

## JSP 期末考试复习题

1 ) 当 JSP 页面执行自定义标签，遇到结束标签时将调用()方法执行处理。( 选择一项 )

- a) doStartTag()
- b) doEndTag()
- c) doInitBody()
- d) doAfterBody()

8) 在 Servlet 过滤器的生命周期方法中，每当传递请求或响应时，web 容器会调用 ( ) 方法。( 选择一项 )

- a) init
- b) service
- c) doFilter
- d) destroy

10 ) 给定一个 Servlet 的代码片段如下：

```
Public void doGet(HttpServletRequest request, HttpServletResponse response)
                                throws ServletException, IOException{
```

```
_____
out.println("hi kitty!");
out.close();
}
```

运行次 Servlet 时输出如下：

hi kitty!

则应在此 Servlet 下划线处填充如下代码。( 选项一项 )

- a) PrintWriter out = response.getWriter();
- b) PrintWriter out = request.getWriter();
- c) OutputStream out = response.getOutputStream();
- d) OutputStream out = request.getWriter();

11 ) 给定如下 JSP 代码，假定在浏览器中输入 URL:http://localhost:8080/web/jsp1.jsp，

可以调用这个 JSP，那么这个 JSP 的输出是 ( )。(选项一项)

```
<%@ page contentType="text/html; charset=GBK" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<html>
<body>
<% int counter = 10; %>
<c:if test="${counter%2==1}">
<c:set var="isOdd" value="true"></c:set>
</c:if>
<c:choose>
<c:when test="${isOdd==true}">it's an odd </c:when>
<c:otherwise>it's an even </c:otherwise>
</c:choose>
</body>
</html>
```

- a) 一个 HTML 页面，页面上显示 it's an odd
- b) 一个 HTML 页面，页面上显示 it's an even
- c) 一个空白的 HTML 页面
- d) 错误信息

12) 给定 java 程序 User.java,代码如下:

```
package user;

public class User{

    private String password;
```

```
private String loginName;

public String getLoginName(){

    return loginName;

}

public String getPassword(){

    return password;

}

public void setLoginName(String loginName){

    this.loginName=loginName;

}

public void setPassword(String password){

    this.password=password;

}

}
```

给定 user.jsp,代码片断如下:

\_\_\_\_\_

loginName: <jsp:getProperty name="user" property="loginName"/>

password: <jsp:getProperty name="user" property="password"/>

在浏览器上输入 URL:http://localhost/user.jsp?loginName=sa&password=sa 并回车

要使浏览器上显示:

loginName: sa      password: sa

在 user.jsp 中的下划线处应填入以下()代码. (选择一项)

a) <jsp:useBean id="user" class="user.User" scope="page"/>



<jsp:setProperty name="user" property="loginName" value="\*" />

<jsp:setProperty name="user" property="password" value="\*" />

b) <jsp:useBean id="user" class="user.User" scope="application" />

<jsp:setProperty name="user" property="loginName" value="\*" />

<jsp:setProperty name="user" property="password" value="\*" />

c) <jsp:useBean id="user" class="user.User" scope=" application" />

<jsp:setProperty name="user" property="\*" />

d) <jsp:useBean id="user" class="user.User" scope=" page" />

<jsp:setProperty name="user" property="\*" />

13)某 JSP 程序中声明使用 javaBean 的语句如下:

<jsp:useBean id="user" class="mypackage.User" scope="pge" />

要取出该 javaBean 的 loginName 属性值,以下语句正确的是(). (选择两项)

a) <jsp:getProperty name="user" property="loginName" />

b) <jsp:getProperty id="user" property="loginName" />

c) <%=user.getLoginName()%>

d) <%=user.getProperty("loginName")%>

15)给顶一个JSP程序源代码, 如下:

<jsp:include page="two.jsp" flush="true">

<jsp:param name="location" value="beijing" />

</jsp:include>

在two.jsp中使用 ( ) 代码片段可以输出参数location的值。

a) <jsp:getParam name="location">

b) <jsp:getParameter name="location">

- c) `<%=request.getAttribute("location")%>`
- d) `<%=request.getParameter("location")%>`

17) 在J2EE中，test.jsp文件如下：

```
<html>

<%{%>

<jsp:useBean    id="buffer"    scope="page"    type="java.lang.StringBuffer"/>

<%buffer.append("ABC");%>

<%}%> buffer    is    <%=buffer%>

</html>
```

