

# JSP Implicit Objects with Examples

This is the second post in the series of JSP Tutorials, you might want to check out earlier post about [JSP Tutorial for Beginners](#).

**JSP implicit objects** are created by container while translating JSP page to Servlet source to help developers. We can use these objects directly in **scriptlets** that goes in service method, however we can't use them in JSP Declaration because that code will go at class level.

We have 9 implicit objects that we can directly use in JSP page. Seven of them are declared as local variable at the start of `_jspService()` method whereas two of them are part of `_jspService()` method argument that we can use.

1. [out Object](#)
2. [request Object](#)
3. [response Object](#)
4. [config Object](#)
5. [application Object](#)
6. [session Object](#)
7. [pageContext Object](#)
8. [page Object](#)
9. [exception Object](#)
10. [JSP Implicit Objects Example](#)

## 1. out Object

JSP out implicit object is instance of `javax.servlet.jsp.JspWriter` implementation and it's used to output content to be sent in client response. This is one of the most used JSP implicit object and that's why we have JSP Expression to easily invoke `out.print()` method.

## 2. request Object

JSP request implicit object is instance of `javax.servlet.http.HttpServletRequest` implementation and it's one of the argument of JSP service method. We can use request object to get the request parameters, cookies, request attributes, session, header information and other details about client request.

## 3. response Object

JSP response implicit object is instance of `javax.servlet.http.HttpServletResponse`

implementation and comes as argument of service method. We can response object to set content type, character encoding, header information in response, adding cookies to response and redirecting the request to other resource.

## 4. config Object

JSP config implicit object is instance of `javax.servlet.ServletConfig` implementation and used to get the JSP init params configured in deployment descriptor.

## 5. application Object

JSP application implicit object is instance of `javax.servlet.ServletContext` implementation and it's used to get the context information and attributes in JSP. We can use it to get the `RequestDispatcher` object in JSP to forward the request to another resource or to include the response from another resource in the JSP.

## 6. session Object

JSP session implicit object is instance of `javax.servlet.http.HttpSession` implementation. Whenever we request a JSP page, container automatically creates a session for the JSP in the service method.

Since session management is heavy process, so if we don't want session to be created for JSP, we can use page directive to not create the session for JSP using `<%@ page session="false" %>`. This is very helpful when our login page or the index page is a JSP page and we don't need any user session there.

## 7. pageContext Object

JSP pageContext implicit object is instance of `javax.servlet.jsp.PageContext` [abstract class](#) implementation. We can use pageContext to get and set attributes with different scopes and to forward request to other resources. pageContext object also hold reference to other implicit object.

## 8. page Object

JSP page implicit object is instance of `java.lang.Object` class and represents the current JSP page. page object provide reference to the generated servlet class. This object is very rarely used.

## 9. exception Object

JSP exception implicit object is instance of `java.lang.Throwable` class and used to provide exception details in JSP error pages. We can't use this object in normal JSP pages and it's available only in JSP error pages.

## 10. JSP Implicit Objects Example

Here is a simple JSP page that's showing usage of JSP implicit objects. I am not providing exception object example, that will be covered in JSP Exception Handling post.

index.jsp

```
1  <%@ page language="java" contentType="text/html; charset=US-ASCII"
2     pageEncoding="US-ASCII"%>
3  <%@ page import="java.util.Date" %>
4  <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www
5  <html>
6  <head>
7  <meta http-equiv="Content-Type" content="text/html; charset=US-ASCII">
8  <title>Index JSP Page</title>
9  </head>
10 <body>
11 <!-- out object example --%>
12 <h4>Hi There</h4>
13 <strong>Current Time is</strong>: <% out.print(new Date()); %><br><br>
14
15 <!-- request object example --%>
16 <strong>Request User-Agent</strong>: <%=request.getHeader("User-Agent") %>
17
18 <!-- response object example --%>
19 <%response.addCookie(new Cookie("Test","Value")); %>
20
21 <!-- config object example --%>
22 <strong>User init param value</strong>:<%=config.getInitParameter("User")
23
24 <!-- application object example --%>
25 <strong>User context param value</strong>:<%=application.getInitParameter(
26
27 <!-- session object example --%>
28 <strong>User Session ID</strong>:<%=session.getId() %><br><br>
29
30 <!-- pageContext object example --%>
31 <% pageContext.setAttribute("Test", "Test Value"); %>
32 <strong>PageContext attribute</strong>: {Name="Test",Value="<%=pageContext
33
34 <!-- page object example --%>
35 <strong>Generated Servlet Name</strong>:<%=page.getClass().getName() %>
36
37 </body>
38 </html>
```

Deployment descriptor for the dynamic web project is:

web.xml

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http
3     <display-name>JSPImplicitObjects</display-name>
4     <welcome-file-list>
```

```

5      <welcome-file>index.jsp</welcome-file>
6  </welcome-file-list>
7
8      <context-param>
9      <param-name>User</param-name>
10     <param-value>Admin</param-value>
11 </context-param>
12
13     <servlet>
14     <servlet-name>home</servlet-name>
15     <jsp-file>/index.jsp</jsp-file>
16     <init-param>
17         <param-name>User</param-name>
18         <param-value>Pankaj</param-value>
19     </init-param>
20 </servlet>
21     <servlet-mapping>
22     <servlet-name>home</servlet-name>
23     <url-pattern>/home.do</url-pattern>
24     <url-pattern>/home.jsp</url-pattern>
25 </servlet-mapping>
26 </web-app>

```

When I access above JSP in browser, I get response like below image.



Let's look at the generated servlet class code for better understanding about these objects.

index\_jsp.java

