



Custom Tags & JSTL

Presented by Hosein Zare
Twitter: @zare88r





Custom Tag

- یک custom Tag یک element jsp می باشد که توسط کاربران تعریف شده است.
- زمانی که یک jsp به یک Servlet تعریف میشوند تگ های custom توسط handler خود صدا زده شده و render میشوند
- یک tag handler کلاسی است که توضیح میدهد یک tag خاص چگونه render شود .

```
<ex:Hello />
```



نوشتن یک Custom Tag

- نوشتن یک Tag Handler کلاس
- ساخت یک فایل Tag Library Descriptor (TLD) برای تعریف tag
- ساخت jsp فایل و استفاده کردن از custom tag



Tag Handler

- برای نوشتن یک کلاس Tag Handler
 - پیاده سازی interface Tag یا ارث بری از TagSupport
 - پیاده سازی interface IterationTag برای اضافه شدن متود doAfterBody
 - پیاده سازی interface BodyTag یا ارث بری BodyTagSupport
 - پیاده سازی interface SimpleTag یا ارث بری SimpleTagSupport برای ساده سازی پیاده سازی با یک متود doTag



Tag Handler

```
package com.demisco.hr.web.tag;

import javax.servlet.jsp.tagext.*;
import javax.servlet.jsp.*;
import java.io.*;

public class HelloTag extends SimpleTagSupport {
    public void doTag() throws JspException, IOException {
        JspWriter out = getJspContext().getOut();
        out.println("Hello Custom Tag!");
    }
}
```



TLD

```
<taglib>  
<tlib-version>2.0</tlib-version>  
<short-name>demis</short-name>  
<uri>http://core.demisco.ir</uri>
```

```
<tag>  
  <name>Hello</name>  
  <tag-class> com.demisco.hr.web.tag.HelloTag</tag-class>  
  <body-content>empty</body-content>  
</tag>  
</taglib>
```



JSP File

```
<%@ taglib prefix="p" uri="http://core.demisco.ir" %>
```

```
<p>Hello/>
```



استفاده از Tag Files

- یک Tag File یک قطعه کوچکی از یک jsp می باشد که به شما امکان می دهد یک tag را به شیوه ی jsp پیاده سازی کنید به جای آنکه از Tag Handler های جاوایی استفاده کنید.

```
<%@tag description="A Tag for show Header" body-content="empty"%>
```

```
<%@attribute name="name" required="true" %>
```

```
<%@attribute name="greeting" required="true" %>
```

```
<h1><span style="font-size: larger;">${name} ! ${greeting}</span></h1>
```




تعریف در TLD

```
<tag-file>  
  <name>header</name>  
  <path>/WEB-INF/tag/header.tag</path>  
</tag-file>
```



تعریف در TLD

```
<tag-file>  
  <name>header</name>  
  <path>/WEB-INF/tag/header.tag</path>  
</tag-file>
```



Functions

- JSP EL به شما این اجازه را میدهد یک سری متود از پیش تعریف شده را صدا بزنید و از خروجی آنها استفاده کنید.
- Function ها با همان مکانیزم تعریف tag ها تعریف میشوند



Functions

```
public class TagFunctions {  
    public static double pow(double x, double y) {  
        return Math.pow(x, y);  
    }  
}
```

```
<function>  
    <name>pow</name>  
    <function-class>com.demisco.TagFunctions</function-class>  
    <function-signature>double pow(double,double)</function-signature>  
</function>
```

```
${p:pow(12.1, 4)}
```



JSTL (JSP Standard Tag Library)

• JSTL مجموعه ای از تگ ها میباشد که برای ایجاد سادگی در نوشتن jsp ها طراحی شده است

- Core Tags
- Function Tags
- Formatting Tags
- XML Tags
- SQL Tags



Core Tags

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

Tags	Description
c:out	It display the result of an expression, similar to the way <%=...%> tag work.
c:import	It Retrives relative or an absolute URL and display the contents to either a String in 'var',a Reader in 'varReader' or the page.
c:set	It sets the result of an expression under evaluation in a 'scope' variable.
c:remove	It is used for removing the specified scoped variable from a particular scope.
c:catch	It is used for Catches any Throwable exceptions that occurs in the body.
c:if	It is conditional tag used for testing the condition and display the body content only if the expression evaluates is true.
c:choose, c:when, c:otherwise	It is the simple conditional tag that includes its body content if the evaluated condition is true.
c:forEach	It is the basic iteration tag. It repeats the nested body content for fixed number of times or over collection.
c:forEachTokens	It iterates over tokens which is separated by the supplied delimiters.
c:param	It adds a parameter in a containing 'import' tag's URL.
c:redirect	It redirects the browser to a new URL and supports the context-relative URLs.
c:url	It creates a URL with optional query parameters.



Core Functions

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

JSTL Functions	Description
fn:contains()	It is used to test if an input string containing the specified substring in a program.
fn:containsIgnoreCase()	It is used to test if an input string contains the specified substring as a case insensitive way.
fn:endsWith()	It is used to test if an input string ends with the specified suffix.
fn:escapeXml()	It escapes the characters that would be interpreted as XML markup.
fn:indexOf()	It returns an index within a string of first occurrence of a specified substring.
fn:trim()	It removes the blank spaces from both the ends of a string.
fn:startsWith()	It is used for checking whether the given string is started with a particular string value.
fn:split()	It splits the string into an array of substrings.
fn:toLowerCase()	It converts all the characters of a string to lower case.
fn:toUpperCase()	It converts all the characters of a string to upper case.
fn:substring()	It returns the subset of a string according to the given start and end position.
fn:substringAfter()	It returns the subset of string after a specific substring.
fn:substringBefore()	It returns the subset of string before a specific substring.
fn:length()	It returns the number of characters inside a string, or the number of items in a collection.
fn:replace()	It replaces all the occurrence of a string with another string sequence.



Formatting

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

Formatting Tags	Descriptions
fmt:parseNumber	It is used to Parses the string representation of a currency, percentage or number.
fmt:timeZone	It specifies a parsing action nested in its body or the time zone for any time formatting.
fmt:formatNumber	It is used to format the numerical value with specific format or precision.
fmt:parseDate	It parses the string representation of a time and date.
fmt:bundle	It is used for creating the ResourceBundle objects which will be used by their tag body.
fmt:setTimeZone	It stores the time zone inside a time zone configuration variable.
fmt:setBundle	It loads the resource bundle and stores it in a bundle configuration variable or the named scoped variable.
fmt:message	It display an internationalized message.
fmt:formatDate	It formats the time and/or date using the supplied pattern and styles.