

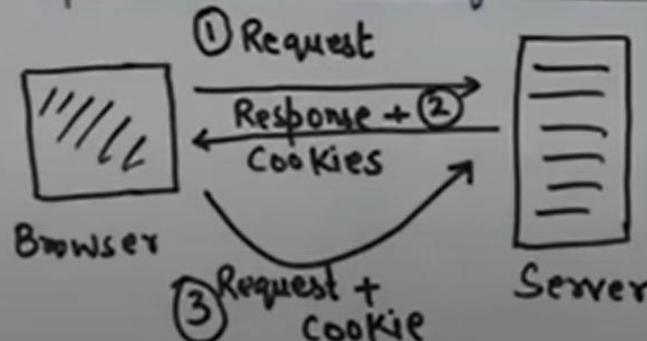


Press Esc to exit full screen

For Engineering Students of GGSIPU, UPTU and Other Universities, Colleges of India

## (Web Engineering Lecture.6) [Cookies in Java Servlet]

- Cookies are the text file stored on the client computer.
- Are used for various tracking purpose.

Steps involved in Identification using Cookies

Cookies

- Non-persistent Cookie (Valid for Single Session)
- Persistent Cookie (Valid for multiple Session)

Steps to Set Cookies

- Create cookie object

```
Cookie c = new Cookie("Key",
    "Value");
```

```
Cookie c1 = new Cookie("Name", "EEC");
```

- Set maximum Age

```
c1.setMaxAge(60 * 60 * 24);
```

- Send cookie into HTTP response Headers.

```
response.addCookie(c1);
```



8:40 / 13:02



# Easy Engineering Classes – Free YouTube Coaching

For Engineering Students of GGSIPU, UPTU and Other Universities, Colleges of India

## (Web Engineering- Servlet lecture.6)

### Example to Set cookie

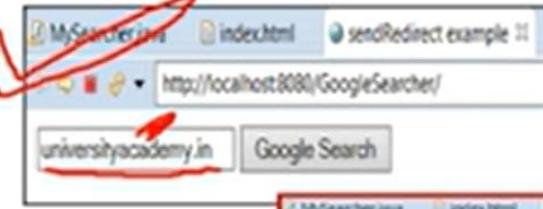
```
Cookie fname = new Cookie("f-name",  
    request.getParameter("first-name"));  
fname.setMaxAge(60*60*24);  
response.addCookie(fname);
```

### Example to Read cookie

```
Cookie CK[] = request.getCookies();  
out.println(CK[0].getName());  
out.println(CK[0].getValue());
```

O/P:- f-name

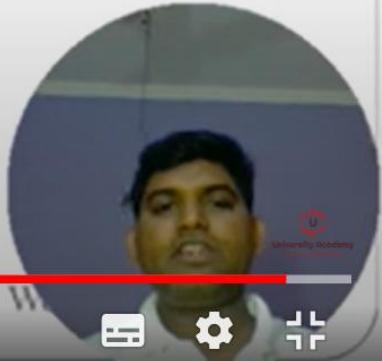
## Example



A screenshot of a Google search results page for "universityacademy.in". The search term "universityacademy.in" is highlighted in the search bar. The results show the official website of University Academy, which is a Non-Profit Organization providing e-Learning Portal. Other results include links to "Home - University Academy", "Compiler Design", "Download - University", and "Contact - University Academy". Red annotations include a large red box highlighting the search results and several red arrows pointing to specific links and text snippets.

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>sendRedirect example</title>
</head>
<body>
<form action="MySearcher">
<input type="text" name="name">
<input type="submit" value="Google Search">
</form>
</body>
</html>
```

