

[Home](#) > [member-paid](#) > 04: JSP overview interview questions and answers

04: JSP overview interview questions and answers

 Posted on [August 24, 2014](#) by [Arulkumaran Kumaraswamipillai](#)


JSP or JavaScript based frameworks like angularjs, ember, etc are more popular than JSPs for web developemnt.

Q1. What is a JSP? How does it differ from a Servlet?

A1. JSP stands for Java Server Pages. JSP technology extends the Servlet technology, which means anything you can do with a Servlet you can do with a JSP as well. Writing **out.println (...)** statements using a Servlet is cumbersome and hard to maintain, especially if you need to send a long HTML page with little dynamic code content. Worse still, every single change requires recompilation of your Servlet.

Both Servlets and JSPs are complementary technologies. You can look at the JSP technology from an HTML designer's perspective as an extension to HTML with embedded dynamic content and from a Java developer's as an extension of the Java Servlet technology. JSP is commonly used as the presentation layer for combining HTML and Java code. While Java Servlet technology is capable of generating HTML with out.println(".....") statements, where "out" is a PrintWriter. This process of embedding HTML code with escape characters is cumbersome and hard to maintain. The JSP technology solves this by providing a level of abstraction so that the developer can use custom

600+ Full Stack Java/JEE Interview Q&As ♥Free ♦FAQs

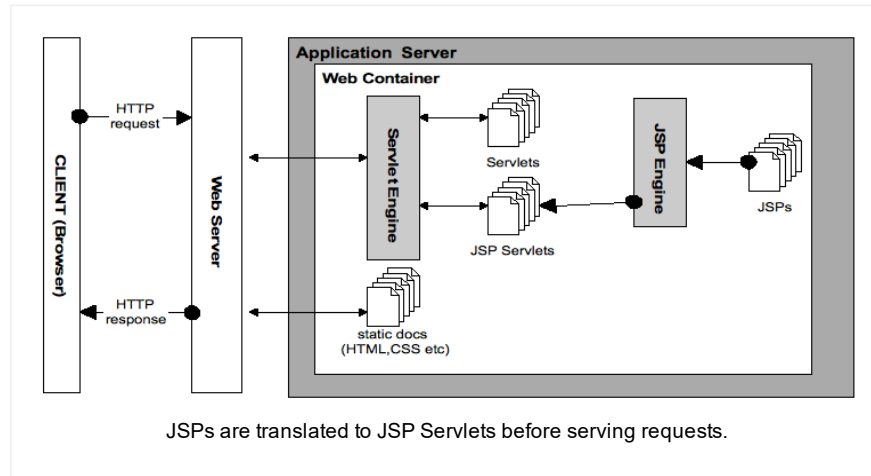
[open all](#) | [close all](#)

- ☒ [Ice Breaker Interview Q&A](#)
- ☒ [Core Java Interview Q&A](#)
- ☒ [JEE Interview Q&A \(37\)](#)
- ☒ [JEE Overview \(2\)](#)
- ☒ [Web basics \(8\)](#)
- ☒ [01: ♦ 12 Web basics](#)
- ☒ [02: HTTP basics on I](#)
- ☒ [03: Servlet interview](#)
- ☒ [04: JSP overview inte](#)
- ☒ [05: Web patterns inte](#)
- ☒ [06: ♦ MVC0, MVC1, I](#)
- ☒ [07: When to use a Se](#)
- ☒ [08: Web.xml interview](#)
- ☒ [WebService \(11\)](#)
- ☒ [JPA \(2\)](#)
- ☒ [JTA \(1\)](#)
- ☒ [JDBC \(4\)](#)
- ☒ [JMS \(5\)](#)
- ☒ [JMX \(3\)](#)
- ☒ [JNDI and LDAP \(1\)](#)
- ☒ [Pressed for time? Java/JE](#)
- ☒ [SQL, XML, UML, JSON, F](#)
- ☒ [Hadoop & BigData Interv](#)
- ☒ [Java Architecture Interview](#)
- ☒ [Scala Interview Q&As \(13\)](#)
- ☒ [Spring, Hibernate, & Mave](#)
- ☒ [Testing & Profiling/Sampli](#)
- ☒ [Other Interview Q&A for J](#)
- ☒ [▶ Free Java Interview Vid](#)

tags and action elements, which can speed up Web development and are easier to maintain.

Q2. What do you understand by the term JSP translation phase or compilation phase?

A2. As shown below in the figure the JSPs have a translation or a compilation process where the JSP engine translates and compiles a JSP file into a JSP Servlet. The translated and compiled JSP Servlet moves to the execution phase (run time) where they can handle requests and send responses.



Unless explicitly compiled ahead of time, JSP files are compiled the first time they are accessed. On large production sites, or in situations involving complicated JSP files, compilation may cause unacceptable delays to users first accessing the JSP page. The JSPs can be compiled ahead of time (i.e. precompiled) using application server tools/settings or by writing your own script.

