

1. 内置对象的含义

在 Jsp 页面中不需要声明就可以在脚本和表达式使用的对象

特点

- 由Web 容器自动载入，不需要实例化
- 通过Web 容器实现和管理
- 在 Jsp 页面直接调用内置对象都是合法的

2. Input/Output 内置对象

2.1. request

- 表示客户端对服务端发送的请求，用于接受客户端通过 HTTP 协议传送给服务器端的数据
- 对象类型为 `javax.servlet.http.HttpServletRequest`
- 作用域为一次 request 请求
- 方法
 - `void setCharacterEncoding(String charset)`
 - 设置请求体参数的解码字符集
 - `String getParameter(String name)`
 - 根据参数名获得单一的参数值
 - `String[] getParameterValues(String name)`
 - 根据参数名获取一组参数值
 - `void setAttribute(String name, Object value)`
 - 以名/值的方式存储请求域属性
 - `Object getAttribute(String name)`
 - 根据属性名获取存储的对象数据

2.2. response

- 响应对象，表示服务端对客户端的响应，将 JSP 处理后的结果传回客户端
- 对象类型为 `javax.servlet.http.HttpServletResponse`
- 方法
 - `void setContentType(String name)`：设置响应内容的类型和字符编码
 - `void sendRedirect(String url)`：重定向到指定的 URL 资源

2.3. out

- 负责把处理结果输出的到客户端，控制管理输出的缓冲区（buffer）和输出流（output stream）向客户端输出数据
- 与 `PrintWriter` 功能相同
- 方法
 - 数据的输出
 - `print()/println()`
 - 缓冲区的处理
 - `void clearBuffer()`
 - `void flush()`
 - `void close()`
 - `int getBufferSize()`
 - `int getRemaining()`
- 向 out 对象的输出流中写入数据时，先存在缓冲区，满则会自动刷新输出
- page 指令的 `autoFlush` 和 `buffer` 属性配置缓冲区参数
 - `True` 自动刷新
 - `False` 非自动刷新

3. Context 内置对象

3.1. session 对象

- 表示浏览器与服务器会话的上下文环境，一次会话
- 一次会话的含义：从客户端浏览器连接服务器开始，到服务器端会话过期或用户主动退出，可以包含浏览器与服务器之间的多次请求与响应
- 方法
 - `void setAttribute(String name, Object value)`
 - `Object getAttribute(String name)`
 - `void invalidate()` : `session` 对象失效，释放所有的属性空间

3.2. application 对象

- 表示应用程序的上下文环境，表示当前应用程序运行环境，用来获取应用程序上下文环境中的信息
- `application` 对象在容器启动时实例化，在容器关闭时销毁。作用域为 Web 容器的生命周期
- 方法
 - `void setAttribute(String name, Object value)`
 - `Object getAttribute(String name)`
 - `void removeAttribute(String name)`

3.3. pageContext

- 表示 JSP 页面的上下文环境，作用范围为当前 JSP 页面
- 可以设置任意的作用域
- 获取内置对象的方法
 - `getRequest()` : 获取请求对象
 - `getResponse()` : 获取响应对象
 - `getSession()` : 获取当前 JSP 页面有联系的会话对象
- 存取域属性
 - `Object getAttribute(String name, int scope)` 默认页面域
 - `void setAttribute(String name, Object value, int scope)` 默认页面域
 - `void removeAttribute(String name, int scope)` 默认页面域
 - `scope` 常量
 - `PAGE_SCOPE = 1`
 - `REQUEST_SCOPE = 2`
 - `SESSION_SCOPE = 3`
 - `APPLICATION_SCOPE = 4`

4. Servlet 内置对象

4.1. page 对象

- 即 `this` 表示当前 jsp 本身，即被 jsp 翻译后的 Servlet
- 可以调用 Servlet 定义的方法

4.2. config 对象

- 页面配置对象，表示 JSP 页面翻译成 Servlet 页面后的 `ServletConfig` 对象
- 存放初始的数据结构

5. Error 内置对象

`exception` 对象

- `page` 指令的 `isErrorPage` 设置 `True`，才可以使用此对象

6. 四种作用域

- 页面域：生命周期是**页面执行期间**，存储在页面与的对象只对于它所在的页面是可以访问的
- 请求域：生命周期是**一次请求过程**，包括**请求被转发或被包含**。存储在请求域中的对象只有此次请求过程中才可以被访问。
- 会话域
 - 生命周期是某个**客户端与服务器所连接的时间**
 - 客户端在**第一次访问服务器时创建会话**，在**关闭浏览器或主动退出后会话结束**
 - 存储在会话域的对象在整个会话期间都可以访问
- 应用域：生命周期是**从服务器开始执行服务到服务器关闭**，存储在应用域中的对象在整个应用程序运行期间可以被所有 **JSP** 和 **Servlet** 共享访问