```
import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
@WebServlet("/MySearcher")
public class MySearcher extends HttpServlet {
protected void doGet(HttpServletRequest request,
HttpServletResponse response) throws ServletException,
IOException {
String name=request.getParameter("name");
response.sendRedirect("https://www.google.co.in/#q="+name);
}
}
```



Answers Were Sorted Based On User's Feedback

What are the security issues in Servlets?..

[Answer / Bikash Khuntia](#)

- 1.Using Cookies is not secure for sending and receiving data  
bcz everyone can view and modify the data in cookies
- 2.Using doGet() for sending user information to server  
being visible in browser

Is This Answer Correct ? 53 Yes 3 No

What are the security issues in Servlets?..

[Answer / Akanksha Agarwal](#)

there are 2 type of applets-  
\*trusted  
\*untrusted

untrusted applets are constrained to operate in sand  
box they can not put the operations in user's machine which

- ◆ Servlets (804)
- ◆ Struts (701)
- ◆ JDBC (771)
- ◆ JMS (324)
- ◆ SunOne (10)
- ◆ J2EE (485)
- ◆ Weblogic (694)
- ◆ Websphere (573)
- ◆ Java Networking (49)
- ◆ Hibernate (1009)
- ◆ Spring Framework (1486)
- ◆ Java J2EE AllOther (116)

PASSCODE BIG  
HIT

Siddha  
Sejal Group  
In Mumbai



SIDDHA|SEJAL



Type here to search



33°C



ENG

10:00  
11-06-2022



## Difference between Servlet and CGI

### Servlet

Servlets are portable and efficient.

In Servlets, sharing of data is possible.

Servlets can directly communicate with the web server.

Servlets are less expensive than CGI.

Servlets can handle the cookies.

### CGI(Common Gateway Interface)

CGI is not portable

In CGI, sharing of data is not possible.

CGI cannot directly communicate with the web server.

CGI are more expensive than Servlets.

CGI cannot handle the cookies.

## CGI (Common Gateway Interface)

Before Servlets, CGI(Common Gateway Interface) programming was used to create web applications. Here's how a CGI program works :



Type here to search



10:02  
11-06-2022

# Easy Engineering Classes – Free YouTube Coaching

For Engineering Students of GGSIPU, UPTU and Other Universities, Colleges of India

## (Web Engineering - JSP Lecture.1) [Introduction, JSP Life Cycle]

### JSP (JavaServer Pages)

↳ Server-Side programming technology that enables the creation of dynamic, platform-independent method for building web-based applications.

→ Helps in building web pages that supports dynamic content.   
 JSP = HTML + Java

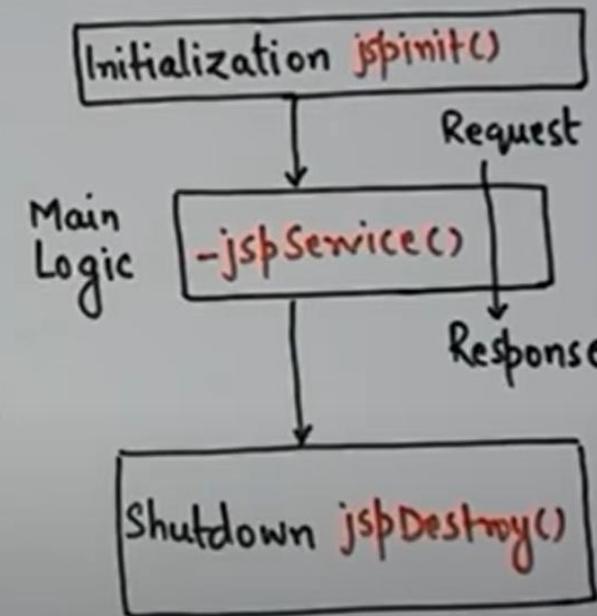
→ Inserts Java Code in HTML Page.

### JSP Life Cycle

Similar to the Servlet Life Cycle with an additional step.

↳ Compilation of JSP Page

into a Servlet Page.



For Engineering Students of GGSIPU, UPTU and Other Universities, Colleges of India

(Web Engineering - JSP Lecture.1)

### JSP life Cycle.

#### i) JSP Compilation

↳ i) Parsing the JSP

ii) Turning the JSP  
into the Servlet

iii) Compiling the Servlet

#### ii) JSP Initialization

↳ `jsplInit()` method, before  
any request is served.

#### iii) JSP Execution (Main Logic)

`jsp engine` → `-jspService() method`

`HttpServletRequest` ←      → `HttpServletResponse`

#### iv) JSP cleanup

`jspDestroy() method.`

↳ `destroy` meth



## Table of Contents

[Arrays in Java](#)

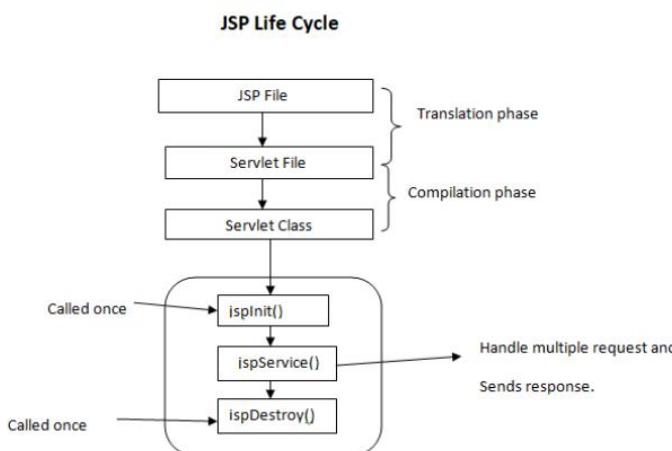
[Split\(\) String method in Java with examples](#)

[Arrays.sort\(\) in Java with examples](#)

[For-each loop in Java](#)

[Reverse a string in Java](#)

[Object Oriented Programming \(OOPS\) Concept in Java](#)



Following steps are involved in the JSP life cycle:

The perfect job is just around the corner

**hirist.com**  
the **tech job** specialist

**Start Your Coding Journey Now!**

[Login](#)

[Register](#)



Type here to search



33°C



12:02  
11-06-2022



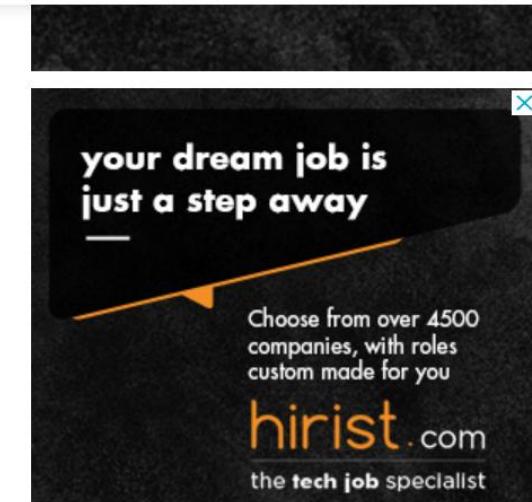


## Table of Contents

[Arrays in Java](#)[Split\(\) String method in Java with examples](#)[Arrays.sort\(\) in Java with examples](#)[For-each loop in Java](#)[Reverse a string in Java](#)[Object Oriented Programming \(OOPS\) Concept in Java](#)

Following steps are involved in the JSP life cycle:

- Translation of JSP page to Servlet
- Compilation of JSP page (Compilation of JSP into test.java)
- Classloading (test.java to test.class)
- Instantiation (Object of the generated Servlet is created)
- Initialization (`jsplInit()` method is invoked by the container)
- Request processing (`_jspService()` is invoked by the container)
