一、填空题（本大题共11小题，每空1分，共20分）

1、指令标识通常以(<%@)标记开始，以(%>)标记结束。

2、如果想在JSP页面中调用exception对象输出错误信息，需要将页面指令的(isErropage)属性设置为true。

3、页面指令的(import)属性可出现多次。

4、表达式用于向页面输出信息，其使用格式是以(<%=)标记开始，以(%>)标记结束。

5、在页面中通过声明标识声明的变量和方法的有效范围为(整个页面)，他们将成为JSP页面被转换成Java类后类中的(属性)和(方法)。

6、通常应用(<jsp:useBean>)动作标识可以在JSP页面中创建一个Bean实例，并且通过属性的设置可以将该实例存储到JSP中的指定范围内。

7、在JSP内置对象中，(request)对象是从客户端向服务器端发出请求，包括用户提交的信息以及客户端的一些信息，此对象的(getparameter)方法可以获取客户端表单中某输入框提交的信息。

8、在JSP内置对象中，(response)对象提供了设置HTTP响应报头的方法。

9、JSP内置对象的有效范围由小到大为(page)、(request)、(session)和(application)。

10、按功能JavaBean可以分为(可视化JavaBean)和(非可视化JavaBean)。

11、通常情况下通过java.lang.Class类的(forName(String className))方法加载要连接数据库的Driver类。

二、简答题（本大题共6小题，每题5分，共30分）

1、简述JSP的处理过程。

（1）客户端通过浏览器向服务器发出请求，该请求中包含了请求的资源的路径

（2）服务器根据被加载的客户端的请求加载被请求的JSP页面

（3）Web服务器中的JSP引擎把被加载的JSP页面转换成servlet

（4）JSP引擎把生成的JSP页面编译成class文件

（5）服务器执行这个class文件

（6）服务器把执行结果发送给浏览器显示

2、简述请求转发和请求重定向的区别。

进行请求转发时，存储在request对象中的信息会被保留并带到目标页面中；请求重定向是重新生成一个request请求，然后将该请求重定向到指定的URL，所以事先存储在request对象中的信息都不存在了

3、简述include指令标识和<jsp:include>动作标识的差异。

（1）inlcude指令通过file属性来确定被包含的页面，将file属性值看做一个实际存在的文件路径，所以该指令不支持任何表达式；<jsp:include>动作标识通过page属性确定被包含的页面，支持JSP表达式

（2）include指令包含的文件，内容会原封不动的插入到该指令的位置，与该指令合成进行编译，若改变了被包含文件，主文件要重新编译；<jsp:include>动作标识包含文件时，当该标识被执行时，程序会将请求转发到被包含的文件，并将执行结果输出到浏览器中，然后返回到包含文件继续执行后面的代码，JSP编译器对两个文件分别进行编译，被包含文件改动不会影响源文件。

