1. .以下是web工程的目录结构：  
Web工程名（目录）  
——WEB-INF（目录）  
—–lib（目录）  
—–classes（目录）  
—–web.xml（文件）  
——页面  
struts2的配置文件struts.xml应放置的目录是：（）。  
A. web工程目录  
B. WEB-INF目录  
C. lib目录  
D. classes目录  
正确答案：D  
2.

下面不是jQuery对象的访问方法的是（）。  
A. each(callback)  
B. index()  
C. size()  
D. index(subject)  
正确答案：B  
3.

下面是struts.xml中的配置代码：

<package name=”employee” namespace=”/control/employee” extends=”struts-default”>

<action name=”list\_\*” class=”cn.tarena.action.CommonAction” method=”{1}”>

<result name=”success”>/WEB-INF/page/{1}.jsp</result>

</action>

</package>

以下说法不正确的有：

A.

\*号表示通配符

B.

如果访问CommonAction的路径是为 list\_addUI 那么{1}位置的值应该是addUI

C.

如果访问CommonAction的路径是为 list\_addUI那么可以访问到addUI.jsp页面

D.

访问CommonAction时，只有唯一的访问路径

正确答案：D  
4.

下面是FooAction的代码

public class FooAction {

public String service(){

System.out.println(“service method”);

return “success”;

}

public String action() {

System.out.println(“action method”);

return “success”;

}

public String execute() {

System.out.println(“execute method”);

return “success”;

}

}

下面是struts.xml中的配置:

<package name=”struts2″ extends=”struts-default” namespace=”/first”>

<action name=”foo” class=”com.tarena.struts2.action.FooAction”> </action>

</package>

访问FooAction，调用的方法是：（）。

A.

execute

B.

action

C.

service

D.

init

正确答案：A  
5. 在HTTP协议中，用于发送大量数据的方法是（）。  
A. get  
B. post  
C. put  
D. options  
正确答案：B  
6.

在web.xml设置了<url-pattern>\*.do</url-pattern>，以下路径不是正确访问路径的是：（）。

A.

http://localhost:8080/web /list

B.

http://localhost:8080/web/list.do

C.

http://localhost:8080/web/jsp/list.do

D.

http://localhost:8080/web/jsp/emp/list.do

正确答案：A  
7.

在JavaEE中，若要在JSP中正确使用标签：<x:getKing/>，在JSP中声明的taglib指令为：<%@taglib uri = “/WEB-INF/myTags.tld” prefix = “\_\_\_\_\_”>，下划线处应该是（）。  
A.

x  
B.

getKing  
C.

myTags  
D.

king  
正确答案：A  
8.

以下是struts.xml文件的代码：

<package name=”struts” extends=”struts-default” namespace=”/first”>

<action name=”login” class=”com.tarena.struts2.action.LoginAction”>

<result name=”success”>/success.jsp</result>

<result name=”fail”>/fail.jsp</result>

</action>

</package>  
访问LoginAciotn的正确路径是（）  
A.

<a href=“struts/login.action”>访问LoginAction</a>  
B.

<a href=“first/login.action”>访问LoginAction</a>  
C.

<a href=“login.action”>访问LoginAction</a>  
D.

<a href=”first/login.do”>访问LoginAction</a>  
正确答案：B  
9.

以下是struts.xml文件的代码：

<package name=”first” extends=”struts-default” namespace=”/first” >

<action name=”one” class=”com.tarena.struts2.action.OneAction”>

<result name=”success”>/success.jsp</result>

</action>

</package>

<package name=”first” extends=”struts-default” namespace=”/second”>

<action name=”two” class=”com.tarena.struts2.action.TwoAction”>

<result name=”success”>/success.jsp</result>

</action>

</package>  
关于上述代码，以下说法正确的是（）。  
A.

可以通过<a href=“first/one.action”>clickme</a>访问OneAction  
B.

可以通过<a href=“second/two.action”>clickme</a>访问TwoAction  
C.

编译出错  
D.

启动服务器加载struts.xml文件时出错  
正确答案：D  
10.