- JSP Cleanup (`jspDestroy()` method is invoked by the container)



## WHAT'S NEW

Start Your Coding Journey Now!

[Login](#)[Register](#)

Type here to search



33°C

12:02  
11-06-2022



## Table of Contents

[Arrays in Java](#)[Split\(\) String method in Java with examples](#)[Arrays.sort\(\) in Java with examples](#)[For-each loop in Java](#)[Reverse a string in Java](#)[Object Oriented Programming \(OOPS\) Concept in Java](#)

### Translation of JSP page to Servlet :

This is the first step of the JSP life cycle. This translation phase deals with the Syntactic correctness of JSP. Here test.jsp file is translated to test.java.



Content Improvement League 2022

[View Details](#)

### Compilation of JSP page :

Here the generated java servlet file (test.java) is compiled to a class file (test.class).



Data Structures & Algorithms- Self Paced Course

[View Details](#)

### Classloading :

Servlet class which has been loaded from the JSP source is now loaded into the container.



### Instantiation :

Here an instance of the class is generated. The container manages one or more

Start Your Coding Journey Now!

[Login](#)[Register](#)

Type here to search



33°C

12:02  
11-06-2022



## Table of Contents

[Arrays in Java](#)[Split\(\) String method in Java with examples](#)[Arrays.sort\(\) in Java with examples](#)[For-each loop in Java](#)[Reverse a string in Java](#)[Object Oriented Programming \(OOPS\) Concept in Java](#)

Here an instance of the class is generated. The container manages one or more instances by providing responses to requests.

### Initialization :

`jsInit()` method is called only once during the life cycle immediately after the generation of Servlet instance from JSP.

### Request processing :

`_jspService()` method is used to serve the raised requests by JSP. It takes request and response objects as parameters. This method cannot be overridden.

### JSP Cleanup :

In order to remove the JSP from the use by the container or to destroy the method for servlets `jspDestroy()` method is used. This method is called once, if

**we have the best  
of both worlds**

Curated roles with  
salaries that match  
your expectations

Start Your Coding Journey Now!

[Login](#)[Register](#)

Type here to search



33°C

12:02  
11-06-2022



Category ▾

Search tutorials ...



Prime Packs

Courses

eBooks

Library

Q/A

Login

⊕ JSP - Form Processing

⊕ JSP - Writing Filters

⊕ JSP - Cookies Handling

⊕ JSP - Session Tracking

⊕ JSP - File Uploading

⊕ JSP - Handling Date

⊕ JSP - Page Redirect

⊕ JSP - Hits Counter

⊕ JSP - Auto Refresh

⊕ JSP - Sending Email

## Advanced JSP Tutorials

⊕ JSP - Standard Tag Library

⊕ JSP - Database Access

⊕ JSP - XML Data

⊕ JSP - Java Beans



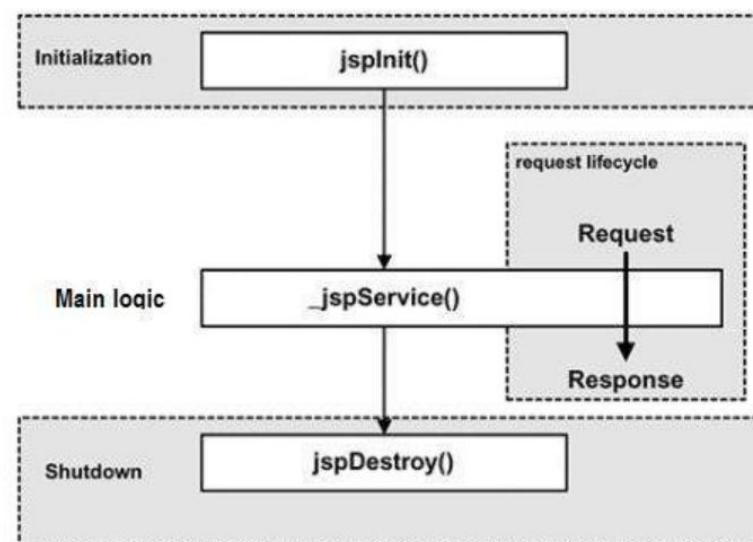
Type here to search



33°C

12:03  
11-06-2022

The four major phases of a JSP life cycle are very similar to the Servlet Life Cycle.  
The four phases have been described below –



## JSP Compilation

When a browser asks for a JSP, the JSP engine first checks to see whether it needs to compile the page. If the page has never been compiled, or if the JSP has been modified since it was last compiled, the JSP engine compiles the page.

The Containers and Cloud-Native Road Show  
Developer Track  
Containers and cloud-native workshop - learn like never before



## Register



June 28, 2022  
9:00 AM - 4:30 PM (GMT+02:00)  
2 Online

The Containers and Cloud-Native Road Show  
Developer Track  
Containers and cloud-native workshop - learn like never before



## Register





Category

Search tutorials ...

Prime Packs

Courses

eBooks

Library

Q/A

Login

JSP - Database Access

JSP - XML Data

JSP - Java Beans

JSP - Custom Tags

JSP - Expression Language

JSP - Exception Handling

JSP - Debugging

JSP - Security

JSP - Internationalization

## JSP Useful Resources

JSP - Questions and Answers

JSP - Quick Guide

JSP - Useful Resources

JSP - Discussion

## JSP Compilation

When a browser asks for a JSP, the JSP engine first checks to see whether it needs to compile the page. If the page has never been compiled, or if the JSP has been modified since it was last compiled, the JSP engine compiles the page.

The compilation process involves three steps –

- Parsing the JSP.
- Turning the JSP into a servlet.
- Compiling the servlet.

## JSP Initialization

When a container loads a JSP it invokes the **jspInit()** method before servicing any requests. If you need to perform JSP-specific initialization, override the **jspInit()** method –

