

Validação de Login Simples com Struts 2

Primeiramente vamos criar a nossa página **index.jsp**, para que possamos dar funcionalidades ajax para a nossa página utilizaremos na tag head o *theme="ajax"*, no nosso form usaremos a action Login e o seu método loginStatus, que iremos ver mais abaixo, colocamos também o atributo *validate* como *"true"* para que possamos fazer a validação dos campos. Usamos o atributo *required* como *"true"* nos campo de login e senha para que os mesmos sejam obrigatórios:

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<%@taglib prefix="s" uri="/struts-tags" %>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Login</title>
<s:head theme="ajax" />
</head>
<body>

    <s:form action="Login!loginStatus" validate="true" >
        <s:textfield name="username" label="Usuário" required="true"
theme="ajax" />
        <s:password name="password" label="Senha" required="true"
theme="ajax" />
        <s:submit value="Login" align="center" required="true" />
    </s:form>

</body>
</html>
```

Agora será criado a página **errorLogin.jsp**: que será usada quando o usuário e/ou senha estiverem incorretos, então redirecionaremos para esta página que é bem simples, só contém uma mensagem em vermelhoinformando que a senha e/ou usuário forem inválidos:

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
    <title>Erro em Login</title>
```

```

</head>
<body>
    <h1 align="center"><font color="red">ERRO LOGIN: Usuário ou senha
inválidos </font></h1>
</body>
</html>

```

E quando o login for válido, a aplicação redirecionará para a página **successLogin.jsp**, que apenas terá uma mensagem de sucesso:

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
    <title>Sucesso em Login</title>
</head>
<body>
    <h1 align="center"><font color="blue">Você realizou o login com
sucesso!!!</font></h1>
</body>
</html>

```

Agora criaremos a nossa action que será chamada de **LoginAction**, que como visto, será usada na página **index.jsp**. Nossa action herdará de **ActionSupport**, e apenas terá um método

para validar o login, como não estamos fazendo conexão com o banco de dados neste exemplo, para efeito de simplificação, consideraremos o usuário e senha corretos como *"admin"*,

então se o que for digitado nos campos de usuário e senha na página **index.jsp** forem *"admin"* e *"admin"* respectivamente o login será válido. É importante não se esquecer dos

métodos getters and setters, pois o Struts usa esses métodos para acesso dos atributos da action:

```

public class LoginAction extends ActionSupport {

    private String _username;
    private String _password;
    private String _status;

```

```

public LoginAction() {

}

public String loginStatus(){
    _status = "invalid";
    if(_username.equals("admin") && _password.equals("admin")){
        _status = "valid";
    }
    return _status;
}

public String getUsername() {
    return _username;
}

public void setUsername(String username) {
    _username = username;
}

public String getPassword() {
    return _password;
}

public void setPassword(String password) {
    _password = password;
}
}
}

```

Depois disto, configuraremos agora o nosso **struts.xml**, iremos mapear a nossa action (LoginAction), na tag `<action>` colocamos o nome da nossa Action como *Login*, e colocamos uma exclamação e um asterisco que indica que o struts vai ler todos os métodos da action, assim podemos usar qualquer método da action, no atributo *class* colocamos o caminho da nossa action. Adicionamos também dois resultados, a String *"invalid"* caso o login não seja válido que redirecionará para a página **errorLogin.jsp** e a String *"valid"* , caso seja válido que retornará para a página **successLogin.jsp**

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE struts PUBLIC
"-//Apache Software Foundation//DTD Struts Configuration 2.0//EN"
"http://struts.apache.org/dtds/struts-2.0.dtd">

<struts>
    <include file="struts-default.xml" />

    <package name="default" extends="struts-default">
        <action name="Login!*" class="cris.exemplo.login.struts2.LoginAction" >
            <result name="invalid">errorLogin.jsp</result>
            <result name="valid">successLogin.jsp</result>
        </action>
    </package>

```

```
</struts>
```

E por último o **LoginAction-validation.xml**, que será responsável pela validação dos campos, o nome deste arquivo deve ser igual ao nome da action correspondente precedido de *-validation*. Neste arquivo passamos o nome do field que vamos validar e o tipo, que no nosso caso é os campos *username* e *password*, o tipo dos dois são *requiredstring*, ou seja esses campos não poderão estar em branco. E passamos também a mensagem que será mostrado caso os campos não forem preenchidos.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE validators PUBLIC
"-//OpenSymphony Group//XWork Validator 1.0.2//EN"
"http://www.opensymphony.com/xwork/xwork-validator-1.0.2.dtd">
<validators>
  <field name="username">
    <field-validator type="requiredstring">
      <message>É necessário informar o usuário</message>
    </field-validator>
  </field>

  <field name="password">
    <field-validator type="requiredstring">
      <message>É necessário informar a senha</message>
    </field-validator>
  </field>
</validators>
```