

1. System Object in JSP:

Object	Description
Request	This is the HttpServletRequest object associated with the request.
Response	This is the HttpServletResponse object associated with the response to the client.
Out	This is the PrintWriter object used to send output to the client.
Session	This is the HttpSession object associated with the request.
Application	This is the ServletContext object associated with application context.
Config	This is the ServletConfig object associated with the page.
pageContext	This encapsulates use of server-specific features like higher performance JspWriters .
Page	This is simply a synonym for this , and is used to call the methods defined by the translated servlet class.
Exception	The Exception object allows the exception data to be accessed by designated JSP.

2. HTTP page status:

Following is a list of HTTP status codes and associated messages that might be returned from the Web Server:

Code:	Message:	Description:
100	Continue	Only a part of the request has been received by the server, but as long as it has not been rejected, the client should continue with the request
101	Switching Protocols	The server switches protocol.
200	OK	The request is OK
201	Created	The request is complete, and a new resource is created
202	Accepted	The request is accepted for processing, but the processing is not complete.
203	Non-authoritative Information	
204	No Content	
205	Reset Content	
206	Partial Content	
300	Multiple Choices	A link list. The user can select a link and go to that location. Maximum five addresses
301	Moved Permanently	The requested page has moved to a new url
302	Found	The requested page has moved temporarily to a new url
303	See Other	The requested page can be found under a different url
304	Not Modified	
305	Use Proxy	
306	<i>Unused</i>	This code was used in a previous version. It is no longer used, but the code is reserved.
307	Temporary Redirect	The requested page has moved temporarily to a new url.
400	Bad Request	The server did not understand the request

401	Unauthorized	The requested page needs a username and a password
402	Payment Required	<i>You can not use this code yet</i>
403	Forbidden	Access is forbidden to the requested page
404	Not Found	The server can not find the requested page.
405	Method Not Allowed	The method specified in the request is not allowed.
406	Not Acceptable	The server can only generate a response that is not accepted by the client.
407	Proxy Authentication Required	You must authenticate with a proxy server before this request can be served.
408	Request Timeout	The request took longer than the server was prepared to wait.
409	Conflict	The request could not be completed because of a conflict.
410	Gone	The requested page is no longer available.
411	Length Required	The "Content-Length" is not defined. The server will not accept the request without it.
412	Precondition Failed	The precondition given in the request evaluated to false by the server.
413	Request Entity Too Large	The server will not accept the request, because the request entity is too large.
414	Request-url Too Long	The server will not accept the request, because the url is too long. Occurs when you convert a "post" request to a "get" request with a long query information.
415	Unsupported Media Type	The server will not accept the request, because the media type is not supported.
417	Expectation Failed	
500	Internal Server Error	The request was not completed. The server met an unexpected condition
501	Not Implemented	The request was not completed. The server did not support the functionality required.
502	Bad Gateway	The request was not completed. The server received an invalid response from the upstream server
503	Service Unavailable	The request was not completed. The server is temporarily overloading or down.
504	Gateway Timeout	The gateway has timed out.
505	HTTP Version Not Supported	The server does not support the "http protocol" version.

3. JSTL tags

- Core Tags
- Formatting tags
- SQL tags
- XML tags
- JSTL Functions

Core Tags:

Tag	Description
<u><c:out ></u>	Like <%= ... >, but for expressions.
<u><c:set ></u>	Sets the result of an expression evaluation in a 'scope'
<u><c:remove ></u>	Removes a scoped variable (from a particular scope, if specified).
<u><c:catch></u>	Catches any Throwable that occurs in its body and optionally exposes it.
<u><c:if></u>	Simple conditional tag which evaluates its body if the supplied condition is true.
<u><c:choose></u>	Simple conditional tag that establishes a context for mutually exclusive conditional operations, marked by <when> and <otherwise>
<u><c:when></u>	Subtag of <choose> that includes its body if its condition evaluates to 'true'.
<u><c:otherwise ></u>	Subtag of <choose> that follows <when> tags and runs only if all of the prior conditions evaluated to 'false'.
<u><c:import></u>	Retrieves an absolute or relative URL and exposes its contents to either the page, a String in 'var', or a Reader in 'varReader'.
<u><c:forEach ></u>	The basic iteration tag, accepting many different collection types and supporting subsetting and other functionality .
<u><c:forEachTokens></u>	Iterates over tokens, separated by the supplied delimiters.
<u><c:param></u>	Adds a parameter to a containing 'import' tag's URL.
<u><c:redirect ></u>	Redirects to a new URL.
<u><c:url></u>	Creates a URL with optional query parameters