```
public void jspInit(){  
    // Initialization code...  
}
```

Typically, initialization is performed only once and as with the servlet init method, you generally initialize database connections, open files, and create lookup tables in

The Containers and Cloud-Native Road Show  
Developer Track  
Containers and cloud-native workshop - learn like never before



Donald Software Engineer, Stakater

Stakater | Kubernetes... - Sponsored

Learn how using cloud-native development can change the way you...

Register



June 28, 2022  
9:00 AM - 4:30 PM  
(GMT+02:00)  
Online

The Containers and Cloud-Native Road Show  
Developer Track  
Containers and cloud-native workshop - learn like never before



Donald Software Engineer, Stakater

Stakater | Kubernetes... - Sponsored

Learn how using cloud-native development can change the way you...

Register



Type here to search



33°C



12:03  
11-06-2022





Category ▾

Search tutorials ...



Prime Packs

Courses

eBooks

Library

Q/A

Login

JSP - Internationalization

## JSP Useful Resources

JSP - Questions and Answers

JSP - Quick Guide

JSP - Useful Resources

JSP - Discussion

## Selected Reading

UPSC IAS Exams Notes

Developer's Best Practices

Questions and Answers

Effective Resume Writing

HR Interview Questions

Computer Glossary

Who is Who

When a container loads a JSP it invokes the **jsplInit()** method before servicing any requests. If you need to perform JSP-specific initialization, override the **jsplInit()** method –