运行时，将发生（ ）。 （ 选择一项 ）

- a) 转译期错误
  - b) 运行期错误
  - c) 运行后，浏览器上显示：buffer is null
  - d) 运行后，浏览器上显示：buffer is ABC
- 18) 如 JSP 页面执行时遇到自定义标签，doStartTag()方法将被调用（ ）次。（ 选择一项 ）
- a) 1
  - b) 1 或多
  - c) 0 或 1
  - d) 0 或多

19) 给定 include1.jsp 文件代码片断，如下：

```
<% pageContext.setAttribute("User","HAHA");%>
```

\_\_\_\_\_//此处填写代码

给定 include2.jsp 文件代码片断，如下：

```
<%=pageContext.getAttribute("User")%>
```

要求运行 include1.jsp 时，浏览器上输出：HAHA

要满足以上条件，jsp1.jsp 中下划线处应填入以下（ ）语句。（ 选择一项 ）

- a) `<jsp:include page="include2.jsp" flush="true"/>`
- b) `<%@ include file="include2.jsp"%>`
- c) `<jsp:forword page="include2.jsp"/>`
- d) `<% response.sendRedirect("include2.jsp");%>`

20) 在 J2EE 中，在 web.xml 中，有如下代码：

```
<session-config>
    <session-timeout>30</session-timeout>
</session-config>
```

上述代码定义了默认的会话超时时长，时长为 30 ( )。 ( 选择一项 )

- a) 毫秒
- b) 秒
- c) 分钟
- d) 小时

21) 在 inc.jsp 中包含数行代码与一张图片，现在要创建一个 home.jsp，每次浏览 home.jsp 时都能够以最快的速度显示 inc.jsp 中的所有内容以及 home.jsp 自身要显示的内容，应该在 home.jsp 中使用的语句是( )。 ( 选择一项 )

- a) `<%@include file = "inc.jsp"%>`
- b) `<%@include page="inc.jsp" flush="true">`
- c) `<jsp:include file = "inc.jsp" flush = "false">`
- d) `<jsp:forward page = "inc.jsp"/>`

22) 给定一个 JSP 程序源码如下：

```
<jsp:include page = "two.jsp" flush = "true">
<jsp:param name = "location" value = "beijing"/>
</jsp:include>
```

在 two.jsp 中加入 ( ) 代码片断可以输出参数 location 的值。 ( 选择一项 )



a) `<jsp:getParam name = "location">`      b) `<jsp:getParameter name = "location">`

c) `<% = request.getAttribute("location")%>`      d) `<% = request.getParameter("location")%>`

23 ) 使用自定义标签时，要为 JSP 引擎提供有关自定义标签的相关信息，下列信息描述语句正确的是 (      )。(选择一项)

a) `<tag>`  
    `<name>hello</name>`  
    `<tag-class>com.tagdemo.HelloTag</tag-class>`  
    `</tag>`

b) `<taglib>`  
    `<name>hello</name>`  
    `<tag-class>com.tagdemo.HelloTag</tag-class>`  
    `</taglib>`

c) `<tag-info>`  
    `<name>hello</name>`  
    `<tagclass>com.tagdemo.HelloTag</tagclass>`  
    `</tag-info>`

d) `<tag-desc>`  
    `<name>hello</name>`  
    `<tagclass>com.tagdemo.HelloTag</tagclass>`  
    `</tag-desc>`

24) 要设置某个 JSP 页面为错误处理页面，以下 page 指令正确的是 (      )。(选择一项)

- a) `<%@ page errorPage="true"%>`                      b) `<%@ page`

`isErrorPage="true"%>`

- c) `<%@ page extends="javax.servlet.jsp.JspErrorPage"%>`

- d) `<%@ page info="error"%>`

25) JavaMail 是一组用于发送和接收邮件消息的 API。发送邮件使用 ( ) 协议，接收邮件使用 ( ) 协议。 (选择一项)

- a) POP3 SMTP

- b) POP3 TCP

- c) SMTP TCP

- d) SMTP POP3

29) 在 J2EE 实现企业级应用开发中, ( ) 是描述标记库的 XML 文档。 (选择一项)

- a) TLD 文件

- b) DTD 文件

- c) WAR 文件

- d) EAR 文件

30) 给定 JSP 程序源码如下, 可以在下划线处插入并且能够正确输出 WELCOME ,JACK 的语句是 ( )。 (选择二项)

```
<html>
```

```
<body>
```

```
<% String name="JACK"; %>
```

```
WELCOME , _____
```

```
</body></html>
```

- a) name

- b) `<%=name%>`

- c) `out.println(name);`

- d) `<% out.println(name); %>`

31) 在 J2EE 中, 重定向到另一个页面, 以下 ( ) 语句是正确的。 (选择一项)

