**JSP Custom Tags Quiz**

1. Rewrite the JSP page in the first demo (forEach) to use scripting instead of the JSTL forEach.

Without scripting (before):

<c:forEach var="student" items="${students}">

<tr>

<td>${student.name}</td>

<td>${student.age}</td>

</tr>

</c:forEach>

With scripting (after):

First you have to import the Student class putting this on top of the page:

<%@page import="net.endoedgar.Student"%>

Then replace the above code with the following:

<%

Student[] students = (Student[])pageContext.findAttribute("students");

for(Student student : students) {

%>

<tr>

<td><%=student.getName()%></td>

<td><%=student.getAge()%></td>

</tr>

<%

}

%>

1. How do custom tags relate to JSTL?

JSTL (Java Standard Tag Library) are actually custom tags which became part of the JSP language.

1. What is the role of the URI in the TLD and the taglib directive?

What you put in the URI of the TLD (Tag Library Descriptor) is what you should put in the uri of the taglib directive if you want to include custom tags from that TLD! It is basically a way to identify the TLD.

1. What is a tag handler class?

The tag handler class is where you put the behavior of a tag specified in the TLD. It has to override the method doTag of a parent class, in the lab example the SimpleTagSupport class, and on that method you have to write the HTML output to the JspWriter (getJspContext().getOut()).

1. What is the role of attribute setters in a tag handler class?

The attribute setters allows the container to get the parameters specified on the JSP which is using the tag and put them on those specified properties. You have to let the container know which parameters are accepted on the TLD before creating the setters for them on the class.

1. What is the role of the doTag() method in a tag handler class?

The role of doTag() method is to allow you to change the output of the custom tag using the JSPWriter. You can use your object attributes and show it in any way you need using HTML on the output.

1. What does the operation getJspContext().getOut().write(“Hi Bob”) do when called in a doTag() method?

This operation writes Hi Bob everytime the specified tag is written on a page.

1. What does the operation getJspBody().invoke(null) do when called in a doTag() method?

If you change the tag's body-content on the tld file to scriptless instead of empty, this allows your custom tag to have something between it. With it you can use getJspBody().invoke(null) to output what is between the tag beginning and ending to the user directly.