

# JSP 隐式对象

JSP隐式对象是JSP容器为每个页面提供的Java对象，开发者可以直接使用它们而不用显式声明。JSP隐式对象也被称为预定义变量。

JSP所支持的九大隐式对象：

对象	描述
request	<b>HttpServletRequest</b> 接口的实例
response	<b>HttpServletResponse</b> 接口的实例
out	<b>JspWriter</b> 类的实例，用于把结果输出至网页上
session	<b>HttpSession</b> 类的实例
application	<b>ServletContext</b> 类的实例，与应用上下文有关
config	<b>ServletConfig</b> 类的实例
pageContext	<b>PageContext</b> 类的实例，提供对JSP页面所有对象以及命名空间的访问
page	类似于Java类中的this关键字
Exception	<b>Exception</b> 类的对象，代表发生错误的JSP页面中对应的异常对象

## request对象

request对象是javax.servlet.http.HttpServletRequest 类的实例。每当客户端请求一个JSP页面时，JSP引擎就会制造一个新的request对象来代表这个请求。

request对象提供了一系列方法来获取HTTP头信息，cookies，HTTP方法等等。

## response对象

response对象是javax.servlet.http.HttpServletResponse类的实例。当服务器创建request对象时会同时创建用于响应这个客户端的response对象。

response对象也定义了处理HTTP头模块的接口。通过这个对象，开发者们可以添加新的cookies，时间戳，HTTP状态码等等。

## out对象

out对象是 javax.servlet.jsp.JspWriter 类的实例，用来在response对象中写入内容。

最初的JspWriter类对象根据页面是否有缓存来进行不同的实例化操作。可以在page指令中使用buffered='false'属性来轻松关闭缓存。

JspWriter类包含了大部分java.io.PrintWriter类中的方法。不过，JspWriter新增了一些专为处理缓存而设计的方法。还有就是，JspWriter类会抛出IOExceptions异常，而PrintWriter不会。

下表列出了我们将会用来输出boolean，char，int，double，String，object等类型数据的重要方法：

方法	描述
out.print(dataType dt)	输出Type类型的值
out.println(dataType dt)	输出Type类型的值然后换行
out.flush()	刷新输出流

## session对象

session对象是 javax.servlet.http.HttpSession 类的实例。和Java Servlets中的session对象有一样的行为。session对象用来跟踪在各个客户端请求间的会话。

## application对象

application对象直接包装了servlet的ServletContext类的对象，是javax.servlet.ServletContext 类的实例。这个对象在JSP页面的整个生命周期中都代表着这个JSP页面。这个对象在JSP页面初始化时被创建，随着jspDestroy()方法的调用而被移除。通过向application中添加属性，则所有组成您web应用的JSP文件都能访问到这些属性。

## config对象

config对象是 javax.servlet.ServletConfig 类的实例，直接包装了servlet的ServletConfig类的对象。这个对象允许开发者访问Servlet或者JSP引擎的初始化参数，比如文件路径等。以下是config对象的使用方法，不是很重要，所以不常用：

```
config.getServletName();
```

它返回包含在<servlet-name>元素中的servlet名字，注意，<servlet-name>元素在 WEB-INF\web.xml 文件中定义。

## pageContext 对象

pageContext对象是javax.servlet.jsp.PageContext 类的实例，用来代表整个JSP页面。这个对象主要用来访问页面信息，同时过滤掉大部分实现细节。这个对象存储了request对象和response对象的引用。application对象，config对象，session对象，out对象可以通过访问这个对象的属性来导出。pageContext对象也包含了传给JSP页面的指令信息，包括缓存信息，ErrorPage URL,页面scope等。

PageContext类定义了一些字段，包括PAGE\_SCOPE，REQUEST\_SCOPE，SESSION\_SCOPE，APPLICATION\_SCOPE。它也提供了40余种方法，有一半继承自javax.servlet.jsp.JspContext 类。

其中一个重要的方法就是removeAttribute()，它可接受一个或两个参数。比如，pageContext.removeAttribute("attrName")移除四个scope中相关属性，但是下面这种方法只移除特定scope中的相关属性：

```
pageContext.removeAttribute("attrName", PAGE_SCOPE);
```

## page 对象

这个对象就是页面实例的引用。它可以被看做是整个JSP页面的代表。

page 对象就是this对象的同义词。

## exception 对象

exception 对象包装了从先前页面中抛出的异常信息。它通常被用来产生对出错条件的适当响应。

[← JSP 动作元素](#)[JSP 客户端请求 →](#)[✎ 点我分享笔记](#)