- a) `request.sendRedirect("http://www.jb-aptech.com.cn");`

- b) `request.sendRedirect();`



c) response.sendRedirect("http://www.jb-aptech.com.cn");

d) response.sendRedirect();

33) 在 J2EE 中，使用 Servlet 过滤器时，可以在 web.xml 文件的 ( ) 元素中包括 <init-param> 元素。 (选择一项)

a) <filter>

b) <filter-mapping>

c) <filter-name>

d) <filter-class>

34) Servlet 中，使用 ( ) 接口中定义的 ( ) 方法来处理客户端发出的表单数据请求 (选一项)

a) HttpServlet doHead

b) HttpServlet doPost

c) ServletRequest doGet

d) ServletRequest doPost

35) 给定 JSP 程序源码如下，该 JSP 运行后输出的结果是 ( )。 (选择一项)

```
<html>
```

```
<% int count =1;%> Count: <% ++count; %>
```

```
</html>
```

a) Count:1

b) Count:2

c) 1:2

d) Count:

36) 给定如下 Servlet 代码，假定在浏览器中输入 URL：

http://localhost:8080/servlet/HelloServlet，可以调用这个 servlet，那么这个 Servlet 的输出是 ( )。

```
import java.io.*;
```

```
import javax.servlet.*;
```

```
import javax.servlet.http.*;
```

```
public class HelloServlet extends HttpServlet{
```

```
public void service(HttpServletRequest req, HttpServletResponse res)
throws ServletException, IOException{
}
```

```
public void doGet(HttpServletRequest req, HttpServletResponse res)
throws ServletException, IOException {
    res.setContentType("text/html");
    PrintWriter out = res.getWriter();
    out.println("<html>");
    out.println("<body>");
    out.println("doGet Hello World!");
    out.println("</body>");
    out.println("</html>");
    out.close();
}
```

```
public void doPost(HttpServletRequest req, HttpServletResponse res)
throws ServletException, IOException {
    res.setContentType("text/html");
    PrintWriter out = res.getWriter();
    out.println("<html>");
    out.println("<body>");
    out.println("doPost Hello World!");
    out.println("</body>");
}
```

} (选择一项)

- a) 一个 HTML 页面，页面上显示 doGet Hello World!
- b) 一个 HTML 页面，页面上显示 doPost Hello World!
- c) 一个空白的 HTML 页面
- d) 错误信息

37) 在 J2EE 中，request 对象的 ( ) 方法可以获取页面请求中一个表单组件对应多个值时的用户的请求数据。(选择一项)

- a) String getParameter(String name)
- b) String[] getParameter(String name)
- c) String getParameterValues(String name)
- d) String[] getParameterValues(String name)

38) ( ) 是发送邮件的协议。(选择一项)

- |         |         |
|---------|---------|
| a) SMTP | b) LAMP |
| c) POP3 | d) HTTP |

39) WEB 应用中，常用的会话跟踪方法不包括 ( )。(选择一项)

- |           |                |
|-----------|----------------|
| a) URL 重写 | b) Cookie      |
| c) 隐藏表单域  | d) 有状态 HTTP 协议 |

40) 在 J2EE 中，`#{2 + "4"}` 将输出 ( )。(选择一项)

- |          |                   |
|----------|-------------------|
| a) 2 + 4 | b) 6              |
| c) 24    | d) 不会输出，因为表达式是错误的 |

41) 在 J2EE 的 Model II 模式中，模型层对象被编写为 ( )。(选择一项)

- |           |             |
|-----------|-------------|
| a) Applet | b) JSP      |
| c) Server | d) JavaBean |



42)J2EE 中，JSP EL 表达式：\${user.loginName}执行效果等同于 ( )。( 选择一项 )

- a) <% = user.getLoginName()%>
- b) <%user.getLoginName();%>
- c) <% = user.loginName%>
- d) <% user.loginName;%>

43)J2EE 中，( ) 类 ( ) 方法用于返回应用程序的上下文路径。( 选择一项 )

- a) HttpServletRequest、getContextPath()
- b) HttpServletRequest 、getPathInfo()
- c) ServletContext、getContextPath()
- d) ServletContext 、getPathInfo()

44)在 J2EE 中，使用 Servlet 过滤器时，需要在 web.xml 通过 ( ) 元素将过滤器映射到 Web 资源。( 选择一项 )

- a) <filter>
- b) <filter-mapping>
- c) <servlet>
- d) <servlet-mapping>

45)给定 JSP 程序源码如下：

```
<html>  
  <% int count =1;%> _____  
</html>
```

以下 ( ) 语句可以在下划线处插入，并且运行后输出结果是：1。( 选择一项 )

