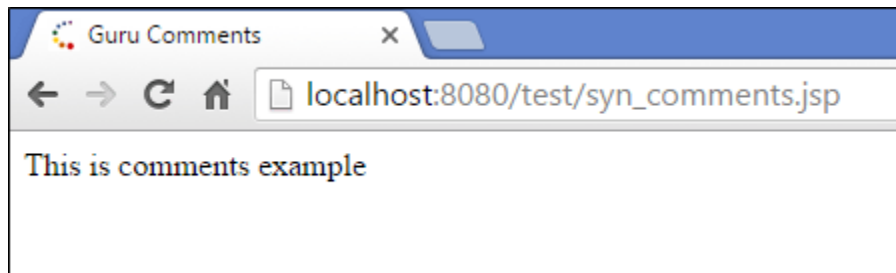
In this example, we are using JSP comments

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=ISO-8859-1">
<title>Guru Comments</title>
</head>
<body>
<%-- Guru Comments section --%>
<% out.println("This is comments example"); %>

</body>
</html>
```

Explanation of the code:

Code Line 10: Here we are adding JSP comments to the code to explain what code has. It is been ignored by the JSP container

When you execute the above code you get the following output:

Output:

We get the output that is printed in println method. Comments are ignored by container

## Creating a simple JSP Page

- A JSP page has an HTML body incorporated with Java code into it
- We are creating a simple JSP page which includes declarations, scriptlets, expressions, comments tags in it.

Example:

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=ISO-8859-1">
<title>Guru JSP Example</title>
</head>
<body>

<%-- This is a JSP example with scriplets, comments , expressions --%>
<% out.println("This is guru JSP Example"); %>
<% out.println("The number is "); %>
<%! int num12 = 12; int num32 = 12; %>
<%= num12*num32 %>
Today's date: <%= (new java.util.Date()).toLocaleString()%>
</body>
</html>
```

Explanation of the code:

Code Line 1: Here we are using directives like language, contentType and pageEncoding. Language is Java and content type is text/html with standard charset ISO 8859. Page encoding is standard charset.
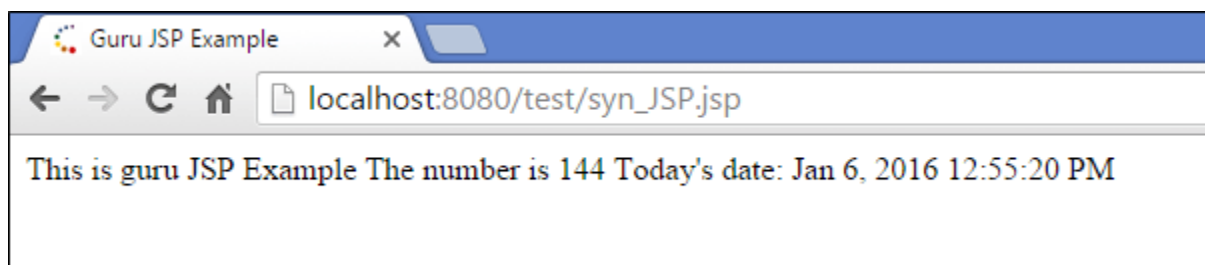
Code Line 11: Here we are using JSP comments to add comments to the JSP

Code Line 14: Here we are declaring variables num12 and num32 initializing with 12.

Code Line 15: Here we are using an expression where we are multiplying two numbers num12 and num32.

Code Line 16: Here we are fetching today's date using date object.

When you execute the above code, you get the following output
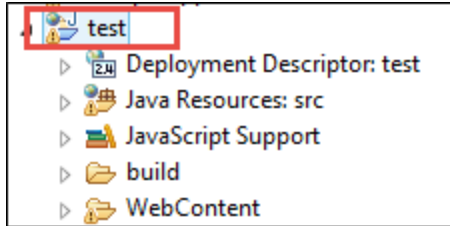


Output:

We are printing overhere,

- This is guru JSP example.
- The number is num12*num32 (12*12).
- Today's date is the current date

# How to run simple JSP Page
- JSP can be run on web servers or application servers.
- Here we will be using a webserver, and we can deploy it on the server enclosing it in a war application.
- We can create JSP in an application (war).

