

# Struts Interview Questions

## 1) What is Struts?

The core of the Struts framework is a flexible control layer based on standard technologies like Java Servlets, JavaBeans, ResourceBundle, and XML, as well as various Jakarta Commons packages. Struts encourages application architectures based on the Model 2 approach, a variation of the classic Model-View-Controller (MVC) design paradigm.

Struts provides its own Controller component and integrates with other technologies to provide the Model and the View. For the Model, Struts can interact with standard data access technologies, like JDBC and EJB, as well as most any third-party packages, like Hibernate, iBATIS, or Object Relational Bridge. For the View, Struts works well with JavaServer Pages, including JSTL and JSF, as well as Velocity Templates, XSLT, and other presentation systems.

The Struts framework provides the invisible underpinnings every professional web application needs to survive. Struts helps you create an extensible development environment for your application, based on published standards and proven design patterns.

## 2) What is Jakarta Struts Framework?

Jakarta Struts is open source implementation of MVC (Model-View-Controller) pattern for the development of web based applications. Jakarta Struts is robust architecture and can be used for the development of application of any size. Struts framework makes it much easier to design scalable, reliable Web applications with Java.

## 3) What is ActionServlet?

The class `org.apache.struts.action.ActionServlet` is the called the ActionServlet. In the the Jakarta Struts Framework this class plays the role of controller. All the requests to the server goes through the controller. Controller is responsible for handling all the requests.

## 4) How you will make available any Message Resources Definitions file to the Struts Framework Environment?

T Message Resources Definitions file are simple .properties files and these files contains the messages that can be used in the struts project. Message Resources Definitions files can be added to the `struts-config.xml` file through `<message-resources />` tag.

Example:

```
<message-resources parameter=\"MessageResources\" />.
```

## 5) What is Action Class?

The Action Class is part of the Model and is a wrapper around the business logic. The purpose of Action Class is to translate the `HttpServletRequest` to the business logic. To use the Action, we need to Subclass and overwrite the `execute()` method. In the Action Class all

the database/business processing are done. It is advisable to perform all the database related stuffs in the Action Class. The ActionServlet (command) passes the parameterized class to Action Form using the execute() method. The return type of the execute method is ActionForward which is used by the Struts Framework to forward the request to the file as per the value of the returned ActionForward object.

## **6) What is ActionForm?**

An ActionForm is a JavaBean that extends org.apache.struts.action.ActionForm. ActionForm maintains the session state for web application and the ActionForm object is automatically populated on the server side with data entered from a form on the client side.

## **7) What is Struts Validator Framework?**

Struts Framework provides the functionality to validate the form data. It can be used to validate the data on the user's browser as well as on the server side. Struts Framework emits the JavaScripts and it can be used to validate the form data on the client browser. Server side validation of form can be accomplished by subclassing your Form Bean with DynaValidatorForm class.

The Validator framework was developed by David Winterfeldt as a third-party add-on to Struts. Now the Validator framework is a part of Jakarta Commons project and it can be used with or without Struts. The Validator framework comes integrated with the Struts Framework and can be used without doing any extra settings.

## **8) Give the Details of XML files used in Validator Framework?**

The Validator Framework uses two XML configuration files: validator-rules.xml and validation.xml. The validator-rules.xml defines the standard validation routines; these are reusable and used in validation.xml to define the form-specific validations. The validation.xml defines the validations applied to a form bean.

## **9) How you will display validation fail errors on JSP page?**

Following tag displays all the errors:

```
<html:errors/>>
```

## **10) How you will enable front-end validation based on the XML in validation.xml?**

The <html:javascript> tag allows front-end validation based on the XML in validation.xml. For example, the code: <html:javascript formName=\"logonForm\" dynamicJavascript=\"true\" staticJavascript=\"true\" /> generates the client-side JavaScript for the form \"logonForm\" as defined in the validation.xml file. The <html:javascript> when added in the JSP file generates the client-side validation script.

## **11) How to get data from the velocity page in a action class?**

We can get the values in the action classes by using `data.getParameter(\"variable name defined in the velocity page\")`;