```
public void jsplInit(){
    // Initialization code...
}
```

Typically, initialization is performed only once and as with the servlet init method, you generally initialize database connections, open files, and create lookup tables in the jsplInit method.

## JSP Execution

This phase of the JSP life cycle represents all interactions with requests until the JSP is destroyed.

Whenever a browser requests a JSP and the page has been loaded and initialized, the JSP engine invokes the **\_jspService()** method in the JSP.

The **\_jspService()** method takes an **HttpServletRequest** and an **HttpServletResponse** as its parameters as follows –

The Containers and Cloud-Native Road Show  
Developer Track  
Containers and cloud-native workshop - learn like never before



Donald Software Engineer, Stakater

support@stakater.com https://stakater.com

Stakater | Kubernetes... - Sponsored  
Learn how using cloud-native development can change the way you...

## Register

June 28, 2022  
9:00 AM - 4:30 PM (GMT+02:00)  
Online

The Containers and Cloud-Native Road Show  
Developer Track  
Containers and cloud-native workshop - learn like never before



Donald Software Engineer, Stakater

support@stakater.com https://stakater.com

Stakater | Kubernetes... - Sponsored  
Learn how using cloud-native development can change the way you...

## Register

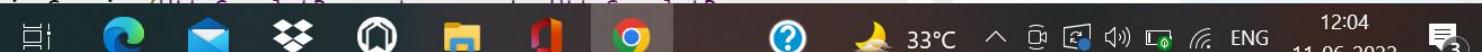


onestop devshop

i X



Type here to search

12:04  
11-06-2022  
ENG  
3



Category

Search tutorials ...



Prime Packs

Courses

eBooks

Library

Q/A

Login

- UPSC IAS Exams Notes
- Developer's Best Practices
- Questions and Answers
- Effective Resume Writing
- HR Interview Questions
- Computer Glossary
- Who is Who

## JSP Execution

This phase of the **JSP life cycle** represents all interactions with requests until the JSP is destroyed.

Whenever a browser requests a JSP and the page has been loaded and initialized, the JSP engine invokes the **\_jspService()** method in the JSP.

The **\_jspService()** method takes an **HttpServletRequest** and an **HttpServletResponse** as its parameters as follows –

```
void _jspService(HttpServletRequest request, HttpServletResponse response
    // Service handling code...
}
```

The **\_jspService()** method of a JSP is invoked on request basis. This is responsible for generating the response for that request and this method is also responsible for generating responses to all seven of the HTTP methods, i.e., **GET, POST, DELETE**, etc.

## JSP Cleanup

Have you heard of Scotiabank in Canada

Yes, but not considering

Yes, and considering

No

Have you heard of Scotiabank in Canada

Yes, and considering

Yes, but not considering

No

Connecting...



Type here to search



onestop  
DEVSHOP



33°C



12:04  
11-06-2022





Category

Search tutorials ...



Prime Packs

Courses

eBooks

Library

Q/A

Login

Have you heard of Scotiabank in Canada

Yes, but not considering

Yes, and considering

No

The **\_jspService()** method of a JSP is invoked on request basis. This is responsible for generating the response for that request and this method is also responsible for generating responses to all seven of the HTTP methods, i.e, **GET, POST, DELETE, etc.**

## JSP Cleanup

The destruction phase of the JSP life cycle represents when a JSP is being removed from use by a container.

The **jspDestroy()** method is the JSP equivalent of the destroy method for servlets. Override **jspDestroy** when you need to perform any cleanup, such as releasing database connections or closing open files.

The **jspDestroy()** method has the following form –

```
public void jspDestroy() {  
    // Your cleanup code goes here.  
}
```

Previous Page

Print Page

Next Page

Advertisements

Have you heard of Scotiabank in Canada

Yes, and considering

Yes, but not considering

No

onestop  
DEVSHOP

Have A Software Development Idea?



Let's Talk

Have you heard of Scotiabank in Canada

Connecting...



Type here to search



12:04  
11-06-2022

3

# Easy Engineering Classes – Free YouTube Coaching

For Engineering Students of GGSIPU, UPTU and Other Universities, Colleges of India

## [Web Engineering - JSP Lecture.2] [JSP Programming Introduction]

i) Scriptlet: Can contain any no. of JAVA language statements, Variable or method declarations, or expressions.

Syntax:- `<% Code %>`  
`</Html>`

(iii) JSP Expression: Contain Scripting language expression that is evaluated, converted to a String.

`<% = expression %>`  
Value returning is converted to string.

ii) JSP Declarations: declares one or more Variables or methods.

`<%! Declaration %>`

(iv) JSP Comments:

`<%-- JSP Comment --%>`

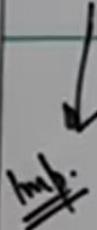
(v) JSP Directives: Affect overall Structure of Servlet class.

`<%@ directive attribute = "Value" %>`

# Easy Engineering Classes – Free YouTube Coaching

For Engineering Students of GGSIPU, UPTU and Other Universities, Colleges of India

## [Web Engineering- JSP Lecture.2]

Directive	Description	JSP Actions :- Use constructs in XML Syntax to control the behavior → Servlet engine.
<%@page...%>	page-dependent attributes. Scripting lang, Errors page etc.	<jsp:action-name attribute = "Value">
<%@include...%>	Includes a file during translation phase	jsp:include
<%@taglib...%>	Declares a tag lib, Containing Custom action.	Includes a file at the time is page is requested.
 <del>html:</del>	jsp: plugin, jsp: body : :	jsp:useBean
		Find and Instantiate JavaBean.
		jsp:forward
		Fowards the requests to a new page.
		jsp:text
		write template text.

## Easy Engineering Classes – Free YouTube Coaching

For Engineering Students of GGSIPU, UPTU and Other Universities, Colleges of India

## (Web Engineering - JSP Lecture.3) [First JSP Program]

Simple JSP Page

