Jsp执行过程

访问jsp页面的过程

1. 访问jsp页面，tomcat扫描jsp文件，在%tomcat%/work把jsp文件翻译成java文件
2. Tomcat把java文件编译成class字节码文件
3. 服务器构造jsp类对象
4. 服务器调用类方法，返回结果到浏览器

第一次访问

走1 ，2， 3， 4

第n次访问

走4

注意：jsp文件被修改或jsp临时文件被删除，要重新走翻译和编译的过程

Servlet生命周期

1 ，构造函数

1. Init
2. Service
3. Destory

Jsp生命周期

1. 翻译
2. 编译
3. 构造函数
4. Init
5. Service
6. Destory

Jsp语法

Jsp表达式

<%=变量或表达式%> //向浏览器输出表达式或变量的值

Jsp脚本

<%

代码

%>

Jsp声明

<%! 变量或方法 %>

这里的变量或方法会翻译成成员变量和成员方法，不同于脚本的，脚本的是局部的

Jsp注释

<%!--\*\*\*\*\*\*--%>

jsp三大指令

Include指令

<%@include page=”/common/header” %>

注意：

原理是把被包含的文件和包含文件合并翻译成一个java源文件，这种叫静态包含（源码包含）

Page指令

告诉tomcat服务器如何翻译jsp文件

<%@page

language=”java”

Import=”java.util.\* , java.text.\* ” 导包属性，多个包逗号隔开

pageEncoding=”UTF-8” 使用什么编码编译jsp文件

contentType=”text/html , charst=UTF-8” 服务器发送给浏览器的格式编码

在开发工具中，只需要设置pageEncoding即可解决中文乱码问题

errorPage=”error.jsp” 指定要跳转的错误页面 , errorPage权限大于全局配置

isErrorPage=”true” 指定当前页面是否为错误页面

如果设置为true，可以使用exception内置对象，可以在页面中使用<%=exception.getMessage()%>

Session=”true” 是否开启session功能，设置为true，可以使用session内置对象

Buffer=”8kb” jsp页面缓冲区大小

isELIgnore 是否忽略EL表达式

%>

错误处理页面也可在xml文件全局配置

<error-page>

<error-code>500</error-code>

<location>500.jsp</location>

</error-page>

<error-page>

<error-code>404</error-code>

<location>404.jsp</location>

</error-page>

Servlet内置对象

9大内置对象

内置对象名 类型

Request HttpServletRequest

Response HttpServletResponse

Config ServletConfig

Application ServletContent

Session HttpSession

Exception Throwable

Page Object(this)

Out jspWriter

pageContext PageContext

Out内置对象

Out对象，是jspWriter类，相当于带缓存的PrintWriter

printWrirter类

Write() 直接向浏览器输出内容，无缓存

jspWriter类

Write() 向缓存池输出内容

以下情况输出缓存池内容

1. 缓存区满
2. 关闭缓存区
3. 手动刷新缓存区
4. Jsp加载完毕

pageContext对象

上下文对象，用于获取其他8个内置对象

使用场景：在自定义标签时，pageContext对象会频繁使用

本身就是一个域对象，默认保存到page域中

域常量

PageContent.PAGE\_SCOPE

PageContent.REQUEST\_SCOPE

PageContent.SESSION\_SCOPE

PageContent.APPLICATION\_SCOPE

PageContext.setAttribute(name , value , PageContext.REQUEST\_SCOPE) ; 可以指定保存到哪个域对象中

pageContext.getAttribute(name , PageContext.REQUEST\_SCOPE); 在哪个域中存的就在哪个域中取出来

findAttribute( name ); //自动搜索属性。一个域没找到就去下个域找

顺序 page域------request域-----session域-------application域

Page域只能在本页保存本页获取，其他页面不能获取