This is an application which has following directory structure, and the application has to be build.

This application has to be built, and the following message will appear after the build is successful:
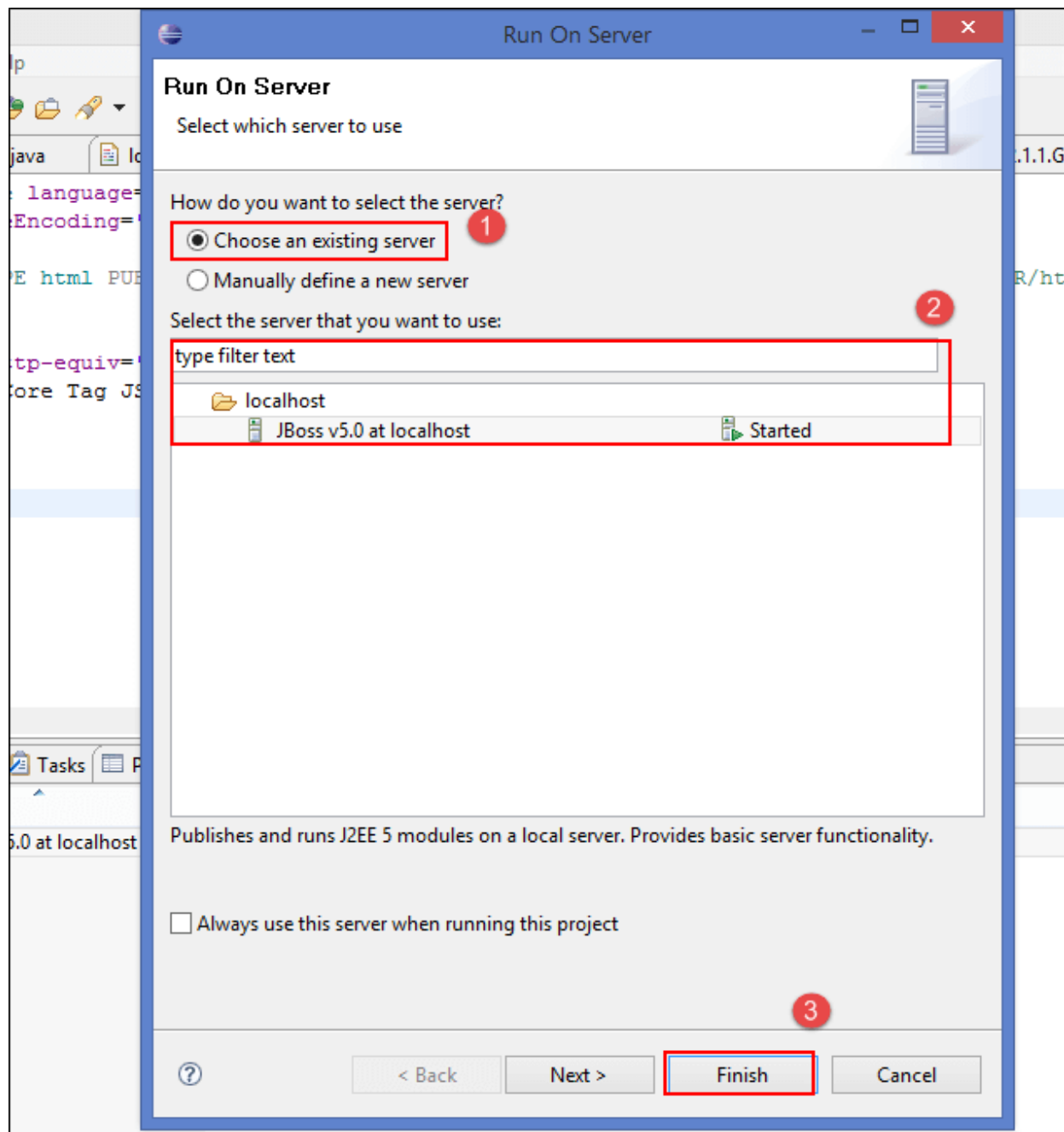
```
Buildfile: E:\Java\eclipse-jee-ganymede-SR2-win32\eclipse\plugins\org.eclipse.jst.server.generic.jboss_1.5.206.v20090115\buildfiles\jb
deploy.j2ee.web:
    [jar] Building jar: E:\Java\New folder\.metadata\.plugins\org.eclipse.wst.server.core\tmp0\test.war
    [move] Moving 1 file to E:\Java\jboss-5.0.0.GA\jboss-5.0.0.GA\server\default\deploy
BUILD SUCCESSFUL
Total time: 10 seconds
```

