



InterviewBit

JSP Interview Questions



To view the live version of the page, [click here.](#)

© Copyright by Interviewbit

Contents

JSP Interview Questions for Freshers

1. What is JSP?
2. How does JSP work?
3. How does JSP Initialization take place?
4. What is the use of JSP?
5. What are some of the advantages of using JSP?
6. What is Java Server Template Engines?
7. What are Servlets?
8. Explain the Life Cycle of a servlet.
9. What are the types of elements with Java Server Pages (JSP)?
10. What is the difference between JSP and Javascript?
11. What is JSP Expression Language (EL)?
12. What are JSP Operators?
13. Explain the JSP for loop.
14. Explain the JSP while loop.

JSP Interview Questions for Experienced

15. What are Implicit JSP Objects?
16. What do you mean by JavaBeans?
17. What is J2EE?
18. What is JSTL?

JSP Interview Questions for Experienced (.....Continued)

19. What are JSTL Core tags used for?
20. Which methods are used for reading form data using JSP?
21. What is an Exception Object?
22. How does JSP processing take place?
23. Explain the anatomy of a JSP page?
24. What are the various action tags used in JSP?
25. What is the JSP Scriptlet?
26. What is MVC in JSP?
27. What is a JSP Declaration?

Conclusion

28. Conclusion

Let's get Started

Introduction to Java Server Pages (JSP):

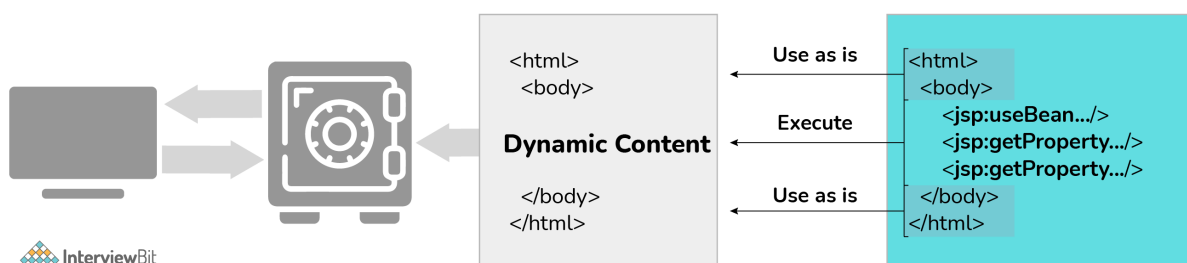
Java Server Pages (JSP) enables the creation of dynamic and a platform-independent way for building web-based applications. It is a server-side programming technology. JSP is a technology that is an integral part of Java EE, a complete platform for enterprise-class applications. This means that JSP can be used from the simplest applications to the most complex and demanding ones.

JavaServer Pages (JSP) often serve the same purpose as programs implemented using the Common Gateway Interface (CGI). However, in addition, JSP offers several advantages as compared to CGI.

JSP Interview Questions for Freshers

1. What is JSP?

JSP stands for Java Server Pages. This technology is used to create dynamic web pages in the form of [HyperText Markup Language](#) (HTML). They have embedded Java code pieces in them. They are an extension to the Servlet Technology and generate Servlet from a page. It is common to use both servlets and JSP pages in the same web apps.



2. How does JSP work?

The JSP container has a special servlet called the page compiler. All HTTP requests with URLs that match the .jsp file extension are forwarded to this page compiler by the configuration of the servlet container. The servlet container is turned into a JSP container with this page compiler. When a .jsp page is first called, the page compiler parses and compiles the .jsp page into a servlet class. The JSP servlet class is loaded into memory on the successful compilation. For the subsequent calls, the servlet class for that .jsp page is already in memory. Hence, the page compiler servlet will always compare the timestamp of the JSP servlet with the JSP page. If the .jsp page is more current, recompilation is necessary. With this process, once deployed, JSP pages only go through the time-consuming compilation process once.

3. How does JSP Initialization take place?

When a container loads a JSP, it invokes the `jspInit()` method before servicing any requests.

