下边哪个是JSP指令标记(　　)

A. <%……%>　　B. <%!……%>

C. <%@……%>　　D. <%=……%>

JavaBean 的生命周期中，哪个是用来跟踪用户会话的(　　)

A. session　　B. request

C. page　　D. application

http 是一个(　　)协议

A.　无状态　　B.　有状态

C.　状态良好的　　D.　局域网

以下哪项陈述是错误的(　　)

A.　在 WEB 项目的共享数据范围内，application 是范围最广泛的

B.　当我们在一个 JSP 页面新开窗口时，新开窗口的页面也共享 session 范围内的数据

C.　当在 JSP 页面中通过<jsp:forward>指令将页面请求转发到的页面中，可以共享一个 page 范围内的数据

D.　当用户重新打开浏览器时，原 session 对象不再有效

JSP 页面经过编译之后，将创建一个(　　)。

A、applet　　B、servlet

C、application　　D、exe 文件

HTML 语言中
的作用是(　　)。

A、下划线　　B、换行

C、字体加粗　　D、斜体

哪个动作用于转向另一个页面。(　　)

A. next　　B. forward

C. include　　D. param

下列变量声明在(　　)范围内有效。

```
<%! Date dateTime;
     int countNum;
%>
```

A. 从定义开始处有效，客户之间不共享　　B. 在整个页面内有效，客户之间不共享

C. 在整个页面内有效，被多个客户共享　　D. 从定义开始处有效，被多个客户共享

以下有关表单的说明中，错误的是(　　)

A. 表单通常用于搜集用户信息。

B. 在 FORM 标记符中使用 action 属性指定表单处理程序的位置。

C. 表单中只能包含表单控件，而不能包含其他诸如图片之类的内容。

D. 在 FORM 标记符中使用 method 属性指定提交表单数据的方法。

include 指令用于在 JSP 页面静态插入一个文件，插入文件可以是 JSP 页面、HTML 网页、文本文件或一段 Java 代码，但必须保证插入后形成的文件是(　　)。

A. 是一个完整的 HTML 文件　　B. 是一个完整的 JSP 文件

C. 是一个完整的 TXT 文件　　D. 是一个完整的 Java 源文件

假定 login.getName() 返回类型为 java.lang.String ，给定 JSP 代码:

Welcome <%= login.getName() %>以下那个选项与此语句的功能相同?（　　）

A. Welocome <% out.print(login.getName());%>

B. Welocome <% Writer.print(login.getName());%>

C. Welocome <% response.out.print(login.getName());%>

D. Welocome <%r esponse.writer.print(login.getName());%>

下面哪一个不能作 JSP 的服务器(　　)

A. IBM WebSphere　　B. BEA　WebLogic

C. Tomcat　　D. pws

在下面哪个 web 应用目录中可以放置所需要的 class 文件?(　　)

A. /WEB-INF/lib　　B. /META-INF/lib

C. /classes 放置已经编译的类文件　　D. /WEB-INF/classes

当一个 Servlet 首次被请求的时候,服务器首先会调用(　　)方法.

A.　doGet　　B.　doPost

C.　doInit　　D.　init

当发布 Web 应用程序时，通常把 Web 应用程序的目录及文件放到 Tomcat 的（　）目录下。

A.　work　　B. temp

C. webapps　　D. conf

page 指令中的哪个属性可多次出现（　　）。

A、contentType　　B、extends

C、import 指令　　D、不存在这样的属性

当访问一个 Servlet 时，Servlet 中的方法执行顺序是（　　）。

A、init () service () destroy ()　　B、init () destroy () service ()

C、service () init () destroy ()　　D、service () destroy () init ()

在 JSP 中，重定向到另一个页面，以下哪项是正确的（　　）

A.　request . sendRedirect(“http://www.hncu.net”);

B.　request . sendRedirect();

C.　response . sendRedirect(“http://www.hncu.net”);

D.　response .sendRedirect();

下面哪项可以准确地获取请求页面的一个名称为 name 的文本框的输入。（　　）

A. request.getParameter (name)　　B. request.getParameter (” name”)

C. request.getParameterValues(name)

D. request.getParameterValues(“name”)

以下不属于JavaBean作用范围的是（　　）。

A.　request　　B. session

C. application　　D. scope

在JSP页面中使用<jsp:setProperty name="beanid"　property="bean的属性" value="字符串" />格式给Long类型的Bean属性赋值，会调用哪个数据类型转换方法。（　　）