```
<html> <head>
<title>Simple JSP
</title></head>
<body> (a1.jsp)
<%
out.println("<h1>
JSP</h1>");
= insert java
%> Code in
</body>
</html>
```

index.html

JSP Page that reads data from HTML form

```
<html>
<head><title>....</title></head>
```

<body>

```
<form method="post" action="form.jsp">
<input type="text" name="f-name">
<input type="Submit" Value="Click" nam
</form> </body> </html>
```

//form.jsp

```
<body>
<% = request.getParameter("f-nam
</body>
```



Same

Settings

# Easy Engineering Classes – Free YouTube Coaching

For Engineering Students of GGSIPU, UPTU and Other Universities, Colleges of India

## [Web Engineering- JSP Lecture 4] [Custom Tags in JSP]

Custom tags are user-defined tags.

- ↳ Eliminates the need of scriptlet tag.

- Separating business logic from jsp

- Re-usability. Syntax:

<prefix:tagname attr1=“Value1” ... attrn=“Value n” />

Create Tag Handler class (inherit Tag Support class and override doStartTag() method)

↳ Create TLD file → info of tag and tag Handler class

↳ using of custom tags in

jsp file.

Thcl.java → Business Logic  
mytag.tld

↳ <tag>

<name>eec</name>  
<tag-class>com.eec.Thcl  
</tag-class>

</tag>

jsp page.

<%@ taglib uri="WEB-INF/  
mytag.tld" %>

prefix = "a".%>

Hello: <a: eec >

[Talk To Us](#)

**Estimate  
your  
Software  
project**

**onestop**

We Help You  
Estimate Your  
Software Project  
and Will Meet  
Your Needs

## JSP Has More Advantages than Servlet

JSP is considered to be better than servlet; there are many reasons why JSP is more beneficial than servlet:

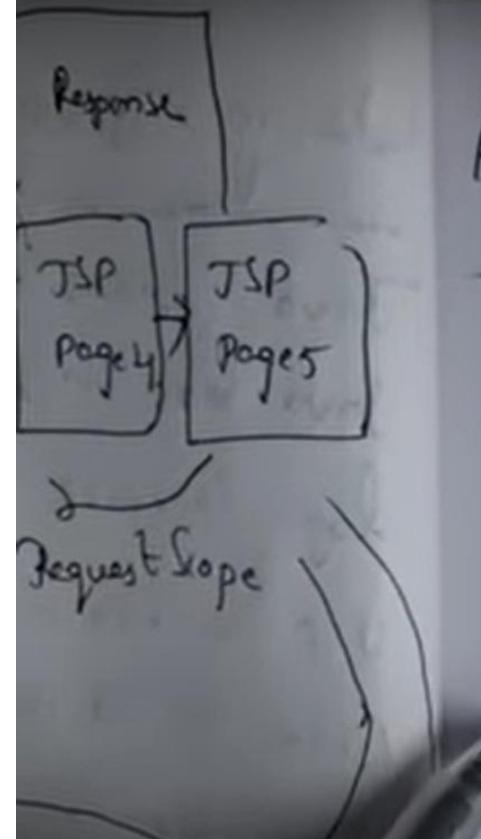
- JSP offers an efficient and more straightforward approach for coding dynamic web pages.
- JSP doesn't need to have additional files such as java class files, web.xml, etc.
- Web Container (or application server like tomcat) handles changes when changes are done in the JSP code, and it does not need recompilation.
- Direct access to JSP pages is possible, and hence web.xml mapping is not necessary as you have to do in servlets.

## Advantages of JSP

Here is the list of benefits that JSP provides:

- JSP codes are easy to write and maintain.
- Development speed becomes fast, and there is no need for recompilation and redeployment.
- It has high performance and scalability compared to other dynamic web development tools.
- JSP is constructed on Java technology, which makes it platform-independent.





## Anatomy of JSP / Components of JSP :-

JSP is built using components such as

- ① Scriptlets tag
  - ② Expressions tag
  - ③ Declarations tag
  - ④ Action tags
  - ⑤ Custom tags
  - ⑥ Directives
- Scripting elements → provides ability to insert java code inside the jsp.

## ① Scriptlet tag :-

→ In JSP, java code can be written inside the JSP page using the Scriptlet tag. A scriptlet tag is used to execute Java Source code in JSP.

Syntax:- <% java Source code %>

### Example of JSP Scriptlet tag :-

<html>

<body>

<% out.print ("JSP is equal to HTML + Java"); %>

Java Source code in JSP.

Syntax:- <% java Source code %>

Example of JSP Scriptlet tag:-

<html>

<body>

<% out.print ("JSP is equal to HTML + Java"); %>

</body>

</html>

② Expression tag:

The code placed within JSP expression tag is written to generate output as response. So you need ~~to write out~~ not to write `out.print()` to write data. It is mainly used to print the values of Variable or method.

Syntax: `<% = Statement %>`

Example:

Example of JSP expression tag, we are

Example:

in this example of jsp expression tag, we are  
simply displaying a welcome message

