**Descriptive question on JSP**

1. **What do you understand by JSP Actions?**

JSP actions are tags that direct the server to use existing components or control the behavior of the JSP engine. JSP Actions consist of a typical (XML-based) prefix of "jsp" followed by a colon, followed by the action name followed by one or more attribute parameters.   
There are six JSP Actions:   
< jsp : include / >   
< jsp : forward / >   
< jsp : plugin / >   
< jsp : useBean / >   
< jsp : setProperty / >   
< jsp : getProperty / >

1. **What is the difference between <jsp:forward page = ... > and response.sendRedirect(url)?**

* The element forwards the request object containing the client request information from one JSP file to another file. The target file can be an HTML file, another JSP file, or a servlet.
* sendRedirect sends HTTP temporary redirect response to the browser, and browser creates a new request to go the redirected page. The response.sendRedirect also kills the session variables.

1. **Identify the advantages of JSP over Servlet?**

a) Embedding of Java code in HTML pages  
b) Platform independence  
c) Creation of database-driven Web applications  
d) Server-side programming capabilities

1. **What are all the different scope values for the <jsp:useBean> tag?**

< jsp : useBean > tag is used to use any java object in the jsp page. Here are the scope values for < jsp : useBean > tag:  
a) page  
b) request  
c) session and  
d) application

1. **What is JSP Scriptlet?**

JSP Scriptlets is a term used to refer to pieces of Java code that can be embedded in a JSP PAge. Scriptlets begins with <% tag and ends with %> tag. Java code written inside scriptlet executes every time the JSP is invoked.

Example

  <%  
  //java codes  
   String userName=null;  
   userName=request.getParameter("userName");  
   %>

1. **What you will handle the runtime exception in your jsp page?**

The errorPage attribute of the page directive can be used to catch run-time exceptions automatically and then forwarded to an error processing page. You can define the error page to which you want the request forwarded to, in case of an exception, in each JSP Page.

1. **What are the implicit objects in JSP?**

Implicit objects are the objects available to the JSP page. These objects are created by Web container and contain information related to a particular request, page, or application. The JSP implicit objects are:  
application,config,exception,out,page,pageContext,request,response and session

1. **Why we use Servlets?**

Servlets are used to process the client requests.  
\* A Servlet can handle multiple requests concurrently and be used to develop high performance of system  
\* A Servlet can be used to load balance among serveral servers, as Servlet can easily forward request.

1. **Write the syntax of EL expression? Why we use them?**

-Syntax of EL Expression:

EL expressions are always within the braces { ..... } and prefixed with the $ or, # sign as:

<jsp:useBean id=”bean” class=”class”/>

${bean.name}

-Expression language use to easily access application data stored in JavaBeans components. For example, the JSP expression language allows a page author to access a bean using simple syntax such as ${name} for a simple variable or ${name.foo.bar} for a nested property.

1. **What is Java Server Pages Standard Tag Library (JSTL)?**

Java Server Pages Standard Tag Library (JSTL) encapsulates as simple tags the core functionality common to many Web applications. It also provides a framework for integrating existing custom tags with JSTL tags.

It is a collection of four tag library. They are –

(a)Core (b) Internationalization (118n) and formatting

(c) Relational database access (d) XML processing

1. **What is deployment descriptor?**

The deployment descriptor(web.xml) is an xml file that contains the basic and most important information that is required to deploy a web application (Servlet)   
Without this, the web server would not know, which requests to entertain/consider as requests to access this servlet.

1. **What is Tag Library Descriptor (TLD)?**

Atag library descriptor is an XML document that contains information about a library as a whole and about each tag contained in the library. TLDs are used by a web container to validate the tags and by JSP page development tools. A TLD must begin with a root taglib element.

The syntax for the taglib directive is as follows:

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

1. **Difference between GET and POST in Java Servlet?**

Get sends information from the browser to the Servlet as contents appended to the query string in the URL while Post uses hidden variables   
\* Because of the above point, post is a lot safer than get  
\* Get has a size limitation - i.e., we can send only approximately 1 Kb of data while Post can send a significantly higher amount of data  
\* Get is the most common type of sending data from a browser to a servlet, while Post is gaining popularity because of its size and safety which is better than the get.

* doPost has no limitations on paramater numbers while doGet has.
* doGet is faster than doPost.
* doPost is secured than doGet.

1. **Write the Expanded Directory Format?**

The web application in its predefined structure is simply copied into the container’s deployment directory.

Root-web-context

META-INF

Context.xml

WEB-INF

Web.xml

Lib-all libraries

Classes-all package and classes

Index.jsp and other jsp pages

1. **How do you configure a Servlet?**

First you configure the servlet. This is done using the <servlet> element. Here you give the servlet a name, and writes the class name of the servlet. Second, you map the servlet to a URL or URL pattern. This is done in the <servlet-mapping> element. In the above example, all URL's ending in html are sent to the servlet.

<servlet>

<servlet-name>controlServlet</servlet-name>

<servlet-class>org.idb.j2ee.ControlServlet</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>controlServlet</servlet-name>

<url-pattern>\*.html</url-pattern>

</servlet-mapping>

1. **What are the advantages of JSF?**

**Ans**: The JSF specification lists the following ways that JSF helps web-application developers to create user

interfaces (UIs):

• Makes it easy to construct a UI from a set of reusable UI components

• Simplifies migration of application data to and from the UI

• Helps manage UI state across server requests

• Provides a simple model for wiring client-generated events to server-side application code

• Allows custom UI components to be easily built and reused

# What are tags in JSF?

JSF application typically uses JSP pages to represent views. JSF provides useful special tags to enhance these views. Each tag gives rise to an associated component. JSF (Sun Implementation) provides 43 tags in two standard JSF tag libraries:

1. JSF Core Tags Library

2. JSF Html Tags Library Even a very simple page uses tags from both libraries. These tags can be used adding the following lines of code at the head of the page.