假设web应用的文件根目录为servlet01，那么可以找到struts-core.jar文件的目录是：（）。

A.

servlet01目录下

B.

servlet01\WEB-INF目录下

C.

servlet01\WEB-INF\classes目录下

D.

servlet01\WEB-INF\lib目录下

正确答案：D  
11.

给定一个 Servlet 的doPost方法中的代码片段，如下：

request.setAttribute(“name”,”zhang”);

request.getRequestDispatcher(“/myservlet”).forward(request, response);//转发到MyServlet

以下说法正确的是：（）。

A.

在 MyServlet 中，可以使用 String str=request.getAttribute(“name”); 把属性 name的值取出来

B.

在 MyServlet 中，可以使用String str=(String)request.getAttribute(“name”); 把属性 name的值取出来

C.

在 MyServlet 中，可以使用String str=request.getParameter (“name”); 把属性 name的值取出来

D.

在 MyServlet 中，无法取出属性name的值

正确答案：B  
12.

下面是struts.xml中的配置代码：

<package name=”tarena” namespace=”/” extends=”struts-default”>

<action name=”list” class=”com.tarena.action.ListAction”>

<result name=”success”>

/employee\_list.jsp

</result>

</action>

<action name=”redirectAction”>

//插入行

</action>

</package>

当访问redirectAction.action时可以跳转到employee\_list.jsp则插入行的代码是（）  
A.

<result type=”redirectAction”>/list</result>  
B.

<result type=”chain”>/list.action</result>  
C.

<result type=”stream”>/list.action</result>  
D.

<result type=”redirect”>/list</result>  
正确答案：A  
13.

关于MVC模式，下列选项中的说法错误的是 ：（）。

A.

MVC是一种软件架构思想，将软件的组成划分成三部分

B.

采用MVC模式使模型和视图可以独立修改

C.

在MVC中，M指的是Model；V指的是View；C指的是Controller

D.

采用MVC模式，模型部分并不能重用，而视图部分则可以完全的重用

正确答案：D  
14.

public class Book {

private int bookid;

private String name;

private int price;

//省略getter和setter

public Book(int bookid, String name, int price) {

this.bookid = bookid;

this.name = name;

this.price = price;

}

}

public class PersonListAction {

private List<Book> books;

//省略getter和setter

public String execute(){

books = new ArrayList<Book>();

books.add(new Book(56, “javaweb”, 90));

books.add(new Book(80, “ejb”, 78));

books.add(new Book(23, “spring”, 50));

return “list”;

}

}

访问PersonListAction进入页面后，使用<s:iterator>标签取出books的内容，下列选项正确的是：（）。

A.

<s:iterator value=”books”>

<s:property value=”#name”/>,<s:property value=”#price”/><br/>

</s:iterator>

B.

<s:iterator value=”#books”>

<s:property value=”#name”/>,<s:property value=”#price”/><br/>

</s:iterator>

C.

<s:iterator value=”#books”>

<s:property value=”name”/>,<s:property value=”price”/><br/>

</s:iterator>

D.

<s:iterator value=”books”>

<s:property value=”name”/>,<s:property value=”price”/><br/>

</s:iterator>

正确答案：D  
15.

类别（Category）类的定义如下：

class Category {

private int id;

private String name;

private List<Category> subCat;

// getter and setter…

}

request对象中存有名为cat的Category对象，可以将该对象的subCat属性迭代输出的代码是：( )。

A.

<c:forEach items=”${cat.subCat}” var=”cat”>…</c:forEach>

B.

<c:forEach items=”${cat}” var=”cat”>…</c:forEach>

C.

<c:forEach items=”${cat.subCat}” var=”${cat}”>…</c:forEach>

D.

<c:forEach items=”${cat}” var=”${cat}”>…</c:forEach>

正确答案：A  
16.