SQL tags:

<%@ taglib prefix="sql" uri="http://java.sun.com/jsp/jstl/sql" %>

Tag	Description
<u><sql:setDataSource></u>	Creates a simple DataSource suitable only for prototyping
<u><sql:query></u>	Executes the SQL query defined in its body or through the sql attribute.
<u><sql:update></u>	Executes the SQL update defined in its body or through the sql attribute.
<u><sql:param></u>	Sets a parameter in an SQL statement to the specified value.
<u><sql:dateParam></u>	Sets a parameter in an SQL statement to the specified java.util.Date value.
<u><sql:transaction ></u>	Provides nested database action elements with a shared Connection, set up to execute all statements as one transaction.

Formatting tags:

<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>

Tag	Description
<u><fmt:formatNumber></u>	To render numerical value with specific precision or format.
<u><fmt:parseNumber></u>	Parses the string representation of a number, currency, or percentage.
<u><fmt:formatDate></u>	Formats a date and/or time using the supplied styles and pattern
<u><fmt:parseDate></u>	Parses the string representation of a date and/or time
<u><fmt:bundle></u>	Loads a resource bundle to be used by its tag body.
<u><fmt:setLocale></u>	Stores the given locale in the locale configuration variable.
<u><fmt:setBundle></u>	Loads a resource bundle and stores it in the named scoped variable or the bundle configuration variable.
<u><fmt:timeZone></u>	Specifies the time zone for any time formatting or parsing actions nested in its body.
<u><fmt:setTimeZone></u>	Stores the given time zone in the time zone configuration variable
<u><fmt:message></u>	To display an internationalized message.
<u><fmt:requestEncoding></u>	Sets the request character encoding

XML tags:

<%@ taglib prefix="x" uri="http://java.sun.com/jsp/jstl/xml" %>

<u><x:out></u>	Like <%= ... >, but for XPath expressions.
<u><x:parse></u>	Use to parse XML data specified either via an attribute or in the tag body.
<u><x:set ></u>	Sets a variable to the value of an XPath expression.
<u><x:if ></u>	Evaluates a test XPath expression and if it is true, it processes its body. If the test condition is false, the body is ignored.
<u><x:forEach></u>	To loop over nodes in an XML document.
<u><x:choose></u>	Simple conditional tag that establishes a context for mutually exclusive conditional operations, marked by <when> and <otherwise>
<u><x:when ></u>	Subtag of <choose> that includes its body if its expression evaluates to 'true'
<u><x:otherwise ></u>	Subtag of <choose> that follows <when> tags and runs only if all of the prior conditions evaluated to 'false'
<u><x:transform ></u>	Applies an XSL transformation on a XML document
<u><x:param ></u>	Use along with the transform tag to set a parameter in the XSLT stylesheet

JSTL Functions:

<%@ taglib prefix="fn" uri="http://java.sun.com/jsp/jstl/functions" %>

Function	Description
<u>fn:contains()</u>	Tests if an input string contains the specified substring.
<u>fn:containsIgnoreCase()</u>	Tests if an input string contains the specified substring in a case insensitive way.
<u>fn:endsWith()</u>	Tests if an input string ends with the specified suffix.
<u>fn:escapeXml()</u>	Escapes characters that could be interpreted as XML markup.
<u>fn:indexOf()</u>	Returns the index withing a string of the first occurrence of a specified substring.
<u>fn:join()</u>	Joins all elements of an array into a string.
<u>fn:length()</u>	Returns the number of items in a collection, or the number of characters in a string.
<u>fn:replace()</u>	Returns a string resulting from replacing in an input string all occurrences with a given string.
<u>fn:split()</u>	Splits a string into an array of substrings.
<u>fn:startsWith()</u>	Tests if an input string starts with the specified prefix.
<u>fn:substring()</u>	Returns a subset of a string.
<u>fn:substringAfter()</u>	Returns a subset of a string following a specific substring.
<u>fn:substringBefore()</u>	Returns a subset of a string before a specific substring.
<u>fn:toLowerCase()</u>	Converts all of the characters of a string to lower case.
<u>fn:toUpperCase()</u>	Converts all of the characters of a string to upper case.
<u>fn:trim()</u>	Removes white spaces from both ends of a string.