- a) <%=++count %>
- b) <% ++count; %>
- c) <% count++; %>
- d) <% =count++ %>

46)在 J2EE 中，在 web.xml 中定义过滤器时可以指定初始化参数，以下定义正确的是 ( )  
( 选择一项 )

- a) <filter>  
 <filter-name>someFilter</filter-name>

<filter-class>filters.SomeFilter</filter-class>

<init-param>

<param-name>encoding</param-name>

<param-value>EUC\_JP</param-value>

</init-param>

</filter>

b) <filter>

<filter-name>someFilter</filter-name>

<init-param>

<param-name>encoding</param-name>

<param-value>EUC\_JP</param-value>

</init-param>

</filter>

c) <filter-mapping>

<filter-name>someFilter</filter-name>

<init-param>

<param-name>encoding</param-name>

<param-value>EUC\_JP</param-value>

</init-param>

</filter-mapping>

d) <filter-mapping>

<filter-name>someFilter</filter-name>

<filter-class>filters.SomeFilter</filter-class>

```
<init-param>
    <param-name>encoding</param-name>
    <param-value>EUC_JP</param-value>
</init-param>
</filter-mapping>
```

47 ) servlet 的生命周期又一系列事件组成，把这些事件按照先后顺序排序，以下正确的是 ( ) ( 选择一项 )

- a) 加载类，实例化，请求处理，初始化，销毁
- b) 加载类，实例化，初始化，请求处理，销毁
- c) 实例化，加载类，初始化，请求处理，销毁
- d) 加载类，初始化，实例化，请求处理，销毁

48) 在 J2EE 中，给定某 Servlet 的代码如下，编译运行该文件，以下陈述正确的是 ( )。  
( 选择一项 )

```
Public class Servlet1 extends HttpServlet{
    Public void init() throws ServletException{
    }
    Public void service(HttpServletRequest request,HttpServletResponse response)
        Throws ServletException,IOException{
        PrintWriter out = response.getWriter();
        Out.println("hello!");
    }
}
```

- a) 编译该文件时会提示缺少 doGet() 或者 doPost() 方法，编译不能够成功通过



- b) 编译后，把 Servlet1.class 放在正确位置，运行该 Servlet，在浏览器中会看到输出文字：hello!
- c) 编译后，把 Servlet1.class 放在正确位置，运行该 Servlet，在浏览器中看不到任何输出的文字
- d) 编译后，把 Servlet1.class 放在正确位置，运行该 Servlet,在浏览器中会看到运行期错误信息

49) 某 web 应用的上下文路径是 root，配置文件 web.xml 中有如下片段：

```
<servlet>

    <servlet-name>process</servlet-name>

    <servlet-class>slt.ProcessServlet</servlet-class>

</servlet>

<servlet-mapping>

    <servlet-name>process</servlet-name>

    <url-pattern>/proc</url-pattern>

</servlet-mapping>
```

以下说法正确的是 ( ) ( 选择二项 )

- a) 在包 slt 中，有一个名为 ProcessServlet.class 的文件
- b) 该 servlet 的访问路径是 http://机器 IP:端口号/root/proc
- c) 该 servlet 的访问路径是 http://机器 IP:端口号/root/ProcessServlet
- d) 该 servlet 的访问路径是 http://机器 IP:端口号/root/ProcessServlet

50) 在 J2EE 中，test.jsp 文件中有如下一行代码：

```
<jsp:useBean id="user" scope="___" type="com.UserBean">
```

要使 user 对象可以作用于整个应用程序，下划线中应添入 ( ) ( 选择一项 )

- a) page
- b) request
- c) session
- d) application

1) 在 J2EE 中，`<%=2+4%>`将输出 ( 6 )。( 选择一项 )

- a) 2+4
- b) 6
- c) 24
- d) 不会输出，因为表达式是错误的

2) 在 J2EE 中，request 对象的 ( ) 方法可以获取页面请求中一个表单组件对应多个值时的用户的请求数据。( 选择一项 )

- a) `String getParameter(String name)`
- b) `String[] getParameter(String name)`
- c) `String getParameterValues(String name)`
- d) `String[] getParameterValues(String name)`

5) 在 J2EE 中，test.jsp 文件中有如下一行代码：

```
<jsp:useBean id="user"scope="_"type="com.UserBean"/>
```

要使 user 对象只能在使用 test.jsp 时使用，当加载页面时就会将其销毁。下划线中应填入 ( )。( 选择一项 )

