## 创建自定义标签有几种方式？如何使用自定义标签？

2种，可以继承TagSupport,BodyTagSupport或者实现Tag,BodyTag

在web.xml中配置taglib,在jsp中导入taglib 中标签的值

## 怎么配置过滤器(写出配置代码)？

<filter>

<filter-name>过滤器的名称</filter-name>

</filter-class>路径</filter-class>

</filter>

<filter-mapping>

<filter-name>过滤器的名称</filter-name>

<url-param>映射</url-param>

</filter-mapping>

## 3.怎么配置监听器(写出配置代码)？

<Listener>

<Listener-class>类路径</Listener-class>

</Listener>

## JSP中引入自定义标签有几种方式？请分别说明？

4种 TagSupport ,BodyTagSupport ,BodyTag,Tag

## 5.描述下 servlet运行的生命周期

1)只有一个Servlet对象（要点）

2)第一次请求的时候被初始化，只此一遍

3)初始化后先调用init方法，只此一遍

4)每个请求，调用一遍service -> service -> doGet/doPost。以多线程的方式运行

5)卸载前调用destroy方法

## 6.JSP有哪些内置对象?作用分别是什么?

1) pageContext表示页容器–>EL、标签、上传  
2) request服务器端取得客户端的信息：头信息、Cookie、请求参数、MVC设计模式  
3) response服务器端回应给客户端信息：Cookie、重定向  
4) session表示每一个用户，用于登录验证上  
5) application表示整个服务器，getRealPath()  
6) config去的初始化参数，初始化参数在web.xml中配置  
7) exception表示的是错误页的处理操作  
8) page如同this一样，表示整个JSP页面  
9) out输出，但是尽量使用表达式输出

## **7.java Servlet  API中forward(**) 与redirect()的区别？

前者仅是容器中控制权的转向，在客户端浏览器地址栏中不会显示出转向后的地址；后者则是完全的跳转，浏览器将会得到跳转的地址，并重新发送请求链接。这样，从浏览器的地址栏中可以看到跳转后的链接地址。所以，前者更加高效，在前者可以满足需要时，尽量使用forward()方法，并且，这样也有助于隐藏实际的链接。在有些情况下，比如，需要跳转到一个其它服务器上的资源，则必须使用sendRedirect()方法。

## 现在sql server 2005中有数据库ht，数据库中有表student,表中包括字段（id int，name varchar，age int ，classid ，int），编写一个JSP页面，要求使用JDBC桥接方式实现连接数据库，并从数据库的表中查询数据。

## 9.编写程序，通过过滤器实现解决乱码问题，同时写出配置代码？

<filter>

<filter-name>过滤器名</filter-name>

<filter-class> 类路径</filter-class>

</filter>

<filter-mapping>

<filter-name>对应上面过滤器名</filter-name>

<url-pattern>/映射名</url-pattern>

</filter-mapping>

## 数据源、连接池的原理？

当服务器启动时，会根据初始化的连接个数向数据库申请连接对象，并将申请到连接对象放到连接池中。当客户端要连接数据库时，会到连接池中查看是否有空闲的连接对象，如果有的话，获取一个连接对象给客户使用，如果没有空闲的连接对象，那么会查看连接池中的连接个数是否已经达到了最大的连接个数，如果没有达到，会重写申请连接对象，否则然后客户等待；客户用完之后，连接池自动回收该连接对象 ;(当客户没有关闭连接对象时，连接池会根据最大的空闲时间来自动回收连接对象)。

## 11.Servlet执行时一般实现哪几个方法？

public void init(ServletConfig config)

public ServletConfig getServletConfig()

public String getServletInfo()

public void service(ServletRequest request,ServletResponse response)

public void destroy()

## 12.请列出JSP四个会话跟踪技术，并分别进行说明各自的特点？

1.隐藏域：<input type="hidden" />适合不需要大量数据存储的会话应用。

2.url重写：可以通过在url后携带参数，和服务器请求一起发送，这些参数名字为键值对形式。

3.session：使用setAttribute(String str,Object obj)方法将对象捆绑到一个会话

4.cookie：一个Cookie是一个小的，已命名数据元素。服务器使用SET-Cookie头标将它作为HTTP响应的一部分传送到客户端，客户端被请求保存Cookie值，在对同一服务器的后续请求使用一个Cookie头标将之返回到服务器。与其它技术比较，Cookie的一个优点是在浏览器会话结束后，甚至在客户端计算机重启后它仍可以保留其值。

