

```

1  package org.wicket.sapporo.handson.session;
2
3  import org.apache.wicket.markup.html.WebPage;
4  import org.apache.wicket.markup.html.form.PasswordTextField;
5  import org.apache.wicket.markup.html.form.StatelessForm;
6  import org.apache.wicket.markup.html.form.TextField;
7  import org.apache.wicket.markup.html.panel.FeedbackPanel;
8  import org.apache.wicket.model.IModel;
9  import org.apache.wicket.model.Model;
10
11 /**
12  * ログインページの例。通常のForm, Linkでも構築できるが、ステートフル扱いになりSession Timeoutエラーなどが
13  * 生じる可能性があるので、StatelessFormとStatelessLinkを使ってステートレスなページにしている。
14  */
15 public class SignInPage extends WebPage {
16     private static final long serialVersionUID = 1L;
17
18     private IModel<String> userNameModel;
19     private IModel<String> passphraseModel;
20
21     /**
22      * コンストラクタ。
23      */
24     public SignInPage() {
25         userNameModel = new Model<>("");
26         passphraseModel = new Model<>("");
27
28         StatelessForm<Void> form = new StatelessForm<Void>("form") {
29             private static final long serialVersionUID = 1L;
30
31             @Override
32             protected void onSubmit() {
33                 super.onSubmit();
34                 // セッションにユーザ名を格納
35                 if (MySession.get().signIn(userNameModel.getObject(), passphraseModel.getObject())) {
36                     setResponsePage(new SecurePage());
37                 } else {
38                     error("IDもしくはパスワードが間違っています");
39                 }
40             }
41         };
42         add(form);
43
44         form.add(new FeedbackPanel("feedback"));
45
46         form.add(new TextField<>("userName", userNameModel));
47
48         // パスワード欄専用のテキストフィールド。
49         form.add(new PasswordTextField("passphrase", passphraseModel));
50     }
51 }

```