在Java Web应用程序中，给定某Servlet程序的代码如下所示：

public class ServletFoo extends HttpServlet{

public void doGet(HttpServletRequest request,HttpServletResponse response)

throws ServletException IOException{

System.out.println(“get”);

}

public void doPost(HttpServletRequestrequest.HttpServletResponse response)

throws ServletException,IOException{

System.out.println(“post”);

doGet(reauest,response);

}

}

用户在浏览器地址栏中键入正确的请求URL并回车后，在控制台上显示的结果是（ ）。  
A.

get  
B.

post  
C.

get

post  
D.

post

get  
正确答案：A  
17. 在JSP中，page指令的属性中，用来引入需要的包或类的是( )。  
A. extends  
B. import  
C. languge  
D. contentType  
正确答案：B  
18.

给定某Servlet程序的片段如下,用户在浏览器地址栏中键盘键入正确的请求URL并回车后,在控制台上显示的结果是 ( )。  
public void doGet(HttpServletRequest request,HttpServletResponse response)  
throws ServletException ,IOException {  
System.out.println(“get”);  
}  
public void doPost(HttpServletRequest request,HttpServletResponse response)  
throws ServletException ,IOException {  
System.out.println(“post”);  
}  
A. get  
B. post  
C. get  
post  
D. post  
get  
正确答案：A  
19. 在a.jsp中有代码片断如下：  
<form action =”b.jsp” method =”post” name=”myForm”>  
loginName:<input type=”text” name=”loginName”/>  
< input type=”submit” name=”submit”/>  
< /form>  
在b.jsp中想输出在a.jsp页面上输入的loginName的值，以下选项正确的是：（）。  
A. <%= request.getParameter(“loginName”)%>  
B. <%=(String )request.gerAttribute(“loginName”)%>  
C. <%= request.getParameter(“loginname”) %>  
D. <%=(String )request.getAttribute(“loginname”) %>  
正确答案：A  
20.

某JSP中有如下代码，显示结果为（）

<%

int a = 5;

request.setAttribute(“a”,”123”);

session.setAttribute(“a”,”456”);

%>

<c:out value=”${a}”/>  
A.

5  
B.

123  
C.

456  
D.

Null  
正确答案：B  
21.

下列不属于OGNL工具的组成部分的是：（）。

A.

OGNL引擎

B.

Root存储区

C.

Context存储区

D.

Result

正确答案：D  
22. Servlet 可以存储数据的三个不同的作用域是（）。  
A. 请求、会话和上下文  
B. 响应、会话和上下文  
C. 请求、响应和会话  
D. 请求、响应和上下文  
正确答案：A  
23.

下面是MethodDemoAction的代码：

public class MethodDemoAction {

public String add(){

System.out.println(“add method invoke………”);

return “success”;

}

}

下面是struts.xml中的配置代码：

<package name=”struts2″ extends=”struts-default” namespace=”/first”>

<action name=”demo” class=”com.tarena.struts2.action.MethodDemoAction”/>

</package>  
下列能调用到MethodDemoAction的add方法的请求连接是：（）。  
A.

<a href=”first/demo!add.action”>add method</a>  
B.

<a href=”first/demo&add.action”> add method</a>  
C.

<a href=”first/demo?add.action”> add method</a>  
D.

<a href=”first/demo.action”> add method</a>  
正确答案：A  
24.

MVC设计模式包含模型层、视图层和控制层，在下列组件中扮演控制器角色的是（ ）。  
A.

Servlet  
B.

JSP  
C.

JavaBean  
D.

HTML  
正确答案：A  
25.

下列不属于Struts2的<s:form>标签的属性的是：（）。

A.

action

B.

theme

C.

namespace

D.

simple

正确答案：D  
26.

有关Servlet的生命周期说法不正确的是：（）。

A.

Servlet的生命周期由Servlet实例控制

B.

init()方法在创建完Servlet实例后对其进行初始化，传递的参数为实现ServletConfig类的对象

C.

service()方法响应客户端发出的请求

D.

在容器移除servlet 时，调用destroy方法，进行资源的释放

正确答案：A  
27. 下列属于抽象类的是（）。  
A. ServletConfig  
B. HttpServlet  
C. Cookie  
D. HttpServletRequest  
正确答案：B  
28. 在JavaEE中，要将某个过滤器映射到Web应用程序的每个Web资源，以下在web.xml，中的配置正确的是（ ）。  
A.