Q3. What are the different scope values in a JSP?

A3. Page, Request, Session, and Application.

Scope	Object	Comment
Page	PageContext	Available to the handling JSP page only.
Request	Request	Available to the handling JSP page or Servlet and forwarded JSP page or Servlet.
Session	Session	Available to any JSP Page or Servlet within the same session.

16 Technical Key Areas

[open all](#) | [close all](#)

- [Best Practice \(6\)](#)
- [Coding \(26\)](#)
- [Concurrency \(6\)](#)
- [Design Concepts \(7\)](#)
- [Design Patterns \(11\)](#)
- [Exception Handling \(3\)](#)
- [Java Debugging \(21\)](#)
- [Judging Experience Interv](#)
- [Low Latency \(7\)](#)
- [Memory Management \(7\)](#)
- [Performance \(13\)](#)
- [QoS \(8\)](#)
- [Scalability \(4\)](#)
- [SDLC \(6\)](#)
- [Security \(13\)](#)
- [Transaction Management](#)

80+ step by step Java Tutorials

[open all](#) | [close all](#)

- [Setting up Tutorial \(6\)](#)
- [Tutorial - Diagnosis \(2\)](#)
- [Akka Tutorial \(9\)](#)
- [Core Java Tutorials \(20\)](#)
- [Hadoop & Spark Tutorials](#)
- [JEE Tutorials \(19\)](#)
- [Scala Tutorials \(1\)](#)
- [Spring & Hibernate Tutori](#)
- [Tools Tutorials \(19\)](#)
- [Other Tutorials \(45\)](#)

100+ Java pre-interview coding tests

[open all](#) | [close all](#)

- [Can you write code? \(22\)](#)
- [♦ Complete the given code](#)

Application	Application	Available to all the JSP pages and Servlets within the same Web Application.
-------------	-------------	--

- [Converting from A to B \(6\)](#)
- [Designing your classes &](#)
- [Java Data Structures & Al](#)
- [Passing the unit tests \(5\)](#)
- [What is wrong with this co](#)
- [Writing Code Home Assign](#)
- [Written Test Core Java \(3\)](#)
- [Written Test JEE \(1\)](#)

Q4. What are implicit objects and list them?

A4. Implicit objects are the objects that are available for the use in JSP documents without being declared first. These objects are parsed by the JSP engine and inserted into the generated Servlet. The implicit objects are:

Implicit object	Scope	Comment
request	Request	Refers to the current request from the client.
response	Page	Refers to the current response to the client.
pageContext	Page	Refers to the page's environment.
session	Session	Refers to the user's session.
application	Application	Same as ServletContext. Refers to the web application's environment.
out	Page	Refers to the outputstream.
config	Page	same as ServletConfig. Refers to the servlet's configuration.
page	Page	Refers to the page's Servlet instance.
exception	Page	<p>exception created on this page. Used for error handling. Only available if it is an errorPage with the following directive:</p> <pre>1 <%@ page isErrorPage="true" %></pre> <p>The "exception" implicit object is not available for global error pages declared through web.xml. You can retrieve the java.lang.Throwable object as follows:</p> <pre>1 <%= request.getAttribute("javax.servlet.error.exception") %></pre>

How good are your?

[open all](#) | [close all](#)

- [Career Making Know-how](#)
- [Job Hunting & Resume W](#)

Q5. What are the enhancements made to the EL in JEE 7?

A5. Java EE 7 is an overhaul of the Java Expression Language API known as Expression Language. The enhancements to the EL API

includes support for lambda expressions, static field and method access, as well as improvements for collection processing. Prior to Java EE 7, the Java Expression Language was a tightly coupled component of the JSF and JSP APIs. The EL API now offers developers the ability to invoke ad-hoc Java EL with “**ELProcessor**”.

```
1 ELProcessor el = new ELProcessor();
2 assert ((Integer)(el.eval("(x,y) -> x+y)(3, 2)")) == 5;
3
```

Q6. What Is the difference between **JSP** and **JSPF**?

A6. **JSPF** is **JSP Fragment**, which is a fragment included into another JSP page with the include directive, but not the include action. JSP Fragments can be compared to server side includes. These fragments are not compiled on their own, however there are compiled along side the page in which its included.

```
1 <%@include file="/WEB-INF/jspf/example.jspf" %>
```

Q7. What are custom tags in JSPs?

A7. Custom tags allow a cleaner separation between the look-and-feel of your application and its logic, compared to the original scriptlet syntax offered by JSP. There are two ways to implement a tag library: **tag files**, and **tag classes**. Tag files use a syntax that is nearly the same as JSP, but can be parameterized with attributes in the tag.

