**Descriptive question on JSP & JSF**

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

Ans: 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)?**

Ans: 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, as long as it is in the same application context as the forwarding JSP file.   
  
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?**

Ans: JSP can contain HTML, JavaScript, XML and Java Code whereas Servlets can contain only Java Code, making JSPs more flexible and powerful than Servlets.   
However, Servlets have their own place in a J2EE application and cannot be ignored altogether. They have their strengths too which cannot be overseen.

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

Ans: < 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?**

Ans: 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.

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

Ans: 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. Also, there should be another JSP that plays the role of the error page which has the flag isErrorPage set to True.

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

Ans: 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?**

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

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

Ans:

1. JavaServer Faces architecture makes it easy for the developers to use. In JavaServer Faces technology, user interfaces can be created easily with its built-in UI component library.

2. Offers a clean separation between behavior and presentation.  
 3. Provides a rich architecture for managing component state, processing component data, validating user input, and handling events.  
 4. Robust event handling mechanism.  
 5. Events easily tied to server-side code.  
 6. Render kit support for different clients  
 7. Component-level control over statefulness  
 8. Highly 'pluggable' - components, view handler, etc  
 9. JSF also supports internationalization and accessibility  
 10. Offers multiple, standardized vendor implementations

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

Ans: 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.

# What are tags in JSF?

Ans:JSF provides two standard JSF tag libraries:

1. **JSF Core Tags  Library**

JSF Core Tags Action is usually used with the HTML actions to modify the behavior of those actions. Its categories are Converters, Listeners, Miscellaneous, Selection, Validators, View etc. <%@ taglib uri=”<http://java.sun.com/jsf/core> “ prefix=”f” %>

1. **JSF Html Tags Library**

# These custom action are used to create HTML elements. These action fall into five categories: input, output, selection, commands and miscellaneous. <%@ taglib uri=”<http://java.sun.com/jsf/html> “ prefix=”h” %>

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

Ans: JavaServer Pages Standard Tag Library (JSTL) encapsulates as simple tags the core functionality common to many Web applications. JSTL has support for common, structural tasks such as iteration and conditionals, tags for manipulating XML documents, internationalization tags, and SQL tags. 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?**

Ans: 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)?**

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

The syntax for the taglib directive is as follows:

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

[or]

Ans: A tag 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?**

Ans: 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.

# What is JSF life cycle and its phases?

# Ans: Request/response cycles can occur in a JSF-enable application. 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

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

Ans: 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. **What is Taglib? Write the syntax of taglib?**

Ans: A tag library (commonly known as a taglib) is simply a collection of one or more custom tags that are generally related in some way. For example the JSP 2.0 specification introduced a standard tag library known as the JSTL.

For example, suppose the **custlib** tag library contains a tag called **hello**. Your JSP file as follows:

<%@ taglib uri="http://www.example.com/custlib" prefix="mytag" %>

<html>

<body><mytag:hello/></body></html>

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

Ans: 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. **Write the default value of EL expression.**

Ans: Default values are type-correct values that are assigned to a subexpression when there is a problem and errors are exceptions to be thrown. An example of such a default value is ‘infinity’. This value is assigned to an expression that results in a divide by zero.

Exp: ${2/0}

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

Ans: We can have a JSF response or a non-JSF response. The following types of request/response handily by JSF –

* 1. Non-JSF request generates JSF response
  2. JSF request generates JSF response
  3. JSF request generates non-JSF response

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

Ans: The name of JSF library as follow –

1. Six JSF JARs: commons-beanutils.jar, commons-collections.jar, commons-digester.jar, commons-logging.jar, jsf-api.jar and jsf-impl.jar
2. Two JSTL JARs: jstl.jar and standard.jar
3. **What is the job of faces-config.xml class?**

Ans:

\*Registering Managed Beans in faces-config.xml

\*Creating Navigation Cases in faces-config.xml

\*Registering Custom Validators and Converters in faces-config.xml

\*Registering a Message Bundle in faces-config.xml

\*Registering Default and Supported Locales in faces-config.xml

\*Registering the ADF Phase Listener in faces-config.xml

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

Ans: There are two basic types of event handlers in JSF

Action Events are emitted for UI Command objects like Command Button or a Hyper-link.

<h:commandButton value = "Test Action Listener"

actionListener = "#{testActionListener.doSomeAction}" />

Value Change Events are applicable for UI Components like Text Field, Check-Box, List and Radio Buttons.

<h:selectOneMenu value="#{addressBean.country}"

    valueChangeListener="#{addressBean.populatePinNumber}">

</h:selectOneMenu>

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

Ans: The primary design pattern of JSF is the Model-View-Controller (MVC) pattern. MVC separates application architecture into three categories of components: model, view and controller. The mode is the abstraction of all the domain data in the system represent by manage bean. The view(jsp) is the visualization of the model. The controller is the set of components that manage the communications between model and view. In JSF it is FacesServlet configured with faces-config.xml.