**1. Which one of the following methods will be invoked when a ServletContext is destroyed?**

1. **contextDestroyed() of javax.servlet.ServletContextListener**
2. contextDestroyed() of javax.servlet.HttpServletContextListener
3. destroy() of javax.servlet.GenericServlet
4. contextDestroyed() of javax.servlet.http.HttpServletContextListener
5. contextDestroyed() of javax.servlet.http.HttpSessionListener

**2.Consider the following class:  
  
import javax.servlet.\*;  
public class MyListener implements ServletContextAttributeListener  
{  
public void attributeAdded(ServletContextAttributeEvent scab)  
{  
System.out.println("attribute added");  
}  
public void attributeRemoved(ServletContextAttributeEvent scab)  
{  
System.out.println("attribute removed");  
}  
}  
Which one of the following statements about the above class is correct?**

1. This class will compile as it is.
2. **This class will compile only if the attributeReplaced() method is added to it.**
3. This class will compile only if the attributeUpdated() method is added to it.
4. This class will compile only if the attributeChanged() method is added to it.
5. This class will compile only if the attributeDestroyed() method is added to it

**3. You want to write Servlet classes for a new web protocol of yours that you have developed.  
Which one of the classes from the following would you extend your servlets from?**

1. HttpServlet
2. **GenericServlet**
3. Servlet
4. AbstractServlet

**4. Given the following web application deployment descriptor:  
<web-app>  
 <servlet>  
 <servlet-name>myServlet</servlet-name>  
 <servlet-class>MyServlet</servlet-class>  
 …  
 </servlet>  
 <servlet-mapping>  
 <servlet-name>myServlet</servlet-name>  
 <url-pattern>\*.jsp</url-pattern>  
 </servlet-mapping>**

1. servlet-mapping element should be inside servlet element
2. url-pattern can't be defined that way
3. **if you make the http call: http://host/Hello.jsp the servlet container will execute MyServlet**
4. It would work with any extension excepting jsp,html,htm
5. It should be <urlpattern> tag instead of <url-pattern>

5**.Which two of the following statements are true?**

1. **The deployment descriptor must be well formed, i.e. every tag every tag must have a matching closing tag (or be closed with a forward slash)**
2. **The deployment descriptor contains information on the configuration of servlets in a web** **application**
3. A single deployment descriptor may be used to configure multiple web applications
4. The deployment descriptor cannot contain comments
5. The deployment descriptor can be edited by the designer

6. **Which one of the following statements is true?**

1. A file marked as load-on-startup will be returned as the default from a request to a directory with no file component.
2. By default if a directory contains a file called welcome.jsp it will be returned from a request where no file name is specified.
3. Only a jsp file can be marked as a welcome file, not a servlet
4. **A file or a jsp can be marked as a welcome file within the deployment descriptor**
5. A jsp file will be placed in the source packages folder

7. **Which three of the following methods are declared in HttpServletRequest as opposed to in ServletRequest?**

1. **getMethod()**
2. **getHeader()**
3. **getCookies()**
4. getInputStream()
5. getParameterName()

8. **.Which one of the following is the abstract method in the HttpServlet abstract class?**

1. service()
2. doGet()
3. doPost()
4. **No abstract methods**

9. **. Which one of the following is a valid tag for configuring a listener class?**

1. <contextlistener>   
    <listener-class>mypackage.SomeListener</listener-class>   
   </contextlistener>
2. **<listener>   
    <listener-class>mypackage.SomeListener</listener-class>   
   </listener>**
3. <context-listener>   
    <listener-class>mypackage.SomeListener</listener-class>   
   </context-listener>
4. <listener>  
   <class>mypackage.SomeListener</class>  
   </listener>

10. **Which two of the following statements are true?**

1. The sendRedirect method can only process an absolute URL as a parameter
2. After a sendRedirect method call, the browser will automatically return to the original URL
3. **If the sendRedirect method is called after the response has been committed, an exception will be thrown**
4. **SendRedirect is a method of the HttpServletResponse Interface**
5. SendRedirect is a method of the HttpServletRequest class

11. **Consider the following servlet code:   
  
import java.io.\*;   
import javax.servlet.\*;   
import javax.servlet.http.\*;   
public class MyFirstServlet extends HttpServlet   
{   
 public void service(HttpServletRequest req, HttpServletResponse res) throws ServletException, IOException   
 {   
 res.setContentType("text/html");   
 }   
  
 public void doGet(HttpServletRequest req, HttpServletResponse res) throws ServletException, IOException   
 {   
 res.getWriter().println("This is doGet!");   
 }   
   
 public void doPost(HttpServletRequest req, HttpServletResponse res) throws ServletException, IOException   
 {   
 res.getWriter().println("This is doPost!");   
 }   
}   
  
This servlet is executed by clicking on a submit button of a form in a web page:   
<form action="/MyFirstServlet" method=POST>   
<input type=submit>   
</form>   
  
Predict the output.  
Select one answer from the following.**

1. This is doPost
2. This is doGet
3. "This is doPost" as well as "This is doGet"
4. **It will not print anything**
5. “This is doGet!” is printed

12. **Which two of the following methods must be implemented to create a class that implements an HttpSessionListener?**