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

4. What is the use of JSP?

Earlier, Common Gateway Interface (CGI) was the only tool for developing dynamic web content and was not very efficient. The web server has to create a new operating system process, load an interpreter and a script, execute the script, and then tear it all down again, for every request that comes in. This is taxing for the server and doesn't scale well when the number of traffic increases.

Alternatives such as ISAPI from Microsoft, and Java Servlets from Sun Microsystems, offer better performance and scalability. However, they generate web pages by embedding HTML directly in programming language code. JavaServer Pages (JSP) changes all of that.

5. What are some of the advantages of using JSP?

- Better performance and quality as JSP is a specification and not a product.
- JSP pages can be used in combination with servlets.
- JSP is an integral part of J2EE, a complete platform for Enterprise-class applications.
- JSP supports both scripting and element-based dynamic content.

6. What is Java Server Template Engines?

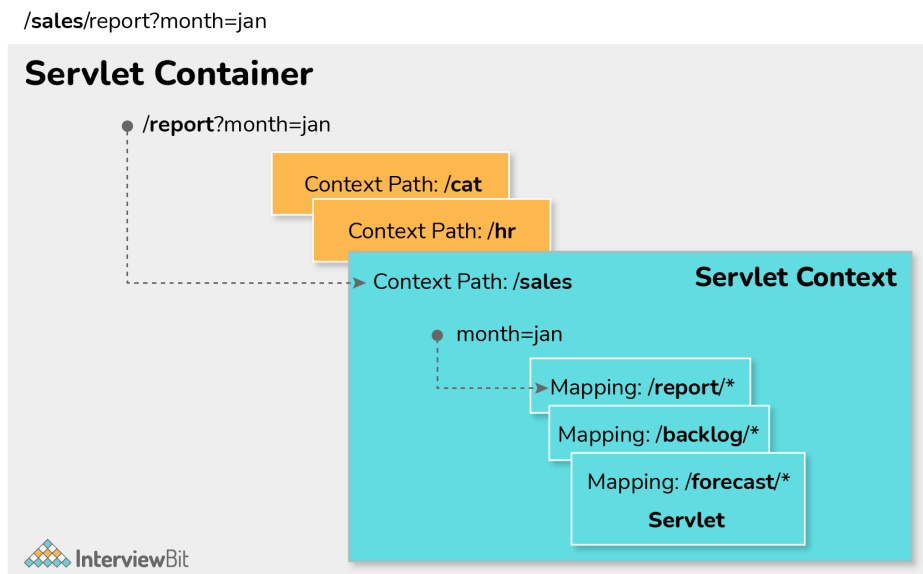
A Java servlet template engine is a technology for separating presentation from processing. Template engines have been developed as open-source products to help get HTML out of the servlets. These template engines are intended to be used together with pure code components (servlets) and use only web pages with scripting code for the presentation part.

Two popular template engines are WebMacro (<http://www.webmacro.org>) and FreeMarker (<http://freemarker.sourceforge.net>).

7. What are Servlets?

JSP pages are often combined with servlets in the same application. The JSP specification is based on the Java servlet specification. Simply put, a servlet is a piece of code that adds new functionality to a web server, just like CGI and proprietary server extensions such as NSAPI and ISAPI. Compared to other technologies, servlets have a number of advantages:

- Platform and vendor independence
- Integration
- Efficiency
- Scalability
- Robustness and security