<filter-mapping>  
< filter-name>someFilter</filter-name>  
< url-pattern>/\*<url-pattern>

</filter-mapping>  
B. <filter>  
< filter-name>someFilter</filter-name>  
< url-pattern>/\*<url-pattern>  
< /filter>  
C. <filter-mapping>  
< filter-name>someFilter</filter-name>  
< url-pattern>\*<url-pattern>  
< /filter-mapping>  
D. <filter>  
< filter-name>someFilter</filter-name>  
< url-pattern>\*<url-pattern>  
< /filter>  
正确答案：A  
29.

下面是struts.xml配置代码：

<package name=”employee” namespace=”/control/employee” extends=”struts-default”>

<action name=”list” class=”com.tarena.action.EmployeeAction”>

<result name=”success” type=”redirect”>/employee\_add.jsp</result>

</action>

</package>

<package name=”dept” namespace=”/control/department”extends=”struts-default”>

<action name=”dept”>

//插入行

</action>

</package>

当访问dept.action时,进入EmployeeAction，则插入行处的代码是：（）。

A.

<result type=”redirect”>list</result>

B.

<result type=”redirect”>/list.action</result>

C.

<result type=”redirectAction”>/control/employee/list.action</result>

D.

<result type=”redirectAction”>

<param name=”actionName”>list</param>

<param name=”namespace”>/control/employee</param>

</result>

正确答案：D  
30.

下面是FooAction的代码:

public class FooAction {

public String service(){

System.out.println(“service method”);

return “success”;

}

public String action() {

System.out.println(“action method”);

return “success”;

}

public String execute() {

System.out.println(“execute method”);

return “success”;

}

}

下面是struts.xml中的配置:

<package name=”struts2″ extends=”struts-default” namespace=”/first” method=”service”>

<action name=”foo” class=”com.tarena.struts2.action.FooAction”></action>

</package>  
访问FooAction，调用的方法是：（）。

A.

execute  
B.

action  
C.

service  
D.

init  
正确答案：C  
31.

DefaultAction类的代码如下：

public class DefaultAction extends ActionSupport

{

public String execute(){

System.out.println(“execute方法执行”);

return “success”;

}

}

DefaultInterceptor类的代码如下：

public class DefaultInterceptor extends AbstractInterceptor{

private String name;

public void setName(String name){

this.name = name;

}

public String intercept(ActionInvocation ai) throws Exception {

System.out.println(this.name + “执行”);

String result = ai.invoke();

System.out.println(this.name + “执行完毕”);

return result;

}

}

struts.xml文件代码：

<package name =”testInterceptor” extends=”struts-default”>

<interceptors>

<interceptor name =”order” class =”com.tarena.interceptor.DefaultInterceptor”/>

</interceptors>

<action name=”order” class=”com.tarena.action.DefaultAction”>

<result name=”success”>/success.jsp</result>

<interceptor-ref name=”order” >

<param name=”name”>拦截器1</param>

</interceptor-ref>

<interceptor-ref name=”order” >

<param name=”name”>拦截器2</param>

</interceptor-ref>

<interceptor-ref name=”order” >

<param name=”name”>拦截器3</param>

</interceptor-ref>

<interceptor-ref name=”defaultStack”/>

</action>

</package>  
直接访问DefaultAction类，控制台输出是：（）

A.

拦截器1执行

拦截器2执行

拦截器3执行

execute方法执行

拦截器1执行完毕

拦截器2执行完毕

拦截器3执行完毕  
B.

拦截器1执行

拦截器2执行

拦截器3执行

execute方法执行

拦截器3执行完毕

拦截器2执行完毕

拦截器执行完毕  
C.

execute方法执行

拦截器1执行

拦截器2执行

拦截器3执行

拦截器1执行完毕

拦截器2执行完毕

拦截器3执行完毕  
D.

