Java Server Faces Interview Questions And Answers Guide.

Java Server Faces Job Interview Preparation Guide.

Question # 1

What is JSF Taglibs?

Answer:-

The JSF Taglibs are of two.
   HTML Tag lib
   Core Tag lib
HTML Tag lib: This tag library contains all the tags that pertains all the tags that are to deal with forms and other HTML-specific stuff.
Core Tag lib: The Core Tag lib contains the tags that pertaining all the tags that are to deal with logic, validation, controller and other JSF-specific stuff.
The two tag libs are to be specified as follows:
<%@ taglib uri="http://java.sun.com/jsf/html" prefix="h" %>
<%@ taglib uri="http://java.sun.com/jsf/core" prefix="f" %>

Read More Answers.

Question # 2

Tell me how the components of JSF are rendered?

Answer:-

JSF libraries need to be added in an application.
On the .jsp page, a tag library needs to be added:
<%@ taglib uri="http://java.sun.com/jsf/core" prefix="f"%>
<%@ taglib uri="http://java.sun.com/jsf/html" prefix="h"%>
In an XML style:
<?xml version="1.0"?>
<jsp:root version="2.0" xmlns:jsp="http://java.sun.com/JSP/Page" xmlns:f="http://java.sun.com/jsf/core" xmlns:h="http://java.sun.com/jsf/html">
The JSF components can be accessed using the prefix attached.
In an IDE, JSF can be added easily. However, when working without them the faces-config.xml needs to be updated and populated with classes i.e. Managed Beans between the following tags

Read More Answers.

Question # 3

Explain JSF Architecture?

Answer:-

JSF developed based on MVC design pattern. Therefore, the applications can be scaled better and well maintained. It is standardized pattern driven by Java Community Process. The advantage of JSF is both Java web user interface and a framework which perfectly fits with MVC. The presentation and behaviour of a web application is a clean separation. UI can be created by web page authors and the business logic can be utilized by managed beans. There are controllers which can be used to perform user actions.

Read More Answers.

Question # 4

Tell me JSF supports AJAX?

Answer:-

Ajax support can be added to the following panel components.
   <hx:panelActionbar>
   <hx:panelBox>
   <hx:panelDialog>
   <hx:panelFormBox>
   <h:panelGrid>
   <h:panelGroup>
   <hx:panelLayout>
   <hx:panelMenu>
   <hx:panelSection>
To add Ajax support to a JSF page:

1. Drag a panel component from the palette onto the JSP.
2. Drag additional components (for example an input component) from the palette onto the panel component.
3. In the properties view, select the panel component tag. Select the Ajax tab.
4. On the Ajax tab, select the Allow Ajax updates checkbox.
5. Select the type of Ajax request that you want to use:
    o Refresh - A GET request for the same page.
    o Submit - A POST request for the same page.
    o External - A GET request for a different page.
6. Configure the request parameters.
    o hx:ajaxRefreshRequest
    o hx:ajaxExternalRequest
Note: If the component, the value of which needs to be used as a parameter is contained by a data table, you must manually add $$AJAXROW$$component client id> to the parameter set. Where <component client id> is a client ID of the component whose value you want to send as the parameter.

Read More Answers.

Question # 5

What is rendering of page in JSF?

Answer:-

A JSF page has components that are made with the help of JSF library. The JSF components lke h:form, h:inputText, h:commandButton, etc., are rendered / translated to HTML output. This process is known as encoding. Encoding also assigns a unique id to a component by the framework and the ids are generated at random.

Read More Answers.

Question # 6

What is Facelets?

Answer:-

The JSF and JSP are aligned in a web application; Facelets provides a high performance JSF-centric view which is the outside of JSP specification. The designing of Facelets are similar to that of JSP creation. The difference between these is that, the entire JSP vendor API is removed and enhances the JSF performance which provides easy plug-and-go development.

Read More Answers.

Question # 7

Explain Shale?

Answer:-

Shale is a framework for web applications, based on JSF. Shale architecture includes a set of loosely coupled services which can be combined to meet specific application requirements. The additional functionality provided by Shale - application event callbacks, dialogs with conversation-scoped state, a view technology named Clay, annotation-based functionality. These functionalities will reduce the requirement of configuration and provides support for remote computing. Other framework links for integration is provided by Shale, which eases the combined technology based development.

Read More Answers.

Question # 8

Do you know what is AJAX and what problem does it solve?

Answer:-

Ajax is a set of client side technologies that allows asynchronous communication between client and web server. In synchronous communication, complete round trip happens with each request/response action event when small data of the page to be refreshed. Ajax has solved this problem of posting entire information every time through asynchronous communication.
XmlHttpRequest is the basic fundamental behind Ajax. This allows browser to communicate with server without making post backs.