## 13.简单回答Servlet的工作过程

( 1 )客户机将请求发送到服务器。

（2）服务器上的Web容器实例化Servlet，并为Servlet进程创建线程。Servlet是在出现第一个请求时装入的，在服务器关闭之前不会卸载它。

（3）Web容器将请求信息发送到Servlet。

（4）Servlet创建一个响应，并将其返回到Web容器。Servlet使用客户端请求中的信息及服务器可以访问的其他信息资源如资源文件和数据库等来动态构造响应。

（5）Web容器将响应发回给客户机。

服务器关闭或Servlet空闲时间超过一定限度时，调用destroy方法退出。

## 14.jsp有哪些内置对象?作用分别是什么?

application对象代表应用程序上下文，它允许JSP页面与包括在同一应用程序中的任何Web组件共享信息。

Config对象允许将初始化数据传递给一个JSP页面

Exception对象含有只能由指定的JSP“错误处理页面”访问的异常数据。

Out对象代表提供输出流的访问

Page对象代表JSP页面对应的Servlet类实例

PageContext对象是Jsp页面本身的上下文，它提供唯一一组方法来管理具有不同作用域的属性

Request对象提供对Http请求数据的访问，同时还提供用于加入特定请求数据的上下文

Response对象允许直接访问HttpServletResponse对象

Session对象可能是状态管理上下文中使用最多的对话

## 15.JSP中动态INCLUDE与静态INCLUDE的区别？

动态include用jsp:include动作实现

<jsp:include page="included.jsp" flush="true" />它总是会检查所含文件中的变化，适合用于包含动态页面，并且可以带参数

静态include用include伪码实现,不会检查所含文件的变化，适用于包含静态页面

<%@ include file="included.htm" %>

## 16.请编写代码实现下面的功能？

1、创建一个Cookie。

2、给Cookie设置存活时间为3000秒。

3、将Cookie发送到客户端。

Cookie cookie = new Cookie("usr", "zhangsan");

cookie.setMaxAge(3);

response.addCookie(cookie);

## 请写出实现购物车的思路及原理？（要求采用会话来保存购车信息）

当一个用户点击购买一件商品时，将该商品的对象存放到一个list集合中  
2、之后就将这个集合存进session  
3、当要再购买一件商品时，同理，将该商品存进session中的list集合  
4、下单时，就将session中的list集合遍历存放到订单明细表中（订单表与订单明细表是一对多关系）  
把物品的信息先放到list里面，然后把list放到session里面，最后在页面取出session,然后遍历list

## 18.jsp有哪些动作?作用分别是什么?

<jsp:param>操作被用来以“名—值”队的形式为其他标签提供附加信息   
<jsp:include>操作允许在请求是在现成的JSP页面里包含静态或者动态资源   
<jsp:forward>操作允许将请求转发到另一个JSP、servlet或静态资源文件   
<jsp:useBean>标签用来在JSP页面中创建一个JavaBean实例并指定他的名字以及作用范围   
<jsp:setProperty>此操作和setBean一起协作，用来设置JavaBean的属性   
<jsp:getProperty>操作是对<jsp:setProperty>操作的补充，用来访问一个Bean的属性   
<jsp:plugin>用来产生客户端浏览器的特别标签，可以使用它来插入Applet和JavaBean   
<jsp:fallback>

## 19.SmartUpload实现上传下载的步骤.(简述,并用代码实现)

首先创建Html文件   
<html>   
<body bgcolor="white">   
<h1>jspsmartupload : Sample 1</h1>   
<hr>   
<form action="jsp/sample1.jsp" method="post" enctype="multipart/form-data">   
<input type="file" name="file1" size="50"><br>   
<input type="file" name="file2" size="50"><br>   
<input type="file" name="file3" size="50"><br>   
<input type="file" name="file4" size="50"><br>   
<input type="submit" value="Upload">   
</form>   
</body>   
</html>   
在创建JSP页面   
<%@ page language="java" import="com.jspsmart.upload.\*"   
pageEncoding="ISO-8859-1"%>   
<jsp:useBean id="mySmartUpload" scope="page"   
class="com.jspsmart.upload.SmartUpload"></jsp:useBean>   
<html>   
<body bgcolor="white">   
<h1>   
jspSmartUpload : Sample 1   
</h1>   
<hr>   
<%   
int count = 0;   
mySmartUpload.initialize(pageContext);   
mySmartUpload.setTotalMaxFileSize(1000000);   
mySmartUpload.upload();   
try {   
count = mySmartUpload.save("/upload");   
out.println(count + "file(s) upload.");   
} catch (Exception E {   
out.println(e.toString());   
}   
%>   
</body>   
</html>