A. Long.parseLong(String s)　　B. Integer.parseInt(Stirng s)

C. Double.parseDouble(String s)　　D. 不确定

下面是创建Statement接口并执行executeUpdate方法的代码片段：

```
conn=DriverManager.getConnection("jdbc:odbc:book","","");
stmt=conn.createStatement();
String strsql="insert into book values('TP003', 'ASP.NET','李','清华出版社',35)";
n=stmt.executeUpdate(strsql);
```

代码执行成功后n的值为　（　　）。

A. 1　　B. 0

C. -1　　D. 一个整数

在当前页面中包含 a.htm 的正确语句是（　　）

A. <%@ include=”a.htm”%>　　B. <jsp:include file=”a.htm” />

C. <%@ include page=”a.htm”%>　　D. <%@ include file=”a.htm”%>

在 JSP 程序中若想定义一个方法，必须将该方法放在下列哪种标记里　　（　　）

A.<%　　%>　　B. <%@　　%>

C.<%!　　%>　　D. <%--　　--%>

以下哪个不是 Servlet 的方法　　（　　）

A. destory()　　B. init()

C. post()　　D.service()

表单中的数据要提交到的处理文件由表单的哪个属性指定？（　　）

A.　method　　B. name

C. action　　D. 以上都不对

在 JSP 中可以通过下面哪个对象中的 getCookies（）方法获取 Cookie 中的数据。（　　）

A、response　　B、request

C、get　　D、read

在 JSP 页面中，能够完成输出操作的内置对象是 （　）

A.　out　　B.　response

C.　request　　D.　config

下列选项哪些是正确的 JSP 表达式语法格式（　　）。

A、<%String name="YXQ"%>　　B、<%=String name="您好"%>

C、<%="您好";%>　　D、<%="YXQ"%>

在编译 Servlet 或 JavaBean 时，我们使用的命令是（　　）。

A、javac　　B、java

C、Servlet　　D、以上都不是

给定TheBean类，假设还没有创建TheBean类的实例，以下哪些JSP动作语句能创建这个bean的一个新实例，并把它存储在请求作用域（　　）

A. <jsp :useBean name="myBean" type="com.example.TheBean"/>

B. <jsp :takeBean name="myBean" type="com.example.TheBean"/>

C. <jsp :useBean id="myBean" class="com.example.TheBean" scope="request"/>

D. <jsp :takeBean id="myBean" class="com.example.TheBean" scope="request"/>

关于session的使用，下列说话正确的是（ ）

A. 不同的用户打开同一个页面具有相同的session

B. 同一用户打开不同的页面窗口具有相同的session

C. 不能禁止session的使用

D. session永远不可能超时

假设在 helloapp 应用中有一个 Javabean 文件 HelloServlet，它位于 org.javathinker 包下，那么这个 bean 的 class 文件应该放在什么目录下？（　　）

A、helloapp/HelloServlet.class　　B、helloapp/WEB-INF/HelloServlet.class

C、helloapp/WEB-INF/classes/HelloServlet.class

D、helloapp/WEB-INF/classes/org/javathinker/HelloServlet.class

下面哪个方法可使 session 无效（　　）

A. session.setAttribute()　　B. session.getAttribute()

C. session.invalidate()　　D. session.removeAttribute()

在 JSP 页面中，正确引入 JavaBean 的是（　）

A. <%jsp: useBean id =”myBean” scope =”page” class=”pkg.MyBean” %>

B. <jsp: useBean name=”myBean” scope =”page” class=”pkg.MyBean” >

C. <jsp: useBean id =”myBean” scope =”page” class=”pkg.MyBean” />

D. <jsp: useBean name=”myBean” scope =”page” class=”pkg.MyBean” />

按作用域从大到小排列正确的是 （　　）

A. application page request response　　B. session page request application

C. public application session request　　D. application session request page

以下可用于获取 session 对象的属性 userid 的值是（　）

A. session. getAttribute(“userid”);

B. session. setAttribute(“userid”);

C. request. getParameter(userid);

D. session. getAttribute(userid);

下面哪项不是 useBean 动作可能的属性值(　　)。

A、contentType　　B、id

C、scope　　D、class

下面关于 JSP 作用域对象的说法错误的是 (　　)