Read More Answers.

Question # 9

What is Apache Tomahawk?

Answer:-

A series of JSF components are provided by MyFaces Tomahawk. Tomahawk provides several custom components that are 100% JSF compatible components. It is easier to implement JSF projects with Tomahawk. The extended functionalities are â€" improved inputText components, Converter interfaces and solutions to use JSF and Tiles together.

Read More Answers.

Question # 10

What is JSF life cycle?

Answer:-

i. Restore view phase:
This phase starts when a user requests a JSF page by clicking a link, button etc. In this phase view generation of the page, binding of components to its event handlers and validators are performed and view is saved in the FacesContext object.
ii. Apply request values phase:
The purpose of this phase is for each component to retrieve its current state.
iii. Process validations phase:
During this phase local values stored for the component in the tree are compared to the validation rules registered for the components.
iv. Update model values phase:
After confirming that data is valid in the previous phase local values of components can be set to corresponding server side object properties i.e. backing beans.

v. Invoke application phase:
Before this phase the component values have been converted, validated, and applied to the bean objects, so you can now use them to execute the application's business logic.
vi. Render response phase:
In this phase JSP container renders the page back to the user, if jsp is used by application i.e. view is displayed with all of its components in their current state

<u>**Read More Answers.**</u>

Question # 11

Do you know JSF and AJAX?

Answer:-

AJAX and JSF together makes a rich web application. AJAX usage is focused on creating richer user interfaces. The UI components in JSF and MyFaces were originally developed using HTML. AJAX-enabled components can easily be added to JSF-based User Interfaces, as JSF supports the output formats at multiple levels. The support for AJAX for the implementation of UI logic that is to run on the client and also on the server; is provided by JSF 2.0 specification. When the Java Script in the browser is disabled, JSF 2.0 provides the support for degradation.

<u>**Read More Answers.**</u>

Question # 12

Explain JSF framework?

Answer:-

JavaServer Faces (JSF) is a user interface framework for building web applications that run on the server side and render the user interface back to the client.
It lets you develop tools that simplify coding web-based Java applications.
Users of your JSF-based web applications can use the wide range of user actions made available by JSF controls. More features can be offered more conveniently than they can be with a standard HTML front end. JSF requires little more effort than an ordinary JSP configuration but still has more benefits.

<u>**Read More Answers.**</u>

Question # 13

Explain JSF life cycle?

Answer:-

The JavaServer Faces page is similar to JSP page in terms of their life cycle.
When user makes a request, the life cycle of a JSF web application, starts.
Several tasks are performed by JSF when a user submits a page. These tasks include data validation, data type conversion, etc.
A JSF application supports two kinds of responses and two kinds of requests.
Faces response:
Request processing life cycle creates a servlet response by the execution of the Render Response Phase of the.
Non-Faces response:
A servlet response is not created by the execution of the render response phase.
Faces request:
A servlet request is sent from a previously generated Faces response. E.g.: The request URI in the form submission from a JSF UI component, identifies the JSF component tree to use for processing the request.
Non-Faces request:
A servlet request is sent to an application component, such as a servlet or JSP page, rather than directed to a JavaServer Faces component tree.
The steps in the life cycle of a JSF application are:
1. Restore View Phase
2. Apply Request Value Phase
3. Process Validations Phase
4. Update Model Value Phase
5. Invoke Application Phase
6. Render Response Phase

<u>**Read More Answers.**</u>

Question # 14

What is Seam?

Answer:-

Seam is a framework for web application, which combines the existing popular frameworks â€" EJB and JSF. A back-end component of an enterprise (EJB) can be accessed by the front-end by addressing its name of the Seam component.
The concept of â€˜context' is expanded by Seam. The context contains each Seam component. The context of a session captures the actions of all logged in users until he logs out. A command line tool seam-gen can be used to automatically generate the actions â€" create, read, update, delete (CRUD) of a web application. The integration of Seam can be with JBOSS, RichFaces or ICEFaces AJAX libraries.

<u>**Read More Answers.**</u>

Question # 15

What is JavaServer Faces, JSF?

Answer:-

JavaServer Faces technology includes:
    A set of APIs for representing UI components and managing their state, handling events and input validation, defining page navigation, and supporting internationalization and accessibility.
    A JavaServer Pages (JSP) custom tag library for expressing a JavaServer Faces interface within a JSP page.
Designed to be flexible, JavaServer Faces technology leverages existing, standard UI and web-tier concepts without limiting developers to a particular mark-up language, protocol, or client device.
Ease-of-use being the primary goal, the JavaServer Faces architecture clearly defines a separation between application logic and presentation while making it easy to connect the presentation layer to the application code.