拦截器1执行

拦截器2执行

拦截器3执行

拦截器3执行完毕

拦截器2执行完毕

拦截器执行完毕

execute方法执行  
正确答案：B  
32.

请看下列代码：

public class Dept {

private Integer id;

private String name;

private List<Emp> empList;

//省略getter和setter

}

public class Emp {

private Integer id;

private String name;

private Dept dept;

//getter和setter

}  
使用OGNL访问Emp类的对象（emp）的dept属性的name信息，下列选项正确的是：（）。  
A.

Object name = Ognl.getValue(“dept.name”, emp);  
B.

Object name = Ognl.getValue(“emp.dept.name”, emp);  
C.

Object name = Ognl.getValue(“emp.dept.name”, Emp);  
D.

Object name = Ognl.getValue(“dept.name”, Emp);  
正确答案：A  
33.

使用Struts2做文件上传时，UploadAction类的代码如下：

public class UploadAction {

private File mf ;

private String mfFileName;

private String mfContentType;

//省略getter和setter

public String execute(){

//实现文件上传

}

}  
请推断出文件上传的页面的代码：（）。  
A.

<s:form action=”upload” method=”post enctype=”multipart/form-data”>

<s:file name=”some” label=”File”></s:file>

<s:submit value=”上传”></s:submit>

</s:form>  
B.

<s:form action=”upload” method=”get ” enctype=”multipart/form-data”>

<s:file name=”some” label=”File”></s:file>

<s:submit value=”上传”></s:submit>

</s:form>  
C.

<s:form action=”upload” method=”get” enctype=”multipart/form-data”>

<s:file name=”mf” label=”File”></s:file>

<s:submit value=”上传”></s:submit>

</s:form>  
D.

<s:form action=”upload” method=”post enctype=”multipart/form-data”>

<s:file name=”mf” label=”File”></s:file>

<s:submit value=”上传”></s:submit>

</s:form>  
正确答案：D  
34.

下面为部署文件中的Servlet映射代码片段，正确的是：( )

A.

<servlet-mapping>

<servlet>Demo</servlet>

<url-pattern>/demo</url-pattern>

</servlet-mapping>

B.

<servlet-name>

<servlet-mapping>demo</servlet-mapping>

<url-pattern>/Demo</url-pattern>

</servlet-name>

C.

<servlet-mapping>

<servlet-name>Demo</servlet-name>

<url-pattern>/demo/\*</url-pattern>

</servlet-mapping>

D.

<servlet >

<servlet-name>Demo</servlet-name>

<url-pattern>/demo/\*</url-pattern>

</servlet >

正确答案：C  
35.

在jQuery中$(“#test”)和$(“.test”)表示的含义是:( )。

A.

$(“#test”)表示选择id为test的元素

$(“.test”) 表示选择id为test的元素

B.

$(“#test”)表示选择id为test的元素

$(“.test”) 表示选择所用CSS类为test的元素

C.

$(“#test”)表示选择所用CSS类为test的元素

$(“.test”) 表示选择id为test的元素

D.

$(“#test”)表示选择所用CSS类为test的元素

$(“.test”) 表示选择所用CSS类为test的元素

正确答案：B  
36.

以下代码实现的功能是使用struts2判断用户是否能成功登录，如果登录成功，把用户的信息放入session对象中保存，显示在main.jsp界面上。

User类代码如下：

package com.tarena.outman.day03.entity;

public class User {

private int userId;

private String username;

private String password;

//省略getter和setter

}

BaseAction.java类代码如下：

public class BaseAction implements SessionAware{

//在struts使用session

空白处1

}

LoginAction.java代码如下：

public class LoginAction extends BaseAction{

private User user;

//省略getter和setter

private UserDao userDao = new UserDao();

public String login() {

user = userDao.validate(user.getUsername(), user.getPassword());

if (user != null) {

// 登录成功 把user对象放到session对象中保存

空白处2

return “success”;

} else {

// 登录失败

return “fail”;

