**假如我们获取了从数据库查出来的一个结果集，我们此处希望调用标签来将结果集以表格的形式输出来，此时我们的这个结果集又该如何传到标签处理类中呢？这时我们可以使用属性。具体看代码：**

**<%@ page contentType="text/html;charset=UTF-8" language="java" %>**

**<%@taglib uri="mytid" prefix="mytag"%>**

**<html>**

**<head>**

**<title></title>**

**</head>**

**<body>**

**<%**

**HashMap<String,Integer> maps = new HashMap<String, Integer>();**

**maps.put("李四",53);**

**maps.put("张三",23);**

**maps.put("walker",22);**

**pageContext.setAttribute("map",maps);**

**%>**

**<table>**

**<mytag:hello map="map"/>**

**</table>**

**</body>**

**</html>**

**<taglib xmlns="http://java.sun.com/xml/ns/j2ee"**

**xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"**

**xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee http://java.sun.com/xml/ns/j2ee/web-jsptaglibrary\_2\_0.xsd"**

**version="2.0">**

**<description>A tag library exercising SimpleTag handlers.</description>**

**<tlib-version>1.0</tlib-version>**

**<short-name>SimpleTagLibrary</short-name>**

**<uri>mytid</uri>**

**<tag>**

**<description>Outputs a colored tile</description>**

**<name>hello</name>**

**<tag-class>Test.MyTag</tag-class>**

**<body-content>empty</body-content>**

**<attribute>**

**<name>map</name>**

**<required>true</required>**

**<rtextprvalue>true</rtextprvalue> //可以接收运行时表达式的动态值**

**<fragment>true</fragment> //可以有jsp脚本,页面里有jsp脚本**

**</attribute>**

**</tag>**

**</taglib>**

**public class MyTag extends SimpleTagSupport {**

**private String map;**

**public String getMap(){**

**return this.map;**

**}**

**public void setMap(String map){**

**this.map = map;**

**}**

**@Override**

**public void doTag() throws JspException, IOException {**

**HashMap<String,Integer> maps = (HashMap<String,Integer>)(getJspContext().getAttribute(map));**

**Object[] array = maps.keySet().toArray();**

**for (String str : maps.keySet()){**

**getJspContext().getOut().write("<tr>");**

**getJspContext().getOut().write("<td>");**

**getJspContext().getOut().write(str);**

**getJspContext().getOut().write("</td>");**

**getJspContext().getOut().write("<td>");**

**getJspContext().getOut().write(""+maps.get(str));**

**getJspContext().getOut().write("</td>");**

**getJspContext().getOut().write("</tr>");**

**}**

**}**

**}**

**我们首先先从index.jsp页面看起，首先我们定义了一个HashMap用来存放一个简单的个人信息，键为姓名值为年龄。最后我们设置共享范围为当前page。**

**然后<mytag:hello map="map"/>，这里的map=“map”，第一个map是属性名，第二个只是一个字符串。**

**下面进入到tld页面看，这个页面基本没有什么改动，只是多了个attribute元素，attribute中有几个子元素，第一个是name指定这个属性的唯一标识，第二个required指定该属性是否是必须属性。**

**主要关心name这个元素。这个值和jsp页面调用标签时使用的属性名必须一样，并且这个属性值还必须和标签处理类中的私有属性名一样(这个私有属性必须是String类型)，这就是为了jsp页面的属性值能够自动的传入到标签处理类的属性中，我们看这个标签处理类.**

**这个类定义了私有属性map，和我们的tld文件中的属性名是一致的。getJspContext().getAttribute(map），首先是获得了调用该标签的jsp页面的pageContext，这就是方法getJspContext的返回值，因为我们在jsp页面设置了一个共享数据（map），于是我们同名名字获取该对象，这里的map就是我们的私有属性，他的值被自动赋值了，具体的值就是jsp页面传入的参数。**

**练习: 实现这个样的标签 <max num1="" num2="" /> 显示两个数的最大值**