```
<html>
  <body>
    <% = "welcome to jsp" %>
  </body>
</html>
```

↓ no expression tag that prints the user name

</body>

</html>

O/P: welcome Sai

### ③ Declaration tag:-

⇒ The JSP Declaration tag is used to declare fields and methods.

⇒ the code written inside the JSP declaration tag is placed outside the service() method of Servlet. So it doesn't occupy memory for each request.

</body>  
</html>

o/p: welcome Sai

### ③ Declaration tag:-

⇒ The JSP Declaration tag is used to declare fields and methods.

⇒ the code written inside the JSP declaration tag is placed outside the service method of Servlet. So it doesn't occupy memory for each request.



Syntax-

`<% ! field or method declaration %>`

Difference Between JSP Scriptlet tag and Declaration tag

Example:

<html>

<body>

! int data = 50 ; %>

<% = "value of Variable is " + data %>

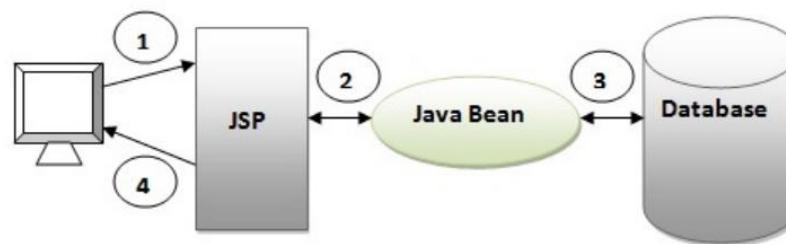
</body>

</html>

O/P: value of Variable is 50

[SessionAware](#)[ServletContextAware](#)[Struts2 with I18N](#)[Zero Configuration](#)[By convention](#)[By annotation](#)[Struts2 with Tiles2](#)[Hibernate with Struts2](#)[Spring with Struts2](#)[UI Tags](#)[DateTimePicker](#)[Iterator Tag](#)[Development](#)[Registration Example](#)[Login and Logout](#)[Fetch All Records](#)[Interview Questions](#)[Struts2 Quiz](#)[Struts Basics Quiz-1](#)[Struts Basics Quiz-2](#)[↑ SCROLL TO TOP](#)

presentation and business logic can be easily separated. You don't need to redeploy the application if JSP page is modified. JSP provides support to develop web application using JavaBean, custom tags and JSTL so that we can put the business logic separate from our JSP that will be easier to test and debug.



As you can see in the above figure, there is picture which show the flow of the model1 architecture.

1. Browser sends request for the JSP page
2. JSP accesses Java Bean and invokes business logic
3. Java Bean connects to the database and get/save data
4. Response is sent to the browser which is generated by JSP

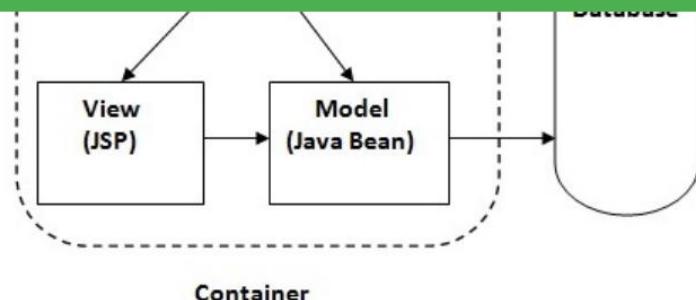
## Advantage of Model 1 Architecture

## Have A Software Development Idea?

[Let's Talk](#)

## Have A Software Development Idea?

[Let's Talk](#)

[Servlet](#) [JSP](#) [Struts2](#) [Mail API](#) [Hibernate](#) [Spring](#) [Android](#) [Design Pattern](#) [Quiz](#) [Projects](#) [Interview Q](#)[Let's Talk](#)

Simpler  
cloud.  
Happier  
devs.  
Better  
results.

[↑ SCROLL TO TOP](#)

## Advantage of Model 2 (MVC) Architecture

- **Navigation control is centralized** Now only controller contains the logic to determine the next page.
- **Easy to maintain**
- **Easy to extend**
- **Easy to test**
- **Better separation of concerns**

## Disadvantage of Model 2 (MVC) Architecture

- We need to write the controller code self. If we change the controller code, we need to recompile the class and redeploy the application.

## Have A Software Development Idea?



## JSP Application Design with MVC

. The key point of using MVC is to separate logic into three distinct units: the Model, the View, and the Controller.

50

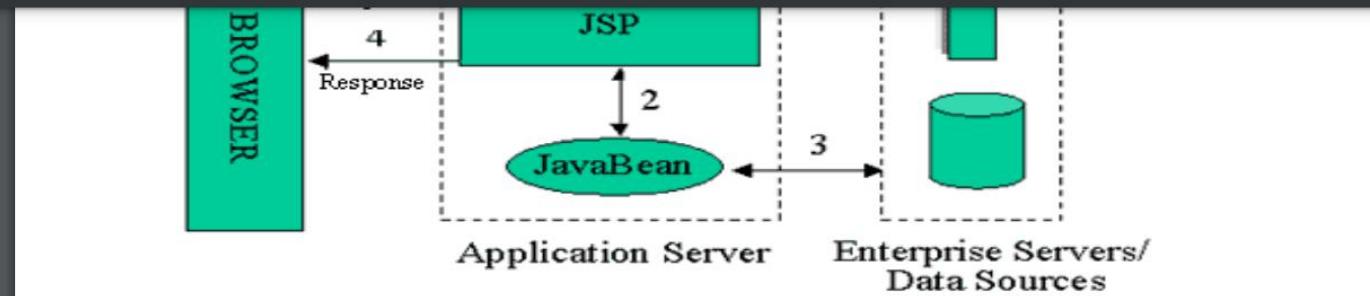
In a server application, we commonly classify the parts of the application as business logic, presentation, and request processing.

*Business logic* is the term used for the manipulation of an application's data, such as customer, product, and order information.

*Presentation* refers to how the application data is displayed to the user, for example, position, font, and size.

And finally, *request processing* is what ties the business logic and presentation parts together.

### JSP Model 1 Architecture



In MVC terms, the Model corresponds to business logic and data, the View to the presentation, and the Controller to the request processing.

Why use this design with JSP? The answer lies primarily in the first two elements. Remember that an application data structure and logic (the Model) is typically the most stable part of an application, while the presentation of that data (the View) changes fairly often. Just look at all the face-lifts many web sites go through to keep up with the latest fashion in web design. Yet, the data they present remains the same.

Another common example of why presentation should be separated from the business logic is that you may want to present the data in different languages or present different subsets of the data to internal and external users.

Access to the data through new types of devices, such as cell phones and personal digital assistants (PDAs), is the latest trend. Each client type requires its own presentation format. It should come as no surprise, then, that separating business logic from the presentation makes it easier to evolve an application as the requirements change; new presentation interfaces can be developed without touching the business logic.

## JSP Implicit Objects

1. request :- ~~ServletRequest, HttpServletRequest~~ ✓ <sup>Object</sup> ~~Object~~ <sup>String</sup> name = request.getParameter("t1")
2. response :- ~~ServletResponse, HttpServletResponse~~ ✓ <sup>Object</sup> ~~Object~~ response.sendRedirect("welcome.jsp")
3. out :- JspWriter out.print("Hello");
4. config :- ServletConfig <Context-param>  
<param-name> p1 </param-name>  
<param-value> Scott </param-value>
5. application :- ~~Context~~ <Context-param> ✓ <sup>Object</sup> ~~Object~~ <sup>String</sup> name = application.getInitParameter("p1")
6. ✓ page :- java.lang.Object
7. pageContext :- PageContext class
8. session :- HttpSession session.getAttribute("n")
9. exception :- java.lang.Throwable . exception



## Implicit JSP Objects

Variable Name	Java Type
Request	javax.servlet.http.HttpServletRequest
Response	javax.servlet.http.HttpServletResponse
PageContext	javax.servlet.jsp.PageContext
Session	javax.servlet.http.HttpSession
Application	javax.servlet.ServletContext
Out	javax.servlet.jsp.JspWriter
Config	javax.servlet.ServletConfig
Page	java.lang.Object



## CONDITIONAL PROCESSING

- ▶ In most web applications, we produce different output based on runtime conditions, such as the state of a bean or the value of a request header such as `UserAgent` (containing information about the type of client that is accessing the page).
- ▶ We can use JSP scripting elements to control which parts of the JSP page are sent to the browser, generating alternative outputs from the same JSP page.

## CONDITIONAL PROCESSING

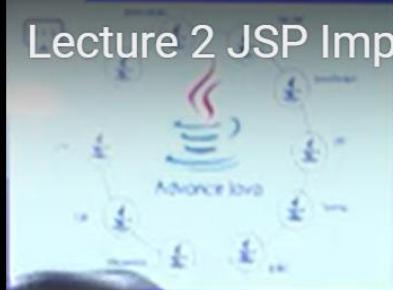


- ▶ If the outputs are completely different, use a separate JSP page for each alternative and passing control from one page to another.

### **1. USING JAVABEANS PROPERTIES**

- ▶ To use a bean's property value in a scripting element, call the accessor method directly. To illustrate this, let's use one of the properties of the java.util.Date class.

# CONDITIONAL PROCESSING



► Example: Browser-Dependent Page (browser.jsp)