4、简述内置对象session和application的区别。

Session对象和用户会话相关，，不同用户的session是完全不同的对象，而用户的application对象都是相同的一个对象，即共享这个对象

5、简述数据库连接池的实现原理。

（1）预先创建一定数量的连接，存放在连接池中

（2）当程序请求一个连接时，连接池是为该请求分配一个空闲连接，而不是去重新建立一个连接；当程序使用完连接后，该链接将重新回到连接池中，而不是直接将连接释放

（3）当连接池中的空闲连接数量低于下限时，连接池将根据管理机制追加创建一定数量的连接；当空线连接数量高于上限时，连接池将释放一定数量的连接

6、简述JSP内置对象、作用及有效范围。

1.application 有效范围：application

作用：用于保存所有应用程序中的公有数据，服务器启动并且自动创建application对像后，只要没关闭服务器，application对象将一直存在；

2．Config 有效范围：page

作用：将初始化数据传递给一个JSP页面；

3.exception 有效范围：page

作用：该对象含有只能由指定的JSP“错误处理页面”访问的异常数据；

4.out 有效范围：page

作用：该对象主要用来想客户端输出各种数据类型的内容，并且管理应用服务器上的输出缓冲区；

5.page 有效范围：page

作用：该对象代表JSP页面对应的servlet类实例；

6.pageContext 有效范围：page

作用：管理不同作用域的属性；

7.request 有效范围：request

作用：从客户端向服务器发出请求，包括用户提交的数据以及客户端的一些数据；

8.response 有效范围：page

作用：用于响应客户请求，想客户端输出信息；

9.session 有效范围：session

作用：用来保存服务器与一个客户端之间需要保存的数据；

四、编程题

1、在Ajax技术之中，最核心的技术就是XMLHttpRequest，为了提高程序的兼容性，请创建一个跨浏览器的XMLHttpRequest对象。（10分）

if(window.XMLHttpRequest){

http\_request=new XMLHttpRequest();

} else if(window.ActiveXObject){

try{

http\_request=new ActiveXObject(“Msxm12.XMLHTTP”);

} catch(e){

try{

http\_request=new ActiveXObject(“Microsoft.XMLHTTP”);

} catch (e) {}

}

}

2、编写一个简单的登录程序，要求登录页面index.jsp中输入用户名(username)、密码(password)，用户点击提交后，验证用户是否输入了用户名、密码。如果用户名为aaa,密码为aaa，则转到登录成功页面，否则显示index.jsp页面。（10分）

<body>

<form action="doLogon.jsp">

用户登录

用户名：<input type="text" name="userName">

<br>

密码：<input type="password" name="userPass">

<br>

<input type="submit" value="登录">

<input type="reset" value="重置">

<%

<%

String name=request.getParameter("userName");

if(name==null) name="";

String password=request.getParameter("userPass");

if(password==null) password="";

if(name=aaa&&password==aaa)

%>

%>

<jsp:forward page=”success.jsp”>

<%

else

%>

<jsp:forward page=”index.jsp”>

</form>

</body>

3、编写一个简单的Servlet程序，通过Servlet向浏览器输出文本信息“Servlet简单应用”。要求写出相应的Servlet类及配置文件。（15分）

public class print extends HttpServlet {

public void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html");

PrintWriter out = response.getWriter();

out.println("Servlet简单应用");

}

}

配置文件

<?xml version="1.0" encoding="UTF-8"?>

<web-app >

<servlet>

<servlet-name>MyServlet</servlet-name>

<servlet-class>print</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>MyServlet</servlet-name>

<urlpattern>/textServlet</urlpattern>

</servlet-mapping>

</web-app>

4、编写自定义标签，并调用该标签显示当前的系统日期和系统时间。要求写出标签类（继承自TagSupport）、标签库表述文件并写出JSP页面引用自定义的标签显示系统日期和系统时间。（注：写出关键代码即可）（15分）

package com;

import javax.servlet.jsp.\*;

import javax.servlet.jsp.tagext.\*;

inport java.util.\*;

public class ShowData extends TagSurpport{

public int doStartTag() throws JspException{

JspWriter out=pageContext.getOut();

try{

Data dt=new Data();

java.sql.Data data=new java.sql.Data(dt.getTime());

out.print(data);

}catch(Exception e){

System.out.println("显示系统日期时出现异常：")+e.getMessage());}

return(SKIP\_BODY);}}

标签库中表述文件：

<tag>

<description>显示当前日期</description>

<name>showData</name>

<tag-class>com.ShowData</tag-class>

<body-content>empty</body-content>

</tag>

在web.xml中加入自定义标签库的引用:

<taglib>

<taglib-uri>showDataUri</taglib-uri>

<taglib-location>/WEB-INF/showData.tid</taglib-location>

</taglib>

在页面中引用

<%@ page language="java" pageEncoding="GBK"%>

<%@ taglib uri="showDataUri" prefix="wghData"%>

<html>

<head><title>自定义标签的应用</title></head>

<body>今天是</wghData:showData/>

</body>

</html>