

- 1、在 tomcat 上发布 javaweb 应用时的默认目录是 webapps 目录。
- 2、javaweb 应用的部署描述符是指 web.xml 文件。
- 3、在 tomcat 中访问 helloapp 应用的 login.html 文件时，在浏览器地址栏应写的 url 为：http://localhost:8080/helloapp/login.html。
- 4、Servlet 容器启动每一个 web 应用时，都会为它创建一个唯一的 ServletContext 对象，该对象和 web 应用有相同的生命周期。
- 5、请求转发源组件的响应结果 不会 发送到客户端，包含 会 发送到客户端。（填“会”或“不会”）
- 6、jsp 中实现请求转发的标记为：<jsp:forward>。
- 7、在 JSP 中，request 内置对象代表请求消息，response 内置对象代表响应消息，session 内置对象代表会话。
- 8、如果使用第四类 jdbc 驱动器连接数据库，jar 文件的存放位置为“web 应用/WEB-INF/lib”。
- 9、Servlet 容器为每一个 HttpSession 对象分配一个唯一标识符，叫做 SessionID。
- 10、javaweb 在 mvc 设计模式下，javabean 是模型，jsp 是视图，servlet 是控制器。
- 11、JSP 的四种范围？范围依次增大 page、request、session、application。
- 12、web 组件的三种关联关系是 请求转发、请求重定向、包含。
- 13、在 struts2 应用中有一个上传文件的标签，它是 <s:file>。有一个进行迭代的标签，它是 <s:iterator>。
- 14、Struts2 请求经过 FilterDispatcher，它在里面 ActionMapper 会判断这个请求是否需要调用 Action。如果 ActionMapper 决定应该调用一个 Action，这个过滤器就把请求委托给 ActionProxy。
- 15、Struts2 框架主要有两个核心配置文件：struts.properties 和 struts.xml。
- 16、一个标准的 Properties 包含了系列的 key-value 对象。
- 17、Struts2 中的 struts.xml 文件主要负责管理应用中的 Action 映射，以及该 Action 包含 Result 定义等。
- 18、Struts.xml 内容主要包括：Action、Interceptor、Packages、Namespace 等
- 19、Struts.xml 中可以使用 <include> 把内容分到几个文件中。
- 20、Struts-default.xml 这个文件被包含在 Struts 2-core.jar 文件中。
- 21、Struts2 中，提供了两种 Action 驱动模式：Property-Driven（属性驱动）和 Model-Driven（模型驱动）。
- 22、Intercept 是拦截器的主要拦截方法，如果需要调用后续的 Action 或者拦截器，只要在该方法中调用 invocation.invoke() 方法即可，在该方法调用的前后可以插入 Action 调用前后拦截器需要做的方法。
- 23、Struts2 的标签库比先前版本有很大的进步，不再依赖于 JSP 标签，通用标签和 UI 标签，在使用的时候与使用 JSP 自定义标签相同第一步就是要导入标签库，Struts2 的引用方法语句 <%@ taglib prefix="s" uri="/struts-tags"%>。
- 24、Struts.properties 文件通常放在 web 应用的 WEB-INF/classes 路径下。



### 1、 jsp 有哪些内置对象?作用分别是什么?

- a) request 表示 HttpServletRequest 对象。它包含了有关浏览器请求的信息, 并且提供了几个用于获取 cookie, header, 和 session 数据的有用的方法。
- b) response 表示 HttpServletResponse 对象, 并提供了几个用于设置送回浏览器的响应的方法 (如 cookies, 头信息等)
- c) out 对象是 javax.jsp.JspWriter 的一个实例, 并提供了几个方法使你能用于向浏览器回送输出结果。
- d) pageContext 表示一个 javax.servlet.jsp.PageContext 对象。它是用于方便存取各种范围的名字空间、servlet 相关的对象的 API, 并且包装了通用的 servlet 相关功能的方法。
- e) session 表示一个请求的 javax.servlet.http.HttpSession 对象。Session 可以存贮用户的状态信息
- f) applicaton 表示一个 javax.servle.ServletContext 对象。这有助于查找有关 servlet 引擎和 servlet 环境的信息
- g) config 表示一个 javax.servlet.ServletConfig 对象。该对象用于存取 servlet 实例的初始化参数。
- h) page 表示从该页面产生的一个 servlet 实例