Developed through the Java Community Process under JSR - 314, JavaServer Faces technology establishes the standard for building server-side user interfaces.

Read More Answers.

Question # 16

Can you explain JSF Ajax components?

Answer:-

The custom web tier components for JSF can be developed using AJAX along with JSF. Using JSF the state of an application, event handling, input validation and page navigation tasks can be done for user interface components. AJAX technology is used for client side Java Script leverage in order to provide richer and more responsive user experience.
To implement the JSF component for spell check text area, the following tag is used.
<jcr: spellCheckTextArea cols="30" rows="15" value="# {user.interests}" />
This component extends HtmlInputTextArea provides a simple AJAX-enabled spell checking facility

Read More Answers.

Question # 17

Explain how JSF different from conventional JSP?

Answer:-

JSP is good for mixing static content and dynamic content pulled from resources made available by other parts of the application; for instance, a servlet.
    JSP's main mission in life is to generate a response to a request; a JSP page is processed in one pass from top to bottom, with JSP action elements processed in the order in which they appear in the page.
    However, JSF has a much more complex lifecycle.
    JSF components get created, process their input (if any), and then render themselves.
    For JSF to work well, these three things must happen separately in a well-defined order, but when JSF is used with JSP, they don't.
    Instead, the component creation and rendering happens in parallel, causing all kinds of problems.

Read More Answers.

Question # 18

Do you know how to declare the Navigation Rules for JSF?

Answer:-

A navigation rule specifies the JSF implementation which page need to send back to the browser after submitting a form. For instance, after the successful login, the page should to Main page or to return on the same login page. The following code snippet describes this.
<navigation-rule>
<from-view-id>/login.jsp</from-view-id>
<navigation-case>
<from-outcome>login</from-outcome>
<to-view-id>/main.jsp<to-view-id>
</navigation-case>
<navigation-case>
<from-outcome>fail</from-outcome>
<to-view-id>/login.jsp<to-view-id>
</navigation-case>
</navigation-rule>
from-outcome to be match with action attribute of the command button of the login.jsp as:
<h:commandbutton value="Login" action="login"/>
Secondly, it should also match with the navigation rule in face-config.xml as
<managed-bean>
<managed-bean-name>user</managed-bean-name>
<managed-bean-class>core.jsf.LoginBean</managed-bean-class>
<managed-bean-scope>session</managed-bean-scope>
</managed-bean>
In the UI component, to be declared / used as:
<h:inputText value="#{user.name}"/>
value attribute is the name property of the user bean.

Read More Answers.

Java Programing Most Popular Interview Topics.

1 : [Java Frequently Asked Interview Questions and Answers Guide.](#)

2 : [Java Swing Programming Frequently Asked Interview Questions and Answers Guide.](#)

3 : [J2EE Frequently Asked Interview Questions and Answers Guide.](#)

4 : [Spring Framework Frequently Asked Interview Questions and Answers Guide.](#)

5 : [Hibernate Frequently Asked Interview Questions and Answers Guide.](#)

6 : [JBoss Frequently Asked Interview Questions and Answers Guide.](#)

7 : [Core Java Frequently Asked Interview Questions and Answers Guide.](#)

8 : [IBM WebSphere Frequently Asked Interview Questions and Answers Guide.](#)

9 : [J2SE Frequently Asked Interview Questions and Answers Guide.](#)

10 : [Java Servlet Programming Frequently Asked Interview Questions and Answers Guide.](#)

About Global Guideline.

Global Guideline is a platform to develop your own skills with thousands of job interview questions and web tutorials for fresher's and experienced candidates. These interview questions and web tutorials will help you strengthen your technical skills, prepare for the interviews and quickly revise the concepts. Global Guideline invite you to unlock your potentials with thousands of **Interview Questions with Answers** and much more. Learn the most common technologies at Global Guideline. We will help you to explore the resources of the World Wide Web and develop your own skills from the basics to the advanced. Here you will learn anything quite easily and you will really enjoy while learning. Global Guideline will help you to become a professional and Expert, well prepared for the future.

* This PDF was generated from https://GlobalGuideline.com at **November 29th, 2023**

* If any answer or question is incorrect or inappropriate or you have correct answer or you found any problem in this document then don't hesitate feel free and e-mail us we will fix it.

You can follow us on FaceBook for latest Jobs, Updates and other interviews material.
 www.facebook.com/InterviewQuestionsAnswers

Follow us on Twitter for latest Jobs and interview preparation guides
https://twitter.com/InterviewGuide

Best Of Luck.

Global Guideline Team
https://GlobalGuideline.com
Info@globalguideline.com