8. Explain the Life Cycle of a servlet.

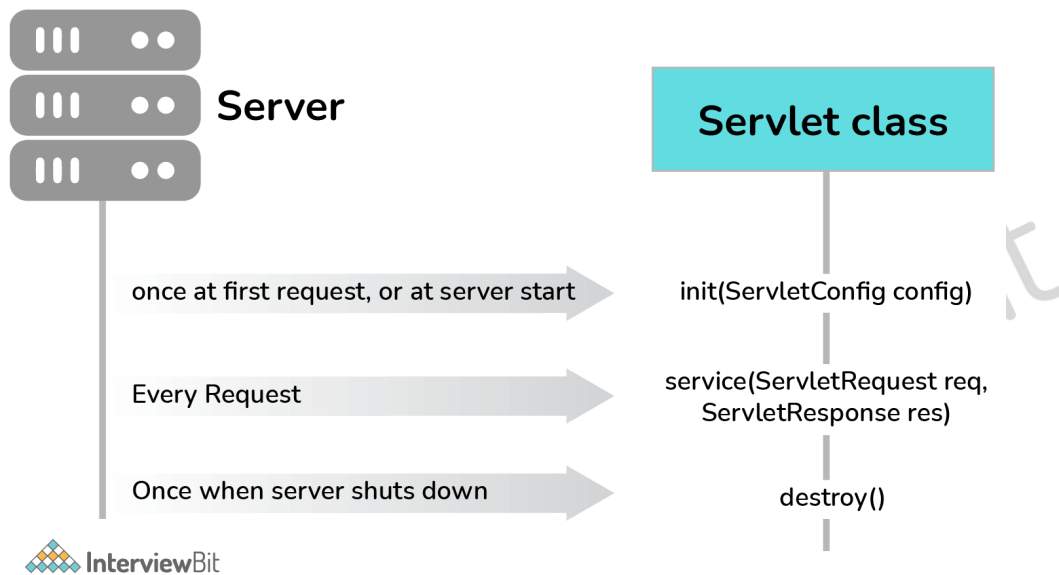
A Java class that uses the Servlet Application Programming Interface (API) is a Servlet. The Servlet API consists of many classes and interfaces that define some methods. These methods make it possible to process HTTP requests in a web server-independent manner.

A servlet is loaded when a web server receives a request that should be handled by it. Once a servlet has been loaded, the same servlet instance (object) is called to process succeeding requests. Eventually, the webserver needs to shut down the servlet, typically when the web server itself is shut down.

The 3 life cycle methods are:

- `public void init(ServletConfig config)`
- `public void service(ServletRequest req, ServletResponse res)`
- `public void destroy()`

These methods define the interactions between the web server and the servlet.



9. What are the types of elements with Java Server Pages (JSP)?

The three types of elements with Java Server Pages (JSP) are directive, action, and scripting elements.

Following are the Directive Elements:

Element	Description
<code><%@ page ... %></code>	Defines page-dependent attributes, such as scripting language, error page, and buffering requirements.
<code><%@ include ... %></code>	Includes a file during the translation phase.
<code><%@ taglib ... %></code>	Declares a tag library, containing custom actions, used on the page.



The Action elements are:

Element	Description
<code><jsp:useBean></code>	This is for making the JavaBeans component available on a page.
<code><jsp:getProperty></code>	This is used to get a property value from a JavaBeans component and to add it to the response.
<code><jsp:setProperty></code>	This is used to set a value for the JavaBeans property.
<code><jsp:include></code>	This includes the response from a servlet or JSP page during the request processing phase.
<code><jsp:forward></code>	This is used to forward the processing of a request to a JSP page or servlet.
<code><jsp:param></code>	This is used for adding a parameter value to a request given to another servlet or JSP page by using <code><jsp:include></code> or <code><jsp:forward></code>
<code><jsp:plugin></code>	This is used to generate HTML that contains the proper client browser-dependent elements which are used to execute an Applet with Java Plugin software.

And lastly, the Scripting elements are:

Element	Description
<code><% ... %></code>	Scriptlet used to embed scripting code.
<code><%= ... %></code>	Expression, used to embed Java expressions when the result shall be added to the response. Also used as runtime action attribute values.
<code><%! ... %></code>	Declaration used to declare instance variables and methods in the JSP page implementation class.

10. What is the difference between JSP and Javascript?

JSP is a server-side scripting language as it runs on the server. Whereas, JavaScript runs on the client. Commonly, JSP is more used to change the content of a webpage, and JavaScript for the presentation. Both are quite commonly used on the same page.