}

}

}

loginform.jsp代码如下：

<%@page pageEncoding=”utf-8″%>

<html>

<head>

<title></title>

</head>

<body>

<form action=”login.action” method=”post”>

<table>

空白处3

<tr colspan=”2″>

<td><input value=”提交” type=”submit” /></td>

</tr>

</table>

</form>

</body>

</html>

main.jsp代码如下：

<html>

<head>

<title></title>

</head>

<body>

<h1>你好， 空白处4 </h1>

</body>

</html>

struts.xml代码如下：

<struts>

<package name=”core” extends=”struts-default” namespace=”/day03″>

空白处5

<result name=”success”>/WEB-INF/jsp/day03/main.jsp</result>

<result name=”fail”>/WEB-INF/jsp/day03/loginform.jsp</result>

</action>

</package>  
< /struts>  
(1).

下列选项中，可以填入横线空白处 1 的代码是（）。  
A.

protected HttpSesion session;

public void setSession(HttpSesion session) {

this.session = session;

}  
B.

private HttpSesion session;

public void setSession(HttpSesion session) {

this.session = session;

}  
C.

protected Map<String, Object> session;

public void setSession(Map<String, Object> session) {

this.session = session;

}  
D.

private Map<String, Object> session;

public void setSession(Map<String, Object> session) {

this.session = session;

}  
正确答案：C  
(2).

下列选项中，可以填入横线空白处 2 的代码是（）。  
A.

session.put(“day03.user”, user);  
B.

session.get(“day03.user”, user);  
C.

session.setAttibute(“day03.user”, user);  
D.

session.getAttibute(“day03.user”, user);  
正确答案：A  
(3).

下列选项中，可以填入横线空白处 3 的代码是（）。  
A.

<tr>

<td>用户名：</td>

<td><input name=”username” type=”text” /></td>

</tr>

<tr>

<td>密码</td>

<td><input name=”password” type=”password” /></td>

</tr>  
B.

<tr>

<td>用户名：</td>

<td><input name=”User.username” type=”text” /></td>

</tr>

<tr>

<td>密码</td>

<td><input name=”User.password” type=”password” /></td>

</tr  
C.

<tr>

<td>用户名：</td>

<td><input name=”user.username” type=”text” /></td>

</tr>

<tr>

<td>密码</td>

<td><input name=”user.password” type=”password” /></td>

</tr  
D.

<tr>

<td>用户名：</td>

<td><input name=”Username” type=”text” /></td>

</tr>

<tr>

<td>密码</td>

<td><input name=”Password” type=”password” /></td>

</tr  
正确答案：C  
(4).

下列选项中，可以填入横线空白处 4 的代码是（）。  
A.

<s:property value=”session['user'].username” />  
B.

<s:property value=”#session['user'].username” />  
C.

<s:property value=”session['day03.user'].username” />  
D.

<s:property value=”#session['day03.user'].username” />  
正确答案：D  
(5).

下列选项中，可以填入横线空白处 5 的代码是（）。  
A.

<action name=”login”

class=”com.tarena.outman.day03.LoginAction”>  
B.

<action name=”login”

class=”com.tarena.outman.day03.LoginAction” method=”execute”>  
C.

<action name=”login”

class=”com.tarena.outman.day03.LoginAction” method=”form “>  
D.

<action name=”login”

class=”com.tarena.outman.day03.LoginAction” method=”login”>  
正确答案：D  
37.

现有 move.jsp 页面，该页面中的jQuery代码可以完成列表框选项的移动。

此页面的代码如下：

<html>

<head>

<meta http-equiv=”Content-Type” content=”text/html; charset=UTF-8″>

<title>移动多选框</title>

空白处1

<script type=”text/javascript”>

$(function(){

//双击id是notinteachers的列表框中的某个选项，

//该选项移动到id是inteachers列表框中

空白处2 {

var $options = $(‘#notinteachers option’);

$options.each(function(index){

if(this.selected){

$(‘#inteachers’).append($(this));

}

});

});

//选中id是notinteachers的列表框中的某个选项，

//点击按钮b1该选项移动到id是inteachers列表框中

$(‘#b1′).click(function(){

var $options = $(‘#notinteachers option’);

var flag = false;