Tag class example:

To create a custom tag class, we need three things:

- 1) Tag handler class: In this class we specify what our custom tag will do when it is used in a JSP page.
- 2) TLD file: Tag descriptor file where we will specify our tag name, tag handler class and tag attributes.
- 3) JSP page: A JSP page where we will be using our custom tag.

```
1 package myapp.com;
2
3 import javax.servlet.jsp.tagext.*;
4 import javax.servlet.jsp.*;
5 import java.io.*;
6
7 public class Message extends SimpleTagSupport {
8     public void doTag() throws JspException, IOException
9     {
10         JspWriter out = getJspContext().getOut();
11         out.println("Sample custom tag.....");
12     }
13 }
```

14

TLD File: This file should present at the location: WEB-INF/ and it should have a .tld extension.

message.tld

```
1 <taglib>
2 <tlib-version>1.0</tlib-version>
3 <jsp-version>2.0</jsp-version>
4 <short-name>My Custom Tag</short-name>
5 <tag>
6 <name>Msg</name>
7 <tag-class>myapp.com.Message</tag-class>
8 <body-content>empty</body-content>
9 </tag>
10 </taglib>
11
```

Using custom tag in JSP: Above we have created a custom tag named "Msg". Here we will be using it.

```
1 <%@ taglib prefix="msgprefix" uri="WEB-INF/message.tld"%>
2 <html>
3 <head>
4 <title>Custom Tags in JSP Example</title>
5 </head>
6 <body>
7 <msgprefix:Msg/>
8 </body>
9 </html>
10
```

Using tag files in JSP 2.2 or later container

```
1 <%@ tag body-content="empty" %>
2 "Sample custom tag....."
3
```

put the tag file in **WEB-INF/tags/mytags/message.tag**

```
1 <%@ taglib prefix="msgprefix" tagdir="/WEB-INF/tags/mytag
2 <html>
3 <head>
4 <title>Custom Tags in JSP Example</title>
5 </head>
6 <body>
7 <msgprefix:message/>
8 </body>
9 </html>
```

Popular Posts

[♦ 11 Spring boot interview questions & answers](#)

825 views

[♦ Q11-Q23: Top 50+ Core on Java OOP Interview Questions & Answers](#)

766 views

[18 Java scenarios based interview Questions and Answers](#)

400 views

[001A: ♦ 7+ Java integration styles & patterns interview questions & answers](#)

388 views

[01b: ♦ 13 Spring basics Q8 – Q13 interview questions & answers](#)

295 views

[♦ 7 Java debugging interview questions & answers](#)

293 views

[01: ♦ 15 Ice breaker questions asked 90% of the time in Java job interviews with hints](#)

285 views

[♦ 10 ERD \(Entity-Relationship Diagrams\) Interview Questions and Answers](#)

279 views

[♦ Q24-Q36: Top 50+ Core on Java classes, interfaces and generics interview questions & answers](#)

239 views

[001B: ♦ Java architecture & design concepts interview questions & answers](#)

201 views

Bio

Latest Posts

**Arulkumaran Kumaraswamipillai**

Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via [Amazon.com](#) in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site. **945+** paid members. [join my LinkedIn Group](#). [Reviews](#)

**About Arulkumaran Kumaraswamipillai**

Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers

to choose from. It pays to prepare. So, published Java interview Q&A books via [Amazon.com](https://www.amazon.com) in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site. **945+** paid members. [join my LinkedIn Group](#). [Reviews](#)

◀ 05: Web patterns interview Q&A

◆ 18 Agile Development interview Q&A for Java developers ▶

Posted in member-paid, Web basics

Empowers you to open more doors, and fast-track

Technical Know Hows

☀ [Java generics in no time](#) ☀ [Top 6 tips to transforming your thinking from OOP to FP](#) ☀ [How does a HashMap internally work? What is a hashing function?](#)
☀ [10+ Java String class interview Q&As](#) ☀ [Java auto un/boxing benefits & caveats](#) ☀ [Top 11 slacknesses that can come back and bite you as an experienced Java developer or architect](#)

Non-Technical Know Hows

☀ [6 Aspects that can motivate you to fast-track your career & go places](#) ☀ [Are you reinventing yourself as a Java developer?](#) ☀ [8 tips to safeguard your Java career against offshoring](#) ☀ [My top 5 career mistakes](#)

Prepare to succeed

☀ [Turn readers of your Java CV go from “Blah blah” to “Wow”?](#) ☀ [How to prepare for Java job interviews?](#)
☀ [16 Technical Key Areas](#) ☀ [How to choose from multiple Java job offers?](#)

Select Category ▼

© Disclaimer

The contents in this Java-Success are copy righted. The author has the right to correct or enhance the current content without any prior notice.

These are general advice only, and one needs to take his/her own circumstances into consideration. The author will not be held liable for any damages caused or alleged to be caused either directly or indirectly by these materials and resources. Any trademarked names or labels used in this blog remain the property of their respective trademark owners. No guarantees are made regarding the accuracy or usefulness of content, though I do make an effort to be accurate. Links to external sites do not imply endorsement of the linked-to sites.