11. What is JSP Expression Language (EL)?

Expression Language (EL) was introduced in JSP 2.0. It is a mechanism that simplifies the accessibility of the data stored in Javabeen components and other objects like request, session, and application, etc. There are many operators in JSP that are used in EL like arithmetic and logical operators to perform an expression.

12. What are JSP Operators?

JSP Operators support most of the arithmetic and logical operators that are supported by java within expression language (EL) tags.

Following are the frequently used jsp operators:

.	Access a bean property or Map entry.
[]	Access an array or List element.
()	Group a subexpression to change the evaluation order.
+	Addition
-	Subtraction or negation of a value
*	Multiplication
/ or div	Division
% or mod	Modulo (remainder)
== or eq	Test for equality
!= or ne	Test for inequality
< or lt	Test for less than
> or gt	Test for greater than
<= or le	Test for less than or equal
>= or ge	Test for greater than or equal
&& or and	Test for logical AND
or or	Test for logical OR
! or not	Unary Boolean complement

13. Explain the JSP for loop.

The JSP For loop is used for iterating the elements for a certain condition, and it has the following three parameters:

- The variable counter is initialized
- Condition till the loop has to be executed
- The counter has to be incremented

The for loop syntax is as follows:

```
for(int i=0;i<n;i++)
{
    //block of statements
}
```

14. Explain the JSP while loop.

The JSP While loop is used to iterate the elements where it has one parameter of the condition.

Syntax of While loop:

```
While(i<n)
{
    //Block of statements
}
```

JSP Interview Questions for Experienced

15. What are Implicit JSP Objects?

Variable Name	Java Type	Description
request	<code>javax.servlet.http.HttpServletRequest</code>	The request object is used to request information like a parameter, header information, server name, etc.
response	<code>javax.servlet.http.HttpServletResponse</code>	The response is an instance of a class that represents a response that can be given to the client.
pageContext	<code>javax.servlet.jsp.PageContext</code>	This is used to get, set, and remove attributes from a particular scope.
session	<code>javax.servlet.http.HttpSession</code>	This is used to get, set, and remove attributes from a session scope and also used to get session ID.

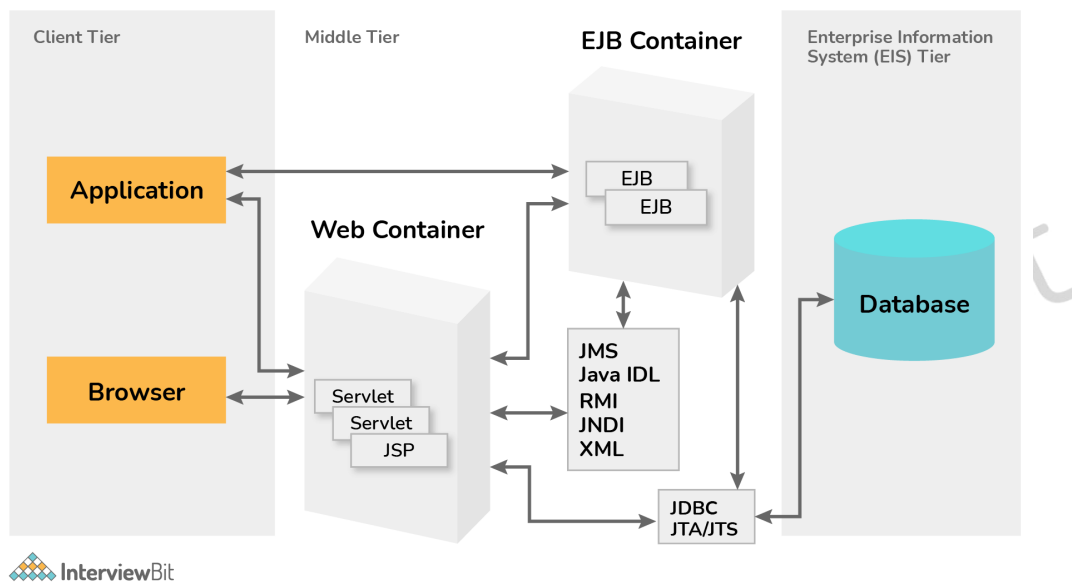
16. What do you mean by JavaBeans?