A. request 对象可以得到请求中的参数

B. session 对象可以保存用户信息

C. application 对象可以被多个应用共享

D. 作用域范围从小到达是 request、session、application

Java 的数据类型分为两种，它们分别是_基本数据类型__和__引用数据类型__。

response.setHeader（“Refresh”，“5”）的含义是指页面刷新时间为 _5 秒___。

JSP 程序中要用到的变量或方法必须首先__声明变量或方法__。

MVC 是三层开发结构，这三个字母按顺序分别代表_模块___、___视图___、___控制___。
JSP 的 page 指令其 language 属性默认值__java____isErrorPage 属性的默认值是___
__true___；buffer 属性的默认值是___8kb____。
____application_____对象对于每个 Web 应用来说只有一个。
使用 page 指令引入 java.util.*的语句为 _<%@ page import=java.util.* %>____。

Word 文件的 MIME 类型是“application/msword”，Excel 文件的 MIME 类型是
____application/msexcel_____。

表单的提交方法包括__get____和___post___方法。

Session 对象中用来获得指定名字的属性的方法是__getAttribute()__方法。

在 jsp 中要建立与数据库的连接必须调用 DriverManager 类的__getConnection()__方法。

在编写 Servlet 时，需要继承__HttpServlet__类，在 Servlet 中声明 doGet()和 doPost()
需要
_HttpRequest____和__HttpResponse_ 类型的两个参数。

javax.servlet.Servlet 接口定义了三个用于 Servlet 生命周期的方法，它们是__
init()___、
_service()__、_destroy()___方法。

表单标记中的___action__属性用于指定处理表单数据程序 url 的地址。

使用 useBean 动作标记的时候 scope 属性有 4 种选项，作用范围由小到大是 page 和
___request___ 、____session____ 、____application_____。

在 JSP 中专门提供三个页面指令来和 JavaBean 交互，分别是_useBean__指令、__setProperty_____指令和_____getProperty___ 指令。

__session__对象封装了属于客户会话的所有信息。

在使用 JSP 对数据库进行操作时，Statement 类的__executeQuery()____方法用于执行 SQL 语言中的查询语句，___executeUpdate()_____方法,用于执行 SQL 语言中的插入、删除和修改语句。

Bean 是一个___公共__类，它必须有一个____无参__的构造方法。

JSP 指令元素主要有 3 种类型的指令，即___page____、____include_____和___taglib__。

___response_____对象的类型是 javax.servlet.ServletResponse 类的实例，JSP 引擎会根据客户端的请求信息建立一个默认的___response____对象。

JSP 中基本的元素类型有___动作____、__指令____、___脚本____三种。

JDBC 的主要任务是：___建立与数据库的连接_____、___向数据库发起查询请求_____、_处理数据库返回结果____。

javax.servlet.Servlet 接口定义了三个用于 Servlet 生命周期的方法，它们是_____init()_____、_____service()_____、_____destroy()_____ 方法。

Java 语言包含三种核心机制：_____Java 虚拟机_____、_____垃圾回收机制_____、_____代码安全机制_____。

数据库连接池的具体实施办法是哪些?