```

1  /*
2   * Generated by the Jasper component of Apache Tomcat
3   * Version: Apache Tomcat/7.0.32
4   * Generated at: 2013-08-22 00:42:24 UTC
5   * Note: The last modified time of this file was set to
6   *       the last modified time of the source file after
7   *       generation to assist with modification tracking.
8   */
9  package org.apache.jsp;
10
11  import javax.servlet.*;
12  import javax.servlet.http.*;
13  import javax.servlet.jsp.*;
14  import java.util.Date;
15
16  public final class index_jsp extends org.apache.jasper.runtime.HttpJspBase
17      implements org.apache.jasper.runtime.JspSourceDependent {
18
19      private static final javax.servlet.jsp.JspFactory _jspxFactory =
20          javax.servlet.jsp.JspFactory.getDefaultFactory();

```

```

21
22 private static java.util.Map<java.lang.String,java.lang.Long> _jspx_dep
23
24 private javax.el.ExpressionFactory _el_expressionfactory;
25 private org.apache.tomcat.InstanceManager _jsp_instancemanager;
26
27 public java.util.Map<java.lang.String,java.lang.Long> getDependants() {
28     return _jspx_dep;
29 }
30
31 public void _jspInit() {
32     _el_expressionfactory = _jspxFactory.getJspApplicationContext(getServ
33     _jsp_instancemanager = org.apache.jasper.runtime.InstanceManagerFacto
34 }
35
36 public void _jspDestroy() {
37 }
38
39 public void _jspService(final javax.servlet.http.HttpServletRequest req
40     throws java.io.IOException, javax.servlet.ServletException {
41
42     final javax.servlet.jsp.PageContext pageContext;
43     javax.servlet.http.HttpSession session = null;
44     final javax.servlet.ServletContext application;
45     final javax.servlet.ServletConfig config;
46     javax.servlet.jsp.JspWriter out = null;
47     final java.lang.Object page = this;
48     javax.servlet.jsp.JspWriter _jspx_out = null;
49     javax.servlet.jsp.PageContext _jspx_page_context = null;
50
51
52     try {
53         response.setContentType("text/html; charset=US-ASCII");
54         pageContext = _jspxFactory.getPageContext(this, request, response,
55             null, true, 8192, true);
56         _jspx_page_context = pageContext;
57         application = pageContext.getServletContext();
58         config = pageContext.getServletConfig();
59         session = pageContext.getSession();
60         out = pageContext.getOut();
61         _jspx_out = out;
62
63         out.write("\n");
64         out.write("\n");
65         out.write("<!DOCTYPE html PUBLIC \"-//W3C//DTD HTML 4.01 Transitor
66         out.write("<html>\n");
67         out.write("<head>\n");
68         out.write("<meta http-equiv=\"Content-Type\" content=\"text/html; c
69         out.write("<title>Index JSP Page</title>\n");
70         out.write("</head>\n");
71         out.write("<body>\n");
72         out.write("\n");
73         out.write("<h4>Hi There</h4>\n");
74         out.write("<strong>Current Time is</strong>: ");
75 out.print(new Date());
76         out.write("<br><br>\n");
77         out.write("\n");
78         out.write("\n");
79         out.write("<strong>Request User-Agent</strong>: ");
80         out.print(request.getHeader("User-Agent") );
81         out.write("<br><br>\n");
82         out.write("\n");
83         out.write('\n');
84 response.addCookie(new Cookie("Test","Value"));
85         out.write('\n');

```

```

86     out.write("\n");
87     out.write("\n");
88     out.write("<strong>User init param value</strong>:");
89     out.print(config.getInitParameter("User") );
90     out.write("<br><br>\n");
91     out.write("\n");
92     out.write("\n");
93     out.write("<strong>User context param value</strong>:");
94     out.print(application.getInitParameter("User") );
95     out.write("<br><br>\n");
96     out.write("\n");
97     out.write("\n");
98     out.write("<strong>User Session ID</strong>:");
99     out.print(session.getId() );
100    out.write("<br><br>\n");
101    out.write("\n");
102    out.write('\n');
103    pageContext.setAttribute("Test", "Test Value");
104    out.write("\n");
105    out.write("<strong>PageContext attribute</strong>: {Name=\"Test\", \n");
106    out.print(pageContext.getAttribute("Test") );
107    out.write("\n")<br><br>\n");
108    out.write("\n");
109    out.write("\n");
110    out.write("<strong>Generated Servlet Name</strong>:");
111    out.print(page.getClass().getName() );
112    out.write("\n");
113    out.write("\n");
114    out.write("\n");
115    out.write("</body>\n");
116    out.write("</html>");
117    } catch (java.lang.Throwable t) {
118        if (!(t instanceof javax.servlet.jsp.SkipPageException)){
119            out = _jspx_out;
120            if (out != null && out.getBufferSize() != 0)
121                try { out.clearBuffer(); } catch (java.io.IOException e) {}
122            if (_jspx_page_context != null) _jspx_page_context.handlePageException(t);
123            else throw new ServletException(t);
124        }
125    } finally {
126        _jspxFactory.releasePageContext(_jspx_page_context);
127    }
128 }
129 }

```

Thats all for JSP implicit objects, in future posts you will see a lot of examples where they will be used.

---