## 20.请说明JSP内置对象的作用范围

a)Application Scope

b)Session Scope

c)Request Scope

d)Page Scope

## 描述Cookie和Session的作用，区别和各自的应用范围

Session用于保存每个用户的专用信息，每个客户端用户访问时，服务器都为每一个用户分配一个sessionid。Session中的信息保存在web服务器内容中，它的数据量可大可小

创建session的时候，服务器将生成一个唯一的session然后用它生成一个关闭浏览器就会失效的cookie，然后再将一个与这个sessionid关联的数据项加入散列表，当浏览器提交到服务器时，会通过sessionid去散列表中寻找属于该用户的session信息

Cookie用于保存客户浏览器请求服务器页面的请求信息，程序员也可以用他存储非敏感性的信息，信息保存的时间可以根据需要来设置，如果没有设置cookie失效日期，那它们仅保存到关闭浏览器程序为止

## 有页面login.jsp，表单提交name和password到LoginServlet，假设已经建立数据库连接对象conn，表名为user，字段名为name和password，请写出此LoginServlet，判断用户是否能够登录，并把用户登录信息记录到session中

## 23.请说明实现用application实现统计人数的步骤

1)、先从Applicaiton中获取到在线人数，再在这个人数的基础上加1

2)、将加后人数设回applicaiton中

4、在web.xml文件配置监听器

1. 在JSP中获取Application中的在线人数，显示到页面中

## 24.转发与重定向的区别？

1） 转发通过<jsp:forward/>来实现，重定向通过response.sendRedirect()来实现。

2） 转发可以共享request对象，而重定向会创建一个新的请求对象，所以不能共享request。

3） 转发是在服务端发生跳转，所以客户不知道，浏览器的地址栏不会发生变化，而 重定向是由客户端发出的，所有浏览器的地址栏会发生变化。

## 25.jsp乱码如何解决，有几种解决方案，请分别介绍？

1）、request.setCharacterEncoding(“编码(utf-8/GBK/GB2312)”);

2）、response.setCharacterEncoding(“编码” ) ;

3）、response.setContentType(“text/html ;charset=编码”)  ;

4）、<%@page pageEncoding= “编码” contentType= “text/html;charset=编码”%>

5）、表单采用post提交

6）、强制转换编码 new String(“内容”.getBytes(“iso-8859-1” ), “UTF-8” ) ;

1. 、过滤器

## 26.统计一下网站当前在线用户，请描述使用技术及设计思路？（1）网站不涉及登录；（2）Servlet容器支持Servlet2.3规范；（3）如果您有多种实现方法，请写出多种方法，并选择最佳的方法，说明原因。

1）、创建HttpSessionListener监听器,实现sessionCreated(SessionEvent event) ; sessionDestroyed() ;

2）、当有新会话时，会自动调用sessionCreated方法，在sessionCreated方法实现在线人数的增加：

<1>、先从Applicaiton中获取到在线人数，再在这个人数的基础上加1

<2>、将加后人数设回applicaiton中

3）、当有会话结束时，会自动调用sessionDestroyed（）方法，在该方法中实现人数减一

<1>、先从Applicaiton中获取到在线人数，再在这个人数的基础上减1

<2>、将加后人数设回applicaiton中

4）、在web.xml文件配置监听器

5）、在JSP中获取Application中的在线人数，显示到页面中

## 27.<% %> 与 <%! %>的区别？

1. <% %>可以声明局部变量，<%! %>可以声明全局变量。
2. <% %> 可以使用内置对象，<%! %>不可以使用内置对象,可以多线程共享数据。
3. <% %>不能声明方法，可以调用方法，<%! %>可以声明方法。

## 28.BS和CS的联系与区别？

1. BS是由客户端浏览器和web服务器端组成，CS是由专用的客户端程序和服务器程序组成.
2. BS通常是采用HTTP协议进行通信，而CS通常采用TCP或者UDP进行通信。
3. BS程序的业务处理都由服务器端完成，而CS程序的客户端可以负责部分业务处理，减轻了服务器端的负荷。