<%@ taglib uri=”http://java.sun.com/jsf/core “ prefix=”f” %> (For Core Tags)

<%@ taglib uri=”http://java.sun.com/jsf/html “ prefix=”h” %> (For Html Tags)

# What is JSF life cycle and its phases?

T he series of steps followed by an application is called its life cycle. A JSF application typically follows six steps in its life.   
1. Restore view phase   
2. Apply request values phase   
3. Process validations phase   
4. Update model values phase   
5. Invoke application phase   
6. Render response phase

# What is JavaServer Pages Standard Tag Library (JSTL)?

A tag library that encapsulates core functionality common to many JSP applications. JSTL has support for common, structural tasks such as iteration and conditionals, tags for manipulating XML documents, internationalization and locale-specific formatting tags, SQL tags, and functions.

1. **What is Tag Library Descriptor (TLD)?**

**Ans:** To use a custom tag library, the web container needs to be made aware of specific information about the library itself. A special file called a tag library descriptor (TLD) is used for this purpose. The TLD file contains essential information about each of the custom actions or tags that are included inside the tag library, such as which attributes are permitted by which tags, whether the tags accept body content, and so on.

1. **Write the Expanded Directory Format? Pg-12**

The Java Servlet specification defines a special structure that all web applications must follow so that servlet/JSP containers know exactly where to find the resources that compose the web application. Most containers allow web applications to be deployed in one of the following two forms:

• **Expanded directory format**: The web application in its predefined structure is simply copied into the container’s deployment directory.

• **Web ARchive file (WAR)**

1. **What is Taglib? Write the syntax of taglib?**

The JavaServer Pages API allows us to define custom JSP tags that look like HTML or XML tags and a tag library is a set of user-defined tags that implement custom behavior.

The **taglib** directive declares that your JSP page uses a set of custom tags, identifies the location of the library, and provides a means for identifying the custom tags in JSP page.

The general syntax :

<%@ taglib uri="uri" prefix="prefixOfTag" >

Example:

<%@ taglib prefix="f" uri="http://java.sun.com/jsf/core"%>

1. **How do you configure a Servlet?**

Several list of configuration for servlet in web.xml (deployment descriptor):

1. Configuring and Mapping a Servlet
2. Servlet Init Parameters
3. Servlet Load-on-Startup
4. Context Parameters

Most important is to Configuring and Mapping a Servlet

<web-app>

<servlet>

<servlet-name>controlServlet</servlet-name>

<servlet-class>com.jenkov.butterfly.ControlServlet</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>controlServlet</servlet-name>

<url-pattern>\*.html</url-pattern>

</servlet-mapping>

</web-app>

First to configure the servlet. This is done using the <servlet> element. Here you give the servlet a name, and writes the class name of the servlet.

Second, map the servlet to a URL or URL pattern. This is done in the <servlet-mapping> element. In the above example, all URL's ending in .html are sent to the servlet.

Other possible servlet URL mappings are:

/myServlet

/myServlet.do

/myServlet\*

1. **Write the default value of EL expression. Pg-98**

Default values are type-correct values that are assigned to a subexpression when there is a problem, and errors are exceptions to be thrown (and then handled by the standard JSP error-handling process). An example of such a default value is 'infinity'. This value is assigned to an expression that results in a divide by zero. For example, the following piece of EL will display infinity rather than causing an error:

${2/0}

The equivalent Java expression would throw an ArithmeticException.

1. **What is the type of request /response handily by JSF? Pg-185**

Several kinds of request/response cycles can occur in a JSF-enabled application. We are concerned with these three request/response pairs:

• Non-JSF request generates JSF response   
• JSF request generates JSF response   
• JSF request generates non-JSF response

Of course, you can also have a non-JSF request that generates a non-JSF response. Because this does not involve JSF in any way, the JSF life cycle does not apply.

1. **Write the name of JSF libraries?**

• **Six JSF JARs:** commons-beanutils.jar, commons-collections.jar, commons-digester.jar,

commons-logging.jar, jsf-api.jar, and jsf-impl.jar

• **Two JSTL JARs:** jstl.jar and standard.jar

1. **What is the job of faces-config.xml class?**

faces-config.xml can contain a lot of information about a web application. Main two things are identify the flow of control and identify the JavaBean used by the application.

1. **cWrite the event handling process in JSF?**

Java Server Faces technology supports three kinds of events:

* action events
* value-change events
* data-model events.

**Different ways of handling events in JSF**

* Implement a method in a backing bean to handle the event and refer that method with a JSF EL expression from the appropriate attribute of the component
* Implement an event listener to handle the event and registers the listener on a component by nesting a listener tag inside the UI component tag.

1. **What do you mean by MVC in JSF?**

In the big architectural picture, JSF code is the V:

**M** - Business domain/Service layer (e.g. EJB/JPA/DAO)  
**V** - Your JSF code  
**C** - FacesServlet

In the smaller developer picture, the architectural **V** is in turn dividable as follows:

**M** - Entity  
**V** - JSP/XHTML page  
**C** - Managed bean

1. **The JSP life cycle**

****