JSP Introduction Quiz

1. **What is the main value of Java Server Pages?**

Java Server Pages (JSP) are a technology for developing web pages that support dynamic content, allowing developers to insert java code in HTML pages by making use of special JSP tags, most of which start with . Eliminates the need to code complex output streams from the response object to produce a view.

1. **How are JSP pages related to servlets?**

* When a browser asks for a JSP, the JSP engine first checks to see whether it needs to compile the page. If the page has never been compiled, or if the JSP has been modified since it was last compiled, the JSP engine compiles the page
* The compilation process involves three steps:
  + Parsing the JSP.
  + Turning the JSP into a servlet.
  + Compiling the servlet

1. **How are JSP pages related to HTML?**

Jsp is html with a little java mixed in

* a little java code bracketed in <% %>, etc
* and lots of plain HTML

1. **Give an example of a JSP scriptlet and show how it will look in the JSP servlet .**

* JSP scriptlet

|  |
| --- |
| <html><body>  **<% int count=0; %>**  The page count is now:  </body></html> |

* How it will look in the JSP Servlet

|  |
| --- |
| public class basicCounter\_jsp extends SomeSpecialHttpServlet {  public void \_jspService(HttpServletRequest request,  HttpServletResponse response)throws java.io.IOException, ServletException {  PrintWriter out = response.getWriter();  response.setContentType(“text/html”);  out.write(“<html><body>”);  **int count=0;**  out.write(“The page count is now:”);  out.write(“</body></html>”);  }  } |

1. **Give an example of a JSP declaration and show how it will look in the JSP servlet.**

* JSP **declaration**

|  |
| --- |
| <html><body>  **<%! int count=0; %>**  </body></html> |

* How it will look in the JSP Servlet

|  |
| --- |
| public class basicCounter\_jsp extends SomeSpecialHttpServlet {  **int count=0;**  public void \_jspService(HttpServletRequest request, HttpServletResponse response)throws java.io.IOException {  } |

1. **Give an example of a JSP comment and show how it will look in the JSP servlet.**

* JSP **comment**

|  |
| --- |
| **<%-- This is a JSP comment --%>** |

* How it will look in the JSP Servlet

|  |
| --- |
| // **This is a JSP comment** |

1. **Give an example of a JSP expression and show how it will look in the JSP servlet.**

* JSP expression

|  |
| --- |
| **<%= ++count %>** |

* How it will look in the JSP Servlet

|  |
| --- |
| public class basicCounter\_jsp extends SomeSpecialHttpServlet {  public void \_jspService(HttpServletRequest request, HttpServletResponse response)throws java.io.IOException {  **out.print( ++count );**  } |

1. **Give an example of a JSP directive and show how it will look in the JSP servlet.**

* JSP directive

|  |
| --- |
| <%@ **page import**="**java.util.List**" %>  <**html**> <**head**>  <**title**>Title</**title**> </**head**> |

* How it will look in the JSP Servlet

|  |
| --- |
| **Import java.util.List;**  public class basicCounter\_jsp extends SomeSpecialHttpServlet {  public void \_jspService(HttpServletRequest request, HttpServletResponse response)throws java.io.IOException {  ...  } |

1. **Explain how an EL expression is evaluated.**

${something}

container evaluates this as follows

* checks page scope for an attribute named "something",
  + if found use it.
* otherwise check request scope for an attribute named "something",
  + if found use it.
* otherwise check session scope for an attribute named "something",
  + if found use it
* otherwise check application scope for an attribute named "something",
  + if found use it.
* otherwise ignore the expression.

1. **Explain how servlet attributes are involved in EL expressions.**

${firstThing

* if firstThing is not an implicit EL object, then search page, request, session and application scopes until attribute "firstThing" is found

${firstThing.secondThing}

* if firstThing is a bean then secondThing is a property of the bean
* if firstThing is a map then secondThing is a key of the map

${firstThing[secondThing]}

* if firstThing is a bean then secondThing is a property of the bean
* if firstThing is a map then secondThing is a key of the map
* if firstThing is a List then secondThing is an index into the List

1. **Explain how servlets and JSPs use request dispatch to interact.**

When you use JSP pages according to a Model 2 architecture, there is a servlet that acts as a controller (process of knowing) that sets attribute values based on computations and results from a business model (knower), then dispatches the request to the servlet generated by the JSP page (known). The JSP servlet then retrieves the attribute values and inserts them into the designated places in the HTML being sent to the browser