预先建立多个数据库连接保存在数据库连接池中，当程序访问数据库时，从连接池中取出空闲连接，访问结束后，再将连接放回连接池

简述 request 对象和 response 对象的作用。

Request 对象是从客户端向服务器发出请求，包括用户提交的信息以及客户端的一些信息。

Response 对象用于响应客户端请求，向客户端输出信息。

简述应用程序使用不可视 JavaBean 的主要步骤。

简述编译和使用 JavaBean 的主要步骤。

编写 Bean 类并使用 javac 命令编译 Bean 类

将编译好的 class 文件放入 WEB-INF\classes 文件夹中

使用<jsp:useBean>命令

分析下面的代码,写出 include.jsp 的运行结果。

include.jsp 代码:

```
<%@ page contentType="text/html;charset=GBK" %>
<html>
<body bgcolor="white" >
```

这里是显示结果：
<hr>
<%@ include file="abc.html"%> <p>
<%@ include file="mytxt.txt"%> <p>
<%@ include file="mycode.cod"%> <p>
</body>
</html>
abc.html 代码:

这是插入的 Html 文件

mytxt.txt 代码:
你好,这是 jsp 程序。
mycode.cod 代码：
<%
String s1="执行代码在这里显示";
out.print("   "+s1);
%>
运行结果：
这里是显示结果：
这是插入的 Html 文件
你好，这时 jsp 程序
执行代码在这里显示
分别写出以下带编号的各条语句的作用：

```
String url   = "jdbc:odbc:jspdb";
String query = "SELECT * FROM testTable1";
try {
    ① Class.forName ("sun.jdbc.odbc.JdbcOdbcDriver");
    ② Connection con = DriverManager.getConnection (url,"sa","");
    ③ Statement stmt = con.createStatement ();
    ④ ResultSet rs = stmt.executeQuery (query);
    rs.close();
    stmt.close();
    ⑤ con.close();
}
```

1. 加载驱动程序
2. 建立数据库连接
3. 建立执行 SQL 语句的实例
4. 获取查询结果集
5. 关闭数据库连接

应用 Date 函数读取系统当前时间，根据不同的时间段，在浏览器输出不同的问候语，例如上午 0~12 点之间输出“早上好”，同时把系统的年、月、日、小时、分、秒和星期输出到用户的浏览器。

```
<html>
<body>
<script type=”text/javascript”>
    var date = new Date();
    var year = date.getFullYear();
    var month = date.getMonth() + 1;
    var day = date.getDate();
    var hour = date.getHours();
    var minute = date.getMinutes();
    var second = date.getSeconds();
    if(hour >= 6 && hour <= 11) {
        document.writeln("早上好");
    } else if(hour == 12) {
        document.writeln("中午好");
    } else if(hour >= 13 && hour <= 17) {
        document.writeln("下午好");
    } else if(hour >= 18 && hour <= 23) {
        document.writeln("晚上好");
    }
    document.writeln(year + "-" + month + "-" + day + " " + hour + ":" +
minute + ":" + second);
</script>
</body>
</html>
```

编写两段代码，第一代码实现将自己的姓名添加至名为“name”的 cookie 中，第二段代码实现读出名为“name”的 cookie 中的值并输出。

```
String name = “qjm”;
Cookie c = new Cookie(“name”,name);
response.addCookie(c);

Cookie[] cookies = request.getCookies();
for(int i=0;i<cookies.length;i++) {
    if(cookies[i].getName().equals(“name”)) {
        out.print(cookies[i].getValue());
    }
}
```

编写程序 reg.htm 和 reg.jsp，做一用户注册界面，包括：用户名，年龄，性别。然后提交到 reg.jsp 进行注册检验，若用户名为 admin，就提示“欢迎你，管理员”，否则，显示“注册成功”并显示出注册信息。

reg.htm 文件

```
<html>
<body>
<form action=”reg.htm” method=”post”>
   用户名：<input type=”text” name=”username” /><br />
   年龄：<input type=”text” name=”userage” /><br />
   性别：<input type=”radio” name=”usersex” values=”男” />男
         <input type=”radio” name=”usersex” values=”女” />女<br />
   <input type=”submit” />
</form>
</body>
</html>
```

reg.jsp 文件

```
<%@ page contentType=”text/html;charset=utf-8” %>
<html>
<body>
<%
      String username = request.getParameter(“username”);
      String userage = request.getParameter(“userage”);
      String usersex = request.getParameter(“usersex”);
      if(“admin”.equals(username)) {
            out.println(“欢迎你，管理员”);
      } else {
            out.println(“注册成功！”);
            out.println(“用户名：” + username);
            out.println(“年龄：” + userage);
            out.println(“性别：” + usersex);
      }
%>
</body>
</html>
```