After the application is built then, the application has to be run on the server.

To run JSP on the webserver, right click on the project of the IDE (eclipse used in this case) and there are many options. Select the option of run on the server. It is shown in the screenshot below;

From the diagram, following points are explained:

1. There are two options either to choose a server or manually add the server to this application. In this case, we have already added JBoss server to the application hence, we select the existing server.
2. Once we select the server the server option is shown in point 2 which server we want to select. There can be multiple servers configured on this application. We can select one server from all those options
3. Once that option is selected click on finish button and application will run on that server.
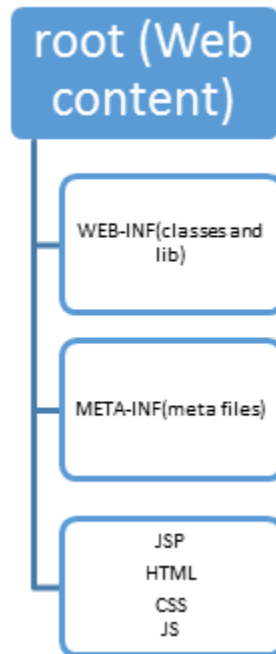
In the below screenshots, you can notice that our JSP program gets executed, and the test application is deployed in JBoss server marked in the red box.

# Directory Structure of JSP

In directory structure, there is a root folder which has folder WEB-INF, which has all configuration files and library files.
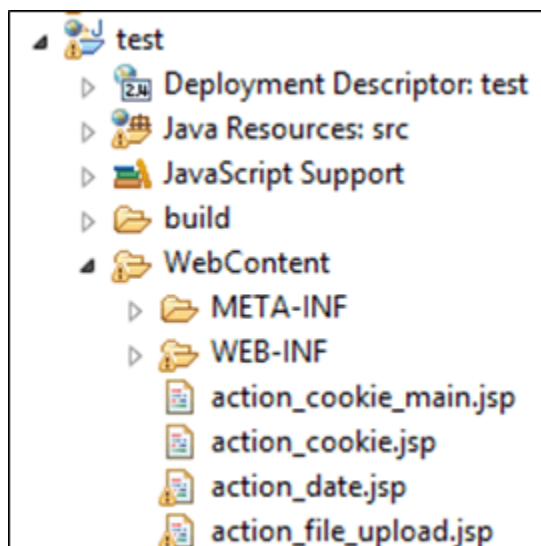
JSP files are outside WEB-INF folder



Directory structure of JSP

Example:

In this example there is test application which has folder structure has following:

Summary:

- In this article, we have learnt about syntactic elements like expression tags, Scriptlet tags which simplify code in JSP.
- Use of JSP comments.
- We have created a simple JSP page and made it run on the server.

*This text is presented here in case of changed or broken links, so that the learner may still have access to the information.*
guru99.com/jsp-elements.html