```
<%@ page language="java" contentType="text/html" %>
<html>
<body bgcolor="white">
<% if (request.getHeader("User-Agent").indexOf("MSIE") != -1) { %>
    You're using Internet Explorer.
<% }
} else
```

# CONDITIONAL PROCESSING



### 3. WORKING WITH ARRAYS

- ▶ Another common use of scriptlets is to loop over an array. In the following example, the user pick a number of items from a group of checkboxes, and then use scriptlets to display all the choices.



## SHARING SESSION AND APPLICATION DATA

- ▶ HTTP is a stateless, request-response protocol. This means that browser sends a request for a web resource & web server processes the request & returns a response.
- ▶ The server then forgets about operations performed along with data. So when the same browser sends a new request, the web server has no idea that this request is related to the previous one.



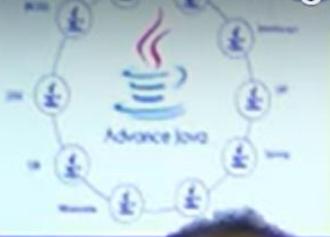


## SHARING SESSION AND APPLICATION DATA

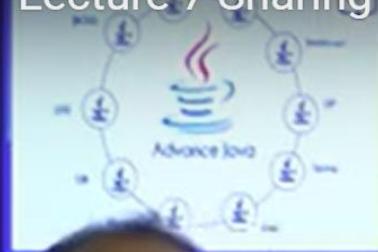


- ▶ The way to solve this problem is to let the server send a piece of information to the browser that the browser then includes in all subsequent requests. This piece of information called as **Session**.
- ▶ It is used by the server to recognize a set of requests from same browser as related: in other words, as part of the same session.

## SHARING SESSION AND APPLICATION DATA



- ▶ A session starts when the browser makes the first request for a JSP page in a particular application.
- ▶ The session can be ended explicitly by the application or JSP container can end it after a period of user inactivity.
- ▶ Information can therefore be saved on the server while processing one request & accessed later when another request is processed.
- ▶ The server uses the session ID to associate the requests with a session object, a temporary memory storage area where servlets and JSP pages can store information.



## SHARING SESSION AND APPLICATION DATA



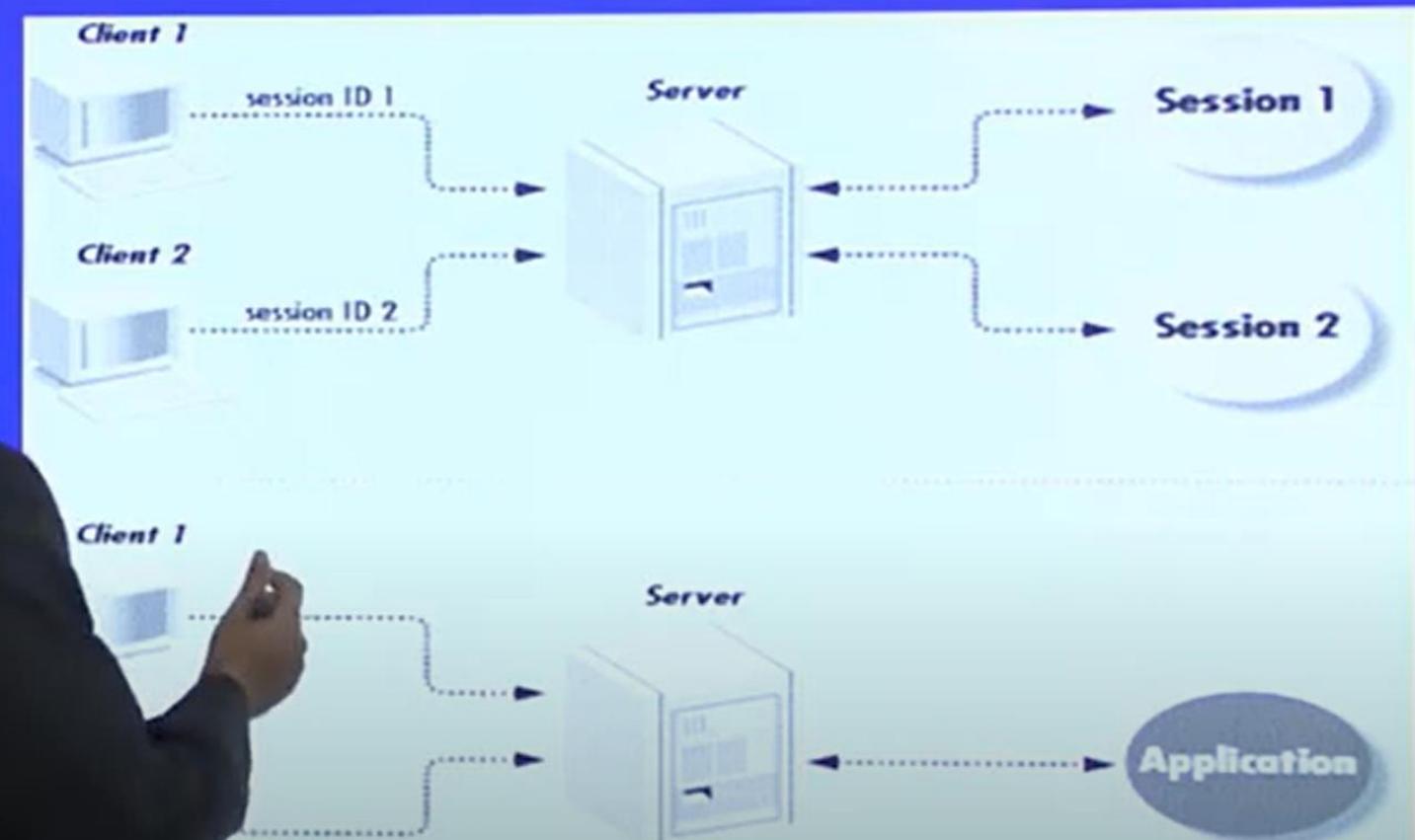
- ▶ The session ID can be transferred between the server and browser in a few different ways.
- ▶ The Servlet 2.2 API identifies three methods: using cookies, using encoded URLs & using the session mechanism built into the Secure Socket Layer (SSL).
- ▶ SSL-based session tracking is currently not supported by any of the major servlet containers, but all of them support the cookie and URL rewriting techniques.

## SHARING SESSION AND APPLICATION DATA



- ▶ JSP hides most of the details about how the session ID is transferred and how the session object is created and accessed, providing with the session scope to handle session data at a convenient level of abstraction.
- ▶ Information saved in the session scope is available to all pages requested by the same browser during the lifetime of the session.
- ▶ The following diagram shows how the server provides access two scopes for different clients.

# SHARING SESSION AND APPLICATION DATA





## MEMORY USAGE CONSIDERATIONS

### MEMORY USAGE CONSIDERATIONS

- ▶ It's easy to calculate how much memory is used for application objects since we have full control over the number of objects.
- ▶ But the total number of objects in the session scope depends on the number of concurrent sessions, so in addition to the size of each object we also need to know how many concurrent users we have and how long a session lasts.

## MEMORY USAGE CONSIDERATIONS



- ▶ If we store larger objects in each session, for instance the results of a database search, with an average of 10 KB per active session that corresponds to roughly 40 MB for 4,000 sessions.
- ▶ Here are some things we can do to keep the memory requirements under control:
  - Place only those objects that really need to be unique for each session in the session scope
  - Set the timeout period for sessions to a lower value than the default.



## MEMORY USAGE CONSIDERATIONS



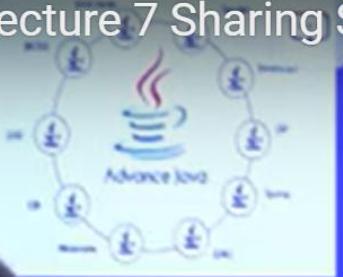
- ▶ Consider simple example of CartBean. It stores only references to product. Consider memory required for storing each product is 24 bytes.
- ▶ If this site has a modest number of customers we may have 10 users shopping per hour.
- ▶ The default timeout for a session is 30 minutes, have 10 active users & 10 sessions that are not active but have not timed out yet.

This gives a total of 20 sessions times 56 bytes per session, a total of 1,120 bytes.



9:36 / 13:57





### MEMORY USAGE CONSIDERATIONS



- Provide a way to end the session explicitly. A good example is a logout function. Another possibility is to invalidate the session when something is completed with `session.invalidate()`