写出以下程序的输出结果：

```
<jsp:useBean id="stud" scope="page" class="test.Student" />
<jsp:setProperty name="stud" property="name" value="Zhang"/>
<%=stud.getName()%>
<%
stud.setAge("19");
%>
```


<jsp:getProperty name="stud" property="age" />

<% stud.setName(“Jack”); %>
<jsp:getProperty name="stud" property="name" />
<jsp:setProperty name="stud" property="age" value="20"/>

<%=stud.getAge()%>
Zhang
19
20

定义一个 JAVABEAN，名称为 TaxRate 其中含有二个简单属性，名称为 product 与 rate，数据类型自定，请编写一个 JSP 页面实现对上述 TaxRate 中二个简单属性的存取。

```
package org.tax;
public class TaxRate{
   private String product;
   private int rate;

   public TaxRate() {}

   public void setProduct(String product) {
      this.product = product;
   }

   public String getProduct() {
      return this.product;
   }

   public void setRate(int rate) {
      this.rate = rate;
   }

   public int getRate() {
      return this.rate;
   }
}

<%@ page contentType=”text/html;charset=utf-8” %>
<jsp: useBean id=”taxbean” scope=”application” class=”org.tax.TaxRate” />
<jsp: setProperty name=”taxbean” property=”product” value=”xxx” />
<jsp: setProperty name=”taxbean” property=”rate” value=”xxx” />

<jsp: getProperty name=”taxbean” property=”product” />
```

```
<jsp: getProperty name=”taxbean” property=”rate” />
```

写出学生系统登录界面 login.htm 的 HTML 代码（包括学号[文本类型]Id 和密码[密码类型]Pwd，并 JavaSript 进行学号和密码不为空验证）

```
<html>
<head>
<title>学生登录</title>
<script type=”text/javascript”>
function validate() {
    var username = document.getElementsById(“username”).value;
    var password = document.getElementsById(“password”).value;
    if(username == null || username == “”) {
        alert(“请输入学号”);
        return false;
    } else if (password == null || password == “”) {
        alert(“请输入密码”);
        return false;
    }
    return true;
}
</script>
</head>
<body>
<form action=”login.jsp” method=”post” onsubmit=”return validate();”>
    学号：<input type=”text” id=”username” name=”username” /><br />
    密码：<input type=”password” id=”password” name=”password” /><br />
    <input type=”submit” />
</form>
</body>
</html>
```

采用 JavaBean 技术把提交的信息初始化 StudentInf 的属性 Id、Pwd，并用 JavaBean 相关 JSP 动作输出学号和密码，并注明相关配置要求。

```
package org.student;

public class StudentInf {
    private String id;
    private String pwd;

    public StudentInf() {}
```

```
public void setId(String id) {
    this.id = id;
}

public String getId() {
    return this.id;
}

public void setPwd(String pwd) {
    public void setPwd(String pwd) {
        this.pwd = pwd;
    }

    public String getPwd() {
        return this.pwd;
    }
}

<%@ page contentType="text/html;charset=utf-8" %>
<html>
<body>
<%
    String username = request.getParameter("username");
    String password = request.getParameter("password");
%>
<jsp: useBean id="student" scope="page" class="org.student.StudentInf" >
student.setId(username);
student.setPwd(password);
<jsp: getProperty name="student" property="id" />
<jsp: getProperty name="student" property="pwd" />
</body>
</html>
```

利用学生登录系统 login.htm 的信息查询数据库 db 的表 student(stdId,Pwd)，如果存在登录成功，否则重新登录。

```
<%
    Class.for("org.mm.gjt.mysql.Driver");
    Connection conn =
DriverManager.getConnection("jdbc:mysql://localhost:3306/db",root,"");
    Statement stmt = conn.createStatement();
    RestultSet rs = stmt.executeQuery("select * from student where stdId = " +
username + " and Pwd = " + password);
    if(rs.next()) {
```

```
        out.print("登录成功！");
    } else {
        out.print("登录失败！");
    }
    rs.close();
    stmt.close();
    conn.close();
%>
```