

Struts 2

Session Aware Interface Example

The **Session Aware** interface must be implemented by the Action class to store the information in the session scope.

Method of Session Aware interface

It contains only one method `setSession`.

Public and abstract Method	Description
void setSession(Map<String,Object> map)	struts framework calls this method by passing the instance of SessionMap class.

SessionMap class

The struts 2 framework passes the instance of **org.apache.struts2.dispatcher.SessionMap**. It extends the **java.util.AbstractMap** class which implements the **java.util.Map.SessionMap**. There are many useful methods of SessionMap class.

Example of SessionAware interface

This example contains three links **login**, **logout** and **profile**. You cannot click on the profile page until you are logged in. After getting logged in, you may go to the profile page. If the end user clicks on the logout link, he will not be able to access the profile page.

Steps are as follows:

1. **add struts and servlet library**
2. **index.jsp** for providing links to the login, logout and profile.
3. **struts.xml** for defining the result and action.

4. **Login.java** for defining login and logout logic.
5. **Profile.java** for checking if the user is logged in or not.
6. **View components** for the displaying results.

1) Add struts and servlet library

You need to add struts 2 and servlet library.(**servlet-api.jar** file)

2) Create index.jsp for input

This jsp page creates three links for login, logout and profile.

3) Define action and result in struts.xml

This xml file defines one package and 4 actions. Each action defines at least one result page.

For the loginprocess and logout actions, we are using the same action class but there invocation methods are different.

4) Create the action class for login and logout

This action class implements the Session Aware interface and overrides the `getSession` method to store the information in the session scope.

For logout, we are simply calling the **invalidate()** method of **SessionMap**.

5) Create the Profile class

This class gets the information from the session scope, if any information is found in the session scope with login name, it returns success otherwise false.

6) Create view components

There are many view components:

- login.jsp
- login-success.jsp
- login-error.jsp
- logout-success.jsp
- profile-success.jsp
- profile-error.jsp

view components for login

login.jsp

This page creates the login form.

login-success.jsp

This page prints the welcome message with the username.

login-error.jsp

This page displays the error message.

view components for logout

logout-success.jsp

This page simply displays the successfully logged out message.

view components for profile

profile-success.jsp

This page prints the welcome to profile message.

profile-error.jsp

This page prints the message to login first and includes the login.jsp page.