JavaBeans component is a Java class that complies with certain coding conventions. JSP elements often work with JavaBeans. For information that describes application entities, JavaBeans are typically used as containers.

17. What is J2EE?

J2EE is basically a compilation of different Java APIs that have previously been offered as separate packages. J2EE Blueprints describe how they can all be combined. J2EE vendors can use a test suite to test their products for compatibility. J2EE comprises the following enterprise-specific APIs:

- JavaServer Pages (JSP)
- Java Servlets
- Enterprise JavaBeans (EJB)
- Java Database Connection (JDBC)
- Java Transaction API (JTA) and Java Transaction Service (JTS)
- Java Naming and Directory Interface (JNDI)
- Java Message Service (JMS)
- Java IDL and Remote Method Invocation (RMI)
- Java XML



18. What is JSTL?

JSTL stands for Java server pages standard tag library. It is a collection of custom JSP tag libraries that provide common functionality for web development.

Following are some of the properties of JSTL:

- Code is Neat and Clean.
- Being a Standard Tag, it provides a rich layer of the portable functionality of JSP pages.
- It has Automatic Javabeans Introspection Support. The JSTL Expression language handles JavaBean code very easily. We don't need to downcast the objects, which have been retrieved as scoped attributes.
- Easier for humans to read and easier for computers to understand.

19. What are JSTL Core tags used for?

The JSTL Core tags are used for the following purposes:

- Iteration
- Conditional logic
- Catch exception
- URL forward
- Redirect, etc.

Following is the syntax to include a tag library:

```
<%@ taglib prefix="c" uri=http://java.sun.com/jsp/jstl/core%>
```

20. Which methods are used for reading form data using JSP?

JSP is used to handle the form data parsing automatically. It does so by using the following methods depending on the situation:

- **getParameter()** – To get the value of a form parameter, call the `request.getParameter()` method.
- **getParameterValues()** – If a parameter appears more than once and it returns multiple values, call this method.
- **getParameterNames()** – This method is used if, in the current request, you want a complete list of all parameters.
- **getInputStream()** – This method is used for reading binary data streams from the client.

21. What is an Exception Object?

The exception object is an instance of a subclass of `Throwable` (e.g., `java.lang.NullPointerException`). It is only available on the error pages. The following table lists out the important methods available in the `Throwable` class:

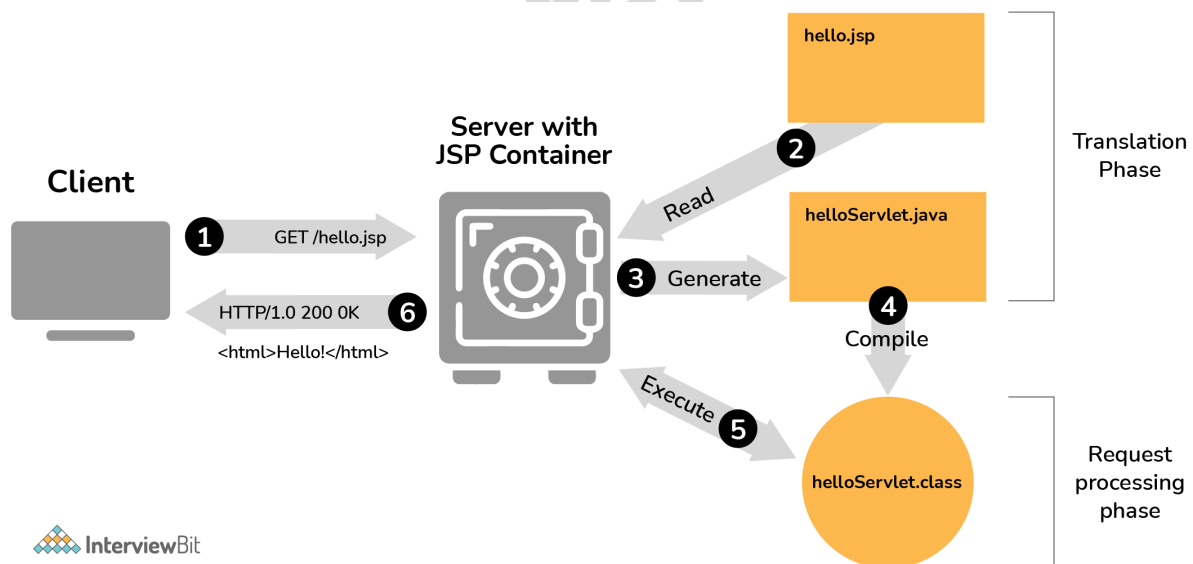
1	public String getMessage() Returns a detailed message about the exception that has occurred. This message is initialized in the Throwable constructor.
2	public Throwable getCause() Returns the cause of the exception as represented by a Throwable object.
3	public String toString() Returns the name of the class concatenated with the result of getMessage().
4	public void printStackTrace() Prints the result of toString() along with the stack trace to System.err, the error output stream.
5	public StackTraceElement [] getStackTrace() Returns an array containing each element on the stack trace. The element at index 0 represents the top of the call stack, and the last element in the array represents the method at the bottom of the call stack.
6	public Throwable fillInStackTrace() Fills the stack trace of this Throwable object with the current stack trace, adding to any previous information in the stack trace.

