

JSP

一 JSP 概述

1 什么是 JSP

Sun 公司提供的动态 WEB 资源开发技术，看起来向 html,但是 jsp 页面可以写 java 代码

2 JSP 由来

Servlet 适合处理业务逻辑，但不适合向外输出页面

Html 适合开发页面，但是不能展示

Jsp 结合了 html 和 servlet 的有点，既可以开发静态页面，也可书写 Java 代码

二 Jsp 语法

1 表达式

<%= 脚本表达式 %> 变量、表达式等
表达式的值会原样输出到当前位置

2 JSP 脚本片段

<% java 语句 %>

最终 java 语句会被翻译为普通的 java 代码执行

注意：

一个 jsp 脚本可以不完整，但是多个片段必须是完整的

3 Jsp 指令

<%@ 指令名称 若干属性%>

3.A 例：page 指令

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

用来声明当前 jsp 页面的基本属性，一般放在 jsp 最上面

3.A.1 属性:

Language: 当前 jsp 使用的开发语言
ContentType:数据编码格式
Import:导包

3.B include 指令

可以实现页面的包含效果

3.C Taglib 指令

用于导入 JSTL 标签库的指令

4 pageContext 隐式对象

代表当前 jsp 页面运行环境的对象

4.A 功能二:获取其它 8 大隐式对象

```
pageContext.getPage(); 获取 page  
pageContext.getRequest();//获取 request  
pageContext.getResponse();//获取 response  
pageContext.getServletConfig();//获取 config  
pageContext.getServletContext();//获取 application  
pageContext.getSession();//获取 session  
pageContext.getException();//获取 exception  
pageContext.getOut();//获取 out
```

9 大隐式对象可以在 JSP 中直接使用，但在 el 表达式中，其它 8 个只能通过 pageContext 获取

4.B 功能二: pageContext 域对象

4.B.1 生命周期:

访问 JSP 页面时创建，访问结束时销毁(4 大域对象中生命周期最短的)

4.B.2 作用范围:

当前 jsp 页面

三 Jsp 标签技术

在 JSP 中写入大量 JAVA 代码，会导致 JSP 页面混乱

所以 sun 公司提出了 jsp 标签技术，推荐使用 jsp 标签来代替 jsp 中的 java 代码，并推荐不要再 jsp 中出现任何一行 java 代码

1 EL 表达式

EL 表达式用来代替 JSP 页面中的 JSP 表达式

EL 只能获取数据，但不能遍历或者设置

1.A 格式: \${表达式}

EL 只能获取数据，但不能遍历或者设置

1.B 获取数据

可以获取常量、变量、数组中的数据、集合中的数据、对象的属性

注意: EL 表达式只能从四大作用域中获取数据，当作用域中有相同的 KEY 时，获取时按作用范围从小到大优先，也就是获取作用范围最小的那个 key 对应的值，和 JAVA 中局部变量和成员变量差不多一个道理

例如: 获取域中的 articles 数据，如果域中都有，则优先获取 pageContext 中的，同理
`${articles}`

1.C 执行运算

1.C.1 算术 JAVA 类似

1.C.2 关系 : JAVA 类似

1.C.3 逻辑 : JAVA 类似

1.C.4 三目 : JAVA 类似

1.C.5 Empty: 判断对象是否为空

`<c:if test="${ empty str}"> str 为空 </c:if>`

1.D 常用开发对象

EL 表达式内置了 11 个常用对象

1.D.1 pageContext

有了它相当于有了其它八大对象

1.D.2 pageScope: 代表 pageContext 域

例如获取 pageContext 中的 str 数据: `${pageContext.str}`

1.D.3 requestScope: 代表 request 域

1.D.4 sessionScope: 代表 session 域

1.D.5 applicationScope: 代表 ServletContext 域

1.D.6 注意:

pageScope、requestScope、sessionScope、applicationScope 代表四大作用域的对象，通常用来指定从哪个域中获取数据，并不能替代 request、session 等对象

例如:

四 . 1. B 中从域中获取数据: `${articles}`

有一个顺序，但可以通过以下对象指定从哪个域获取数据，

`${requestScope.articles}` 指定获取 request 域的 articles 数据

注意: 这四个对象与 request、response...的区别

1.D.7 param

代表请求参数的对象

例:

获取请求参数:

`${param.username}`

1.D.8 cookie

所有的 cookie 对象会保存到这个对象中，可以通过 cookie 的 name 获取 cookie 对象

`${cookie.nickname.value}`

2 JSTL 标签库

用来代替 JSP 页面的脚本片段

JSTL 中有很多子库，我们只使用核心标签库(core)

2.A 导入：

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

prefix 为别名，必须有

2.B <c:set>：向域中设置属性

例：设置 request 域中 username 的值

```
<c:set var="username" value="user" scope="request"></c:set>
```

2.C <c:if>：条件语句

```
<c:if test="${3>2}">3 大于 2</c:if>
```

```
<c:if test="${3<2}">3 小于 2</c:if>
```

最后会在网页显示 3 大于 2

2.D <c:forEach>：用来遍历

```
<c:forEach items="${articles}" var="ar">
    <p class="note">${ar.descri} ... </p>
</c:forEach>
```

生成 articles.size 个 p 标签