### 2、 jsp 标准动作

答: <jsp:include>: 在页面被请求的时候引入一个文件。<jsp:useBean>: 寻找或者实例化一个 JavaBean。<jsp:setProperty>: 设置 JavaBean 的属性。<jsp:getProperty>: 输出某个 JavaBean 的属性。<jsp:forward>: 把请求转到一个新的页面。<jsp:plugin>: 根据浏览器类型为 Java 插件生成 OBJECT 或 EMBED 标记。<jsp:param>: 用于传递参数值。

### 3、 forward 和 redirect 的区别

答: forward 是服务器请求资源, 服务器直接访问目标地址的 URL, 把那个 URL 的响应内容读取过来, 然后把这些内容再发给浏览器, 浏览器根本不知道服务器发送的内容是从哪儿来的, 所以它的地址栏中还是原来的地址。 redirect 就是服务端根据逻辑, 发送一个状态码, 告诉浏览器重新去请求那个地址, 一般来说浏览器会用刚才请求的所有参数重新请求, 所以 session, request 参数都可以获取。

### 4、 servlet 的生命周期

答: Servlet 是一种可以在 Servlet 容器中运行的组件, 那么理所当然就应该有一个从创建到销毁的过程, 这个过程我们可以称之为 Servlet 生命周期。Servlet 的生命周期可以分为加载、实例化、初始化、处理客户请求和卸载五个阶段, 体现在方法上主要是 init ()、service () 和 destroy () 三个方法。生命周期的具体说明如下:

- Servlet 容器完成加载 Servlet 类
- 实例化一个 Servlet 对象
- init () 方法完成初始化工作, 该方法由 Servlet 容器调用完成
- service () 方法处理客户端请求, 并返回响应结果
- destroy () 方法在 Servlet 容器卸载 Servlet 之前被调用, 释放一些资源

### 5、 JSP 和 Servlet 有哪些相同点和不同点, 他们之间的联系是什么?



JSP 技术是建立在 Servlet 技术基础之上的，本质上是 Servlet 的简易方式，更强调应用的外表表达。JSP 编译后是“类 servlet”，所有的 JSP 页面最终都要被转换成 Servlet 来运行。

目前，一个 web 应用往往由多个 servlet 和 jsp 组成。Servlet 和 JSP 最主要的不同点在于，Servlet 的应用逻辑是在 Java 文件中，并且完全从表示层中的 HTML 里分离开来。而 JSP 的情况是 Java 和 HTML 可以组合成一个扩展名为 .jsp 的文件。JSP 侧重于视图，负责显示工作完成；Servlet 主要用于控制逻辑，负责流程控制功能。

6、什么是 Servlet？Servlet 的运行机制是什么？

Servlet 是 SUN 公司利用 Java 语言实现类似 CGI 的服务器端编程技术。它是由容器管理的，基于 Java 技术的 Web 组件，其主要功能在于能与客户端进行交互，产生动态 Web 内容。Servlet 在服务器中的运行包括 4 个阶段：加载、初始化、调用及销毁。在容器启动的时候，Servlet 一般是不会加载的，Servlet 只会在第一次请求的时候被加载和实例化。Servlet 一旦被加载，一般不会从容器中删除，其生命周期知道应用服务器关闭或重新启动。此外，当容器做回收动作时，Servlet 也有可能被删除。

7、JSP 中 Model 2 的工作原理是什么？

JSP 的 Model2 实际上就是引入了 MVC 设计模式，采用 Servlet+JavaBean+JSP 进行开发。Model 2 采用 Servlet 作为控制器 (Controller) 负责接收客户端 Web 浏览器发送来的所有请求，并依据处理的不同结果，转发到对应的 JSP 页面 (Viewer) 实现在浏览器客户端的显示。通常在 Servlet 中只包含了控制逻辑和一些简单的处理逻辑，更加复杂的业务处理逻辑则借助特定的 JavaBean (Model) 具体实现。