22. How does JSP processing take place?

The JSP page is turned into a servlet for all the JSP elements to be processed by the server. Then the servlet is executed. The servlet container and the JSP container—are often combined into one package under the name “web container”.

In the translation phase, the JSP container is responsible for converting the JSP page into a servlet and compiling the servlet. This is used to automatically initiate the translation phase for a page when the first request for the page is received.

In the “request processing” phase, the JSP container is also responsible for invoking the JSP page implementation class to process each request and generate the response.



23. Explain the anatomy of a JSP page?

Different JSP elements are used for generating the parts of the page that differ for each request. A JSP page is a regular web page with different JSP elements. The three types of elements with JavaServer Pages are directive, action, and scripting elements. JSP elements are often used to work with JavaBeans.

The elements of the page that are not JSP elements are simply called the “template text”. The template text is commonly HTML, but it could also be any other text.

When a page request of JSP is processed, the template text and the dynamic content generated by the JSP elements are merged, and the result is sent as the response to the browser.

<code><%@ page language="java" contentType="text/html" %></code>	JSP element
<code><html></code> <code><body bgcolor="white"></code>	template text
<code><jsp:useBean</code> <code>id="userInfo"</code> <code>class="com.ora.jsp.beans.userinfo.UserInfoBean"></code> <code><jsp:setProperty name="userInfo" property="*" /></code> <code></jsp:useBean></code>	JSP element
The following information was saved: <code></code> <code>User Name:</code>	template text
<code><jsp:getProperty name="userInfo"</code> <code>property="userName" /></code>	JSP element
<code>Email Address:</code>	template text
<code><jsp:getProperty name="userInfo"</code> <code>property="emailAddr" /></code>	JSP element
<code></code> <code></body></code> <code></html></code>	template text

24. What are the various action tags used in JSP?

Various action tags used in JSP are as follows:

- **jsp:forward:** This action tag forwards the request and response to another resource.
- **jsp:include:** This action tag is used to include another resource.
- **jsp:useBean:** This action tag is used to create and locates bean objects.
- **jsp:setProperty:** This action tag is used to set the value of the property of the bean.
- **jsp:getProperty:** This action tag is used to print the value of the property of the bean.
- **jsp:plugin:** This action tag is used to embed another component such as the applet.
- **jsp:param:** This action tag is used to set the parameter value. It is used in forward and includes mostly.
- **jsp:fallback:** This action tag can be used to print the message if the plugin is working.

25. What is the JSP Scriptlet?

The JSP Scriptlet tag allows you to write Java code into a JSP file. The JSP container moves statements in the `_jspService()` method while generating servlets from JSP.

For each request of the client, the service method of the JSP gets invoked hence the code inside the Scriptlet executes for every request.