$options.each(function(index){

if( 空白处3 ){

flag = true;

$(‘#inteachers’).append($(this));

}

});

if(!flag){

alert(‘至少选择一个选项’);

}

});

//点击按钮b2, id是notinteachers的列表框中的所有移动到

//id是inteachers列表框中

$(‘#b2′).click(function(){

var $options = $(‘#notinteachers option’);

$options.each(function(index){

空白处4

});

});

//点击按钮confirm, id是inteachers列表框中的每个项全部处于选中状态

$(‘form’).submit(function(){

var $arr = $(‘#inteachers option’);

$arr.each(function(index){

空白处5

});

return true;

});

});

</script>

</head>

<body>

<div id=”d1″>

<div id=”d1\_head”>

移动多选框选项

</div>

<div id=”d1\_body”>

<form action=”" >

<table>

<tr>

<td>

<select id=”notinteachers” multiple style=”width:150px;height:180px;”>

<option value=”1″>

张三

</option>

<option value=”2″>

李四

</option>

<option value=”3″>

王五

</option>

</select>

</td>

<td>

<input id=”b1″ type=”button” value=”–&gt;” style=”width:120px”>

<br>

<br>

<input id=”b2″ type=”button” value=”–&gt;&gt;” style=”width:120px”>

</td>

<td>

<select id=”inteachers” multiple name=”inteachers” style=”width:150px;height:180px;”>

</select>

</td>

</tr>

</table>

<input type=”submit” value=”confirm” />

</form>

</div>

</div>

</body>  
< /html>  
(1).

下列选项中，可以填入横线空白处 1 的代码是（）。  
A.

<script src=”text/javascript” href=”js/jquery-1.7.1.js”></script>  
B.

<script type=”text/javascript” href=”js/jquery-1.7.1.js”></script>  
C.

<script type=”text/javascript” src=”js/jquery-1.7.1.js”></script>  
D.

<script src=”text/javascript” src=”js/jquery-1.7.1.js”></script>  
正确答案：C  
(2).

下列选项中，可以填入横线空白处 2 的代码是（）。  
A.

$(‘#notinteachers’).dblclick(function()  
B.

$(‘#notinteachers’).click(function()  
C.

$(‘#notinteachers’).doubleClick(function()  
D.

$(‘#notinteachers’).dclick(function()  
正确答案：A  
(3).

下列选项中，可以填入横线空白处 3 的代码是（）。  
A.

this.checked  
B.

this.option.checked  
C.

this.selected  
D.

this.option.selected  
正确答案：C  
(4).

下列选项中，可以填入横线空白处 4 的代码是（）。  
A.

$(‘# notinteachers’).append($(this));  
B.

if(this.selected){

$(‘# notinteachers’).append($(this));

}  
C.

if(this.selected){

$(‘#inteachers’).append($(this));

}  
D.

$(‘#inteachers’).append($(this));  
正确答案：D  
(5).

下列选项中，可以填入横线空白处 5 的代码是（）。  
A.

$(this).attr(‘selected’,true);  
B.

$(this).attribute(‘selected’,true);  
C.

$(this).attr(‘checked’,false);  
D.

$(this).attribute(”,false);  
正确答案：A  
38.

以下代码的功能是：用户输入用户名和密码，选上保存用户名和密码的复选框，点击登录，程序使用cookie保存用户的用户名和密码，当再次登录时，用户名和密码是回填状态，保存用户名和密码的复选框是选中状态；如果用户输入用户名和密码，不选择保存用户名和密码的复选框，点击登录，那么程序将不保存任何信息，再次登录时也无回填信息。

login.jsp代码如下：

<%@ page language=”java” pageEncoding=”utf-8″ %>

<html>

<head>

<title>登录</title>

</head>

<body>

<%

String cookieName = “userName”;

String cookiePwd = “pwd”;

// 获得所有cookie

空白处1

String userName = “”;

String pwd = “”;

