# Jstl

Jstl设计模式：

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

<%@ tagilib 编译指令 prefix=”自己定义” uri找到jstl包

<!-- EL表达式：不需要导入标签库

作用：获取四种作用域中的attribute

page request Session application 用scope会依次需找

内省机制：类的属性不是成员变量，而是get与set中之后的名称，所以尽量使用小写代表成员变量

$(user.Username);为空

$(user.username) 就是由于getUsername

1.属性名称如何的得到：

去掉get 和 set 方法以后的单词中如果第二个字母是小字，那么第一字母要发生变化（大写变小写）（变量：Username getUsername

那么属性名就是username）

如果第二字母是大写，那么第一个字母不变化（CPU）

2.如果作用域中有同名的属性，EL表达式默认是从小到大的寻找 page request Session application

request.getSession().setAttribute("user", 1111);

request.setAttribute("user", new UserPojo(3, "admin"));

得 到的：admin

3.如果指定到底从那种作用中获取相应的：

尽量用前如 ：requestScope.user (效率高）

jstl:标签

1.打印： out.print();

<c:out value="${user.username}"></c:out>

2.<c:set value=*"100"* var=*"num"* scope=*"session"*></c:set>

${num }

<!-- set 相当于request.setAttribute()

<c:remove var=""/>

remove 相当于request.removeAttribute()

-->

3.if else if else

<c:if test="${100>54}" var=*"flag"* scope=*"request"*>

hello

${flag }

</c:if>

其中when与otherwise只能在choose中使用

<c:choose>

<c:when test="${100 < 50}"></c:when>

<c:when test="${100<60 }"></c:when>

<c:otherwise>

llll<br>

</c:otherwise>

</c:choose>

4.迭代器：

<c:forEach items="${user }" var =*"n"* begin=*"2"* end=*"4"* step=*"2"*>

${n }</c:forEach> n值，从下标为2开始迭代，结束在4，步长为2

<c:forTokens items="${str }" delims=*","* var=*"n"*>类似于splic分割符号一样，n为值

${n }<br>

</c:forTokens>

request.getSession().setAttribute("user", 12325.0123);

<fmt:formatNumber value="${user }" pattern=*"###,###,###.00"*>

${user }

</fmt:formatNumber>

request.getSession().setAttribute("user", new Date());

<fmt:formatDate value="${user}" pattern=*"yyyy-MM-dd HH:mm:ss"*/>



函数库：

${fn:length("abccc") }

<servlet-mapping>

<servlet-name>GetProductIdServlet</servlet-name>

<url-pattern>/getProductId.do</url-pattern>

</servlet-mapping>