In Scriptlet, a java code is executed every time the JSP is invoked.

Syntax of Scriptlet tag:

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

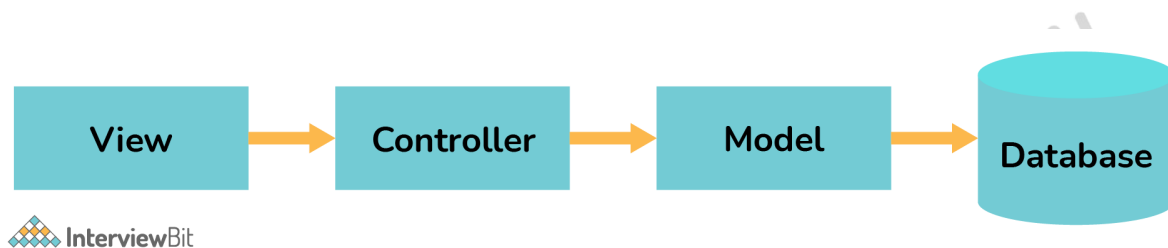
Here `<% %>` tags are scriptlet tags and within it, we can place the java code.

26. What is MVC in JSP?

In MVC,

- **M** stands for Model
- **V** stands for View
- **C** stands for the controller.

It is an architecture that separates business logic, presentation, and data. In this, the flow starts from the view layer, where the request is raised and processed in the controller layer. This is then sent to the model layer to insert data and get back the success or failure message.



27. What is a JSP Declaration?

The tags used in declaring variables are called JSP Declaration tags. These are used in declaring functions and variables. They are enclosed in `<%!%>` tag. Following is the syntax for JSP Declaration:

```
<%@page contentType="text/html" %>
<html>
<body>
<%!
int a=0;
private int getCount(){
a++;
return a;
}
%>
<p>Values of a are:</p>
<p><%=getCount()%></p>
</body>
</html>
```

Conclusion

28. Conclusion

The Java 2 Enterprise Edition (J2EE) takes the task of building an Internet presence and transforms it to the point where developers can use Java to efficiently create multi-tier, server-side applications. In late 1999, Sun Microsystems added a new element to the collection of Enterprise Java tools, called the JavaServer Pages (JSP). The JSP, built on top of Java servlets, is designed to increase the efficiency in which programmers, and even nonprogrammers, can create web content.

JavaServer Pages helps in developing web pages that include dynamic content. A JSP page can change its content based on any number of variable items. A JSP page not only contains standard markup language elements like a regular web page but also contains special JSP elements that allow the server to insert dynamic content in the page. This combination of standard elements and custom elements allows for the creation of powerful web apps.

References:

- JavaServer Pages, 3rd Edition, O'Reilly.
- Web Development with JavaServer Pages, by Duane and Mark.

Links to More Interview Questions

[C Interview Questions](#)

[Php Interview Questions](#)

[C Sharp Interview Questions](#)

[Web Api Interview Questions](#)

[Hibernate Interview Questions](#)

[Node Js Interview Questions](#)

[Cpp Interview Questions](#)

[Oops Interview Questions](#)

[Devops Interview Questions](#)

[Machine Learning Interview Questions](#)

[Docker Interview Questions](#)

[Mysql Interview Questions](#)

[Css Interview Questions](#)

[Laravel Interview Questions](#)

[Asp Net Interview Questions](#)

[Django Interview Questions](#)

[Dot Net Interview Questions](#)

[Kubernetes Interview Questions](#)

[Operating System Interview Questions](#)

[React Native Interview Questions](#)

[Aws Interview Questions](#)

[Git Interview Questions](#)

[Java 8 Interview Questions](#)

[Mongodb Interview Questions](#)

[Dbms Interview Questions](#)

[Spring Boot Interview Questions](#)

[Power Bi Interview Questions](#)

[Pl Sql Interview Questions](#)

[Tableau Interview Questions](#)

[Linux Interview Questions](#)

[Ansible Interview Questions](#)

[Java Interview Questions](#)

[Jenkins Interview Questions](#)