String isChecked = “”;

if (cookies != null) {

for (int i = 0; i < cookies.length; i++) {

// 取出登录名和密码

空白处2

}

}

if((!”".equals(userName))&&(!”".equals(pwd))){

isChecked = “checked”;

}

%>

<center>

<form action=”/Servlet/testcookie” method=”post”>

姓名：

<input type=”text” name=”username” value=”<%=userName %>”>

<br>

密码：

<input type=”password” name=”pwd” value=”<%=pwd %>”>

<br>

保存用户名和密码

空白处3

<br>

<input type=”submit” value=”登录”>

</form>

</center>

</body>

</html>

CookieTest.java代码如下：

public class CookieTest extends HttpServlet {

public void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

Cookie userCookie = new Cookie(“userName”, request.getParameter(“username”));

Cookie pwdCookie = new Cookie(“pwd”, request.getParameter(“pwd”));

if (request.getParameter(“savecookie”) != null

&& request.getParameter(“savecookie”).equals(“yes”)) {

userCookie.setMaxAge(7 \* 24 \* 60 \* 60);//7天

pwdCookie.setMaxAge(7 \* 24 \* 60 \* 60);

} else {

空白处4

}

//保存cookie

空白处5

PrintWriter out = response.getWriter();

out.println(“Welcome,” + request.getParameter(“username”));

}

public void doPost(HttpServletRequest request, HttpServletResponse

response) throws ServletException, IOException {

doGet(request, response);

}  
}  
(1).

下列选项中，可以填入横线空白处 1 的代码是（）。  
A.

Cookie[] cookies = response.getCookies();  
B.

Cookie[] cookies = response.cookies();  
C.

Cookie[] cookies = request.getCookies();  
D.

Cookie[] cookies = request.cookies();  
正确答案：C  
(2).

下列选项中，可以填入横线空白处 2 的代码是（）。  
A.

if (cookies[i].getName().equals(cookieName)) {

userName = cookies[i].getValue();

}

if (cookies[i].getName().equals(cookiePwd)) {

pwd = cookies[i].getValue();

}  
B.

if (cookies[i].getCookieName().equals(cookieName)) {

userName = cookies[i].getValue();

}

if (cookies[i].geCookietName().equals(cookiePwd)) {

pwd = cookies[i].getValue();

}  
C.

if (cookies[i].getCookieName().equals(cookieName)) {

userName = cookies[i].getCookieValue();

}

if (cookies[i].getCookieName().equals(cookiePwd)) {

pwd = cookies[i].getCookieValue();

}  
D.

if (cookies[i].getName().equals(cookieName)) {

userName = cookies[i].getCookieValue();

}

if (cookies[i].getName().equals(cookiePwd)) {

pwd = cookies[i].getCookieValue();

}  
正确答案：A  
(3).

下列选项中，可以填入横线空白处 3 的代码是（）。  
A.

<input type=”checkbox” name=”savecookie”

value=”yes” checked=<%=isChecked %>>  
B.

<input type=”radio” name=”savecookie”

value=”yes” <%=isChecked %>>  
C.

<input type=”checkbox” name=”savecookie”

value=”yes” <%=isChecked %>>  
D.

<input type=”radio” name=”savecookie”

value=”yes” checked=<%=isChecked %>>>  
正确答案：C  
(4).

下列选项中，可以填入横线空白处 4 的代码是（）。  
A.

userCookie.setMaxAge(-1);

pwdCookie.setMaxAge(-1);  
B.

userCookie.setMaxAge(1);

pwdCookie.setMaxAge(1);  
C.

userCookie.setMaxAge();

pwdCookie.setMaxAge();  
D.

userCookie.setMaxAge(0);

pwdCookie.setMaxAge(0);  
正确答案：D  
(5).

下列选项中，可以填入横线空白处 5 的代码是（）。  
A.

request.setCookie(userCookie);

request.setCookie(pwdCookie);  
B.

request.addCookie(userCookie);

request.addCookie(pwdCookie);  
C.

response.setCookie(userCookie);

response.setCookie(pwdCookie);  
D.

response.addCookie(userCookie);

response.addCookie(pwdCookie);  
正确答案：D