- a) page
- b) request
- c) session
- d) application

6) 要设计一个大型的基于多层体系结构的 web 应用系统，系统界面端基于 HTML 和 XML，

该系统要不断的进行版本升级 ( 即日后维护、二次需求开发等任务量较大 ), 要达成这些目

标, 最适合的开发模式是 ( )。( 选择一项 )

- a) DAO
- b) ValueObject
- c) MVC
- d) Proxy

10 ) Web 应用中, 常用的会话跟踪方法不包括 ( )。( 选择一项 )

- a) URL 重写
- b) Cookie
- c) 隐藏表单域
- d) 有状态 HTTP 协议

1、 Tomcat 服务器的默认端口是 8080。怎样修改 tomcat 的端口\_\_在 Tomcat 下打开 conf 文件夹找到 server.xml 文件用记事本打开在<Connector port="8080">中把 8080 更改即可\_\_\_\_,

2、 异常的处理, 可在该方法的代码段中包含 3 类代码: try、catch 和 finally 代码块。

3、 jsp 主要内置对象有: application、exception、pageContext、request、session、response、out、config、page。

4、 理论上, GET 是 用于获取服务器信息并将其作为响应返回给客户端, POST 是 用于客户端把数据传送到服务器。



- 5、 application 对象是 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，而 session 对象则是每个客户专用的。
- 6、 JDBC 的主要任务是：与数据库建立连接、发送 SQL 语句、处理结果。
- 7、 一个 Bean 由三部分组成：实现 java.io.Serializable 接口、提供无参数的构造方法、提供 getter ( ) 和 setter ( ) 方法访问它的属性。
- 8、 Java Servlet 的主要特点是：方便、方便、节省投资。运行 servlet 需要在 web.xml 注册。
- 9、 Servlet 的生命周期分三个时期：装载 Servlet、创建一个 Servlet 实例、销毁。
- 10、 JSP 开发网站的两种模式分为 jsp+javabean 和 jsp+javabean+servlet。

## 二、选择题 ( 20 分 )

- 1、下面哪一个不是 JSP 本身已加载的基本类？ ( b )  
A、 java.lang.\* B、 java.io.\* C、 javax.servlet.\* D、 javax.servlet.jsp.\*
- 2、对于预定义<%!预定义%>的说法错误的是： ( c )  
A、一次可声明多个变量和方法，只要以“;”结尾就行 B、一个声明仅在一个页面中有效  
C、声明的变量将作为局部变量 D、在预定义中声明的变量将在 JSP 页面初始化时初始化
- 3、从“员工”表的“姓名”字段中找出名字包含“玛丽”的人，下面哪条 select 语句正确：  
( D )  
A、 Select \* from 员工 where 姓名 = '\_玛丽\_' B、 Select \* from 员工 where 姓名 = '%玛丽\_'  
C、 Select \* from 员工 where 姓名 like '\_玛丽%' D、 Select \* from 员工 where 姓名 like '%玛丽%'
- 4、下述选项中不属于 JDBC 基本功能的是： ( D )

- A. 与数据库建立连接      B. 提交 SQL 语句      C. 处理查询结果  
D. 数据库维护管理

5、在 JSP 中使用<jsp:getProperty>标记时，不会出现的属性是：( c )

- A. name      B. property      C. value      D. 以上皆不会出现

6、Page 指令用于定义 JSP 文件中的全局属性，下列关于该指令用法的描述不正确的是：  
( D )

- A. <%@ page %>作用于整个 JSP 页面。  
B. 可以在一个页面中使用多个<%@ page %>指令。  
C. 为增强程序的可读性，建议将<%@ page %>指令放在 JSP 文件的开头，但不是必须的。  
D. <%@ page %>指令中的属性只能出现一次。

7、在 JSP 中调用 JavaBean 时不会用到的标记是：( A ) A. <javabean>

- B. <jsp:useBean>      C. <jsp:setProperty>  
D. <jsp:getProperty>

8、关于 JavaBean 正确的说法是：( A )

- A、Java 文件与 Bean 所定义类名可以不同，但一定要注意区分字母的大小写  
B、在 JSP 文件中引用 Bean，其实就是用<jsp:useBean>语句  
C、被引用的 Bean 文件的文件名后缀为.java  
D、Bean 文件放在任何目录下都可以被引用

9、Servlet 程序的入口点是：( A )

- A、init ( )      B、main ( )      C、service ( )      D、doGet ( )

10、不能在不同用户之间共享数据的方法是 ( A )

- A、通过 cookie      B、利用文件系统      C、利用数据库      D、通过 ServletContext 对象