8、JSP 中如何包含一个文件？

答：<%@include file=" 文件路径" %> <jsp:include page=" 文件路径" >

9、<jsp:forward>和 response.sendRedirect(); 有什么区别？

答：<jsp:forward>只能在本网站内跳转 response.sendRedirect 跳转到任何一个地址的页面

10、page、request、session、application 的作用范围都是什么？

答：page 只是当前页面内有效。request 是当前页面和请求的页面有效。session 是所有页面内有效，IE 关闭 session 消失。application 是所有页面内有效，服务器关闭 application 消失。

11、解决汉字乱码问题的常用方法？

答：request.setCharacterEncoding("GB2312");  
name=new String(name.getBytes("ISO8859\_1"), "gb2312");

12、GET 提交方式和 POST 提交方式有什么区别？

答：GET 方式提交地址栏上会显示提交内容，POST 是隐藏的。  
GET 方式提交大小限制为 2KB，POST 是无限大。

13、什么是 MVC 模式？

答：MVC 是 Model—View—Controller 的简写。模型是应用程序的主体部分。模型表示业务数据，或者业务逻辑。视图是应用程序中用户界面相关的部分，是用户看到并与之交互的界面。控制器 (controller) 控制器工作就是根据用户的输入，控制用户界面数据显示和



更新 model 对象状态。MVC 不仅实现了功能模块和显示模块的分离，同时它还提高了应用系统的可维护性、可扩展性、可移植性和组件的可复用性。

14、 会话的创建和使用

创建: `HttpSession session = request.getSession();`

使用: `session.setAttribute(String name, Object value);`

`String name = session.getAttribute(String name);`

15、 写出 JavaBean 的<jsp:setProperty>的四种形式，并说明他们之间的差别。

答: Jsp 提供了四种方法来设定 JavaBean 的属性值:

1).<jsp:setProperty name=" myBean" property=" \*" />

2).<jsp:setProperty name=" myBean" property=" myProperty " />

3)<jsp:setProperty name=" myBean" property=" myProperty" param=" ParamName" />

4)<jsp:setProperty name=" myBean" property=" myProperty" value=" MyValue" >

其中: name 就是<jsp:useBean>的 id, 因此 name 必须和 id 一致。

第一种写法: 由窗体传来的参数值, 通过自省机制, 设定所有的属性值;

第二种写法: 也是通过自省机制, 不过只设定 myProperty 的属性值;

第三种写法: 由窗体传来的参数名称为 ParamName 的值, 传给属性名称为 myProperty;

第四种写法: 它弹性最大, 它可以通过 value, 动态设定 JavaBean 的属性。

16、 请写出编写 Jsp 自定义标签的步骤。

答: 1)创建标记处理器 (Tag handler)类 2)创建标记库描述文件 (TLD) 3)应用程序部署描述符 (web.xml) 文件 4)在 Jsp 页面中的标记库声明

17、 什么是 application 对象, application 对象与 session 对象的区别?

答: 1) "application" 对象负责提供应用程序在服务器中运行时的一些全局信息, 常用的方法有 `getMimeType` 和 `getRealPath` 等。2)与 session 对象不同: 服务器启动后, 就产生 application 对象。当客户访问服务器上的 Jsp 时, Jsp 引擎就为该客户分配这个 application 对象。所有的客户的 application 对象都是同一个。即所有的客户共享这个内置的 application。

18、 什么是 JavaBean? JSP 中怎样调用 JavaBean?

JavaBean 是 SUN 公司推出的一个面向对象的编程接口规范, 它可以让用户设计可重用的程序块, 称 JavaBean 组件。编程中可以通过 JavaBean 的组合可以快速地生成新的应用程序。是一种用 JAVA 语言写成的可重用组件。1. JavaBean 是一个公共的 (public) 类。2. JavaBean 必须有一个无参构造。

JSP 调用 JavaBean

在 JSP 程序中使用 JavaBeans 组件必须应用<jsp:useBean>、<jsp:setProperty>和<jsp:getProperty>这三个标准标签。

<jsp:useBean >标签的作用是声明一个具有一定生存期以及一个唯一的 id 的 JavaBean 对象。JSP 通过 id 来识别 JavaBean, 通过 id.method 类似的语句来操作 JavaBean。

<jsp:setProperty>标签用于对 JavaBean 的属性赋值。<jsp:getProperty>标签用于获取 JavaBean 的属性值。