

SGSonSITE.OrderForms 101

How to insert your Servlets to the SGSonSITE JSP order forms.

INTRODUCTION

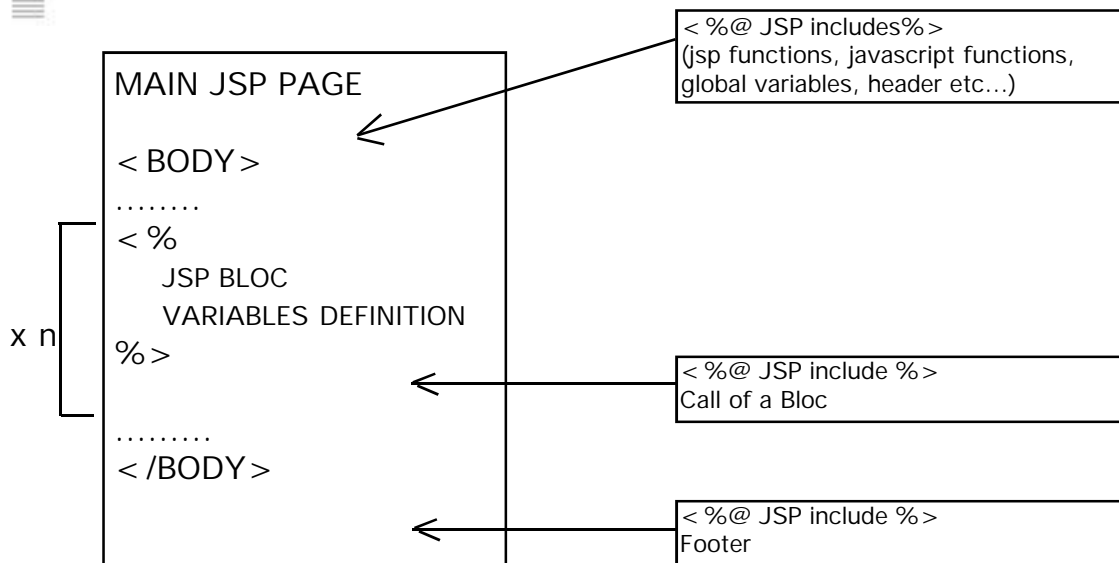
One order form (main page) is made of several "blocs".

A bloc is a JSP Page that is included in the main page with the JSP code:
`< %@ include page= "/mypath/mybloc.jsp" %>`

The main purpose of this is that by changing something in a bloc (ex: a field label) it will be changed in every page that calls it, hereby reducing risks of forgetting a page.

STRUCTURE OF A PAGE

This shema shows how a page is made :



GLOBAL VARIABLES:

This is a list of the global variables used, for an exact list, refer to the Global.jsp page.

```

GLO_Text_Font      = "Arial, Helvetica, sans-serif";
GLO_Text_Size      = "2";
GLO_Bg_Color       = "#FFFFFF";
GLO_Text_Color     = "#000000";
etc....
  
```

JSP BLOC VARIABLES

Before calling a bloc (including it), some variables must be defined. They are: Prefix of field names, Label of bloc, label of one or more fields, the JSP variables that will be called for prefilling and the variables for ErrorHandling

- Prefix of field names :

As some blocs can be called twice in the same page, all the fields are prefixed by a variable given before each bloc.

Example: the bloc asking for a company information (address etc..) is called twice in the SGSonSITE.Inspection order form. If it is called as the SGS customer information, the prefix is CU_; if it is called as the location information, the prefix is OtherParty.

- Label of Bloc : i.e: "Company Information", "Inspection Location" etc...

- Label of fields : with the example above, in the first case, the label for the first field would be "Company name" or "Location name"

- JSP Variables : For prefilling order forms, the value of the fields are set by JSP variables:

```
<INPUT maxLength=40 name="<%=Field_Prefix %>LegalEntityName"
size=30 Value="<%= LegalEntityName %>">
```

Before calling the bloc, all variables are created as empty. It is up to you to fill them with values that will show as default values on the browser, ex :

```
String legalEntityName = order.getCompany().getLegalEntityName();
```

- ErrorHandling : Some fields (mandatory ones mainly) have an associated Boolean value used for error handling. the name of the JSP variable is : < Name of Field> Error (LegalEntityNameError). All the booleans are also defined before calling a bloc. When they are set to true, an error will show on the JSP page on the browser. There is also a global variable, ErrorPage, it should be set to true if there is at least one error.

REPLACING COMMENTS BY CODE

BLOCS

In some blocs, there are JSP comments present waiting to be replaced by code. Those are mainly when some information is needed from the database to fill a drop down list (list of countries, list of certification bodies etc...) The JSP code should generate HTML code, refer to the HTML for FORMS appendix.

MAIN PAGES

Some information needs to be filled in, the variables (seen above) , the ACTION of the FORM tag (see appendix) and other info. Look for JSP comments (< %-- --&> tags) for clues about what to put and where.

HTML for FORMS Appendix

HTML is not a programming language but merely a text file with TAGS that the browser understands.

Each tag must be closed at some stage (for most of them). Example: the `` tag (bold text) and its counterpart `` (end of bold text).

A form uses the `<FORM>` `</FORM>`, the information given in the tag are the method of sending data (post or send) and the action (where to send the data on submission), the tag would look like this:

```
<FORM NAME="MyForm" METHOD="POST" ACTION="/path/MyServlet">
```

A form is made up of INPUTS. They are Text fields, List boxes, Drop Down lists, radio buttons etc...

The one you will mostly need is the SELECT type (drop down list) it usually looks like this:

```
<SELECT name="MyList" size="5">
    <OPTION VALUE="Value1" SELECTED> Option 1 </OPTION>
    <OPTION VALUE="Value2" > Option 2 </OPTION>
</SELECT>
```

note 1 : if size= x is present, a list is created, otherwise it is a drop down combo type.

note 2 : it doesnt matter what is put between `<OPTION>` and `</OPTION>`, the value sent to the server when the submit button is clicke is the value present in `VALUE= ""`

use the example above to generate the JSP code.

Example of modified SELECT field by pam:

```
<SELECT name="<%=Field_Prefix%>country_2C_ID" size="5">
<OPTION<% if (country_2C_ID.equals("")) out.print(" selected"); %>>
Select one</OPTION>
<%@ page import="com.sgs.common.framework.referencedata.*" %>
    <% try {
        Item[] countryList = CountryList.getInstance().getList();
        for(int i = 0; i < countryList.length; i++) {
            %>
            <OPTION value="<%= countryList[i].getId() %>"<% if
(country_2C_ID.equals(countryList[i].getId()) out.print(" selected");
%>><%= countryList[i].getText() %></OPTION>
            <%
        }
    }
    catch (Exception e) { }
    %>
</SELECT>
```