1. **public void sessionCreated(HttpSessionEvent se)**
2. public void sessionAdded(HttpSessionEvent se)
3. **public void sessionDestroyed(HttpSessionEvent se)**
4. public void sessionModified(HttpSessionEvent se)
5. Public void sessionUpdated(HttpSessionEvent se)

**Following is the code for doGet() and doPost() method of TestServlet:**

**public void doGet(HttpServletRequest req, HttpServletResponse res)**

**{**

**try**

**{**

**PrintWriter pw = res.getWriter();**

**pw.println("<html><head></head><body>");**

**String name = req.getParameter("name");**

**String[] hobbies = req.getParameterValues("hobbies");**

**pw.println("Hello, "+name+"!<br> As I understand, your hobbies are :");**

**for(int i=0; i< hobbies.length; i++) pw.println("<br>"+hobbies[i]);**

**pw.println("</body></html>");**

**}**

**catch(Exception e)**

**{**

**e.printStackTrace();**

**}**

**}**

**public void doPost(HttpServletRequest req, HttpServletResponse res)**

**{**

**doGet(req, res);**

**}**

**Which one of the following statements is correct?**

**a) This will only work for HTTP GET requests**

**b) This will only work for HTTP POST requests**

**c) This will work for HTTP GET as well as POST requests.**

**d) It will throw an exception at runtime, as you cannot call doGet() from doPost()**

**e) It'll throw an exception at runtime only if a POST request is sent to it**

**14.** **.Consider the following code snippet:  
  
package com.evaluation;  
import java.io.\*;  
import java.net.\*;  
import javax.servlet.\*;  
import javax.servlet.http.\*;  
  
public class ReqD extends HttpServlet {  
protected void doGet(HttpServletRequest request, HttpServletResponse response)  
throws ServletException, IOException {  
 ServletContext sc = this.getServletContext();  
 RequestDispatcher dis = sc.getRequestDispatcher("/menu.jsp");  
 if (dis != null){  
 dis.include(request, response);  
 }  
 PrintWriter out = response.getWriter();  
 out.print("Output after menu.jsp");  
 }   
}  
Which one of the following will be the correct outcome when the above is executed?**

1. Compilation and output of the contents of menu.jsp only
2. **Compilation and output of the contents of menu.jsp followed by "Output after menu.jsp"**
3. Compilation error
4. Compilation, but runtime error, the buffer has been flushed

15. **.Which two of the following statements are true?**

1. **Every (non distributed) web application has only one instance of ServletContext**
2. Every instance of HttpServlet has only one instance of ServletContext
3. **The ServletContext object is retrieved via the getServletContext method of ServletConfig**
4. The ServletConfig object is retrieved via the getServletConfig method of ServletContext

16. **Which one of the following is the correct way of writing the init-param for a servlet in web.xml deployment descriptor file?**

1. **<servlet>**

**<servlet-name>HelloGeneric</servlet-name>  
 <servlet-class>HelloGeneric</servlet-class>  
 <init-param>  
 <param-name>count</param-name>  
 <param-value>13</param-value>  
 </init-param>  
   
 </servlet>  
  
 <servlet-mapping>  
 <servlet-name>HelloGeneric</servlet-name>  
 <url-pattern>/HelloGeneric</url-pattern>  
 </servlet-mapping>**

1. <servlet>  
     
    <init-param>  
    <param-name>count</param-name>  
    <param-value>13</param-value>  
    </init-param>  
    <servlet-name>HelloGeneric</servlet-name>  
    <servlet-class>HelloGeneric</servlet-class>  
    </servlet>  
    <servlet-mapping>  
    <servlet-name>HelloGeneric</servlet-name>  
    <url-pattern>/HelloGeneric</url-pattern>  
    </servlet-mapping>
2. <servlet>  
     
    <servlet-name>HelloGeneric</servlet-name>  
     
    <init-param>  
    <param-name>count</param-name>  
    <param-value>13</param-value>  
    </init-param>  
     
    <servlet-class>HelloGeneric</servlet-class>  
      
    </servlet>  
     
    <servlet-mapping>  
    <servlet-name>HelloGeneric</servlet-name>  
    <url-pattern>/HelloGeneric</url-pattern>  
    </servlet-mapping>
3. <servlet>  
     
    <servlet-name>HelloGeneric</servlet-name>  
    <servlet-class>HelloGeneric</servlet-class>  
      
    </servlet>  
     
    <servlet-mapping>  
     
    <servlet-name>HelloGeneric</servlet-name>  
    <url-pattern>/HelloGeneric</url-pattern>  
    <init-param>  
    <param-name>count</param-name>  
    <param-value>13</param-value>  
    </init-param>  
   </servlet-mapping>
4. <init-param>  
    <param-name>count</param-name>  
    <param-value>13</param-value>  
    </init-param>  
     
   <servlet>  
     
    <servlet-name>HelloGeneric</servlet-name>  
    <servlet-class>HelloGeneric</servlet-class>  
      
    </servlet>  
     
    <servlet-mapping>  
     
    <servlet-name>HelloGeneric</servlet-name>  
    <url-pattern>/HelloGeneric</url-pattern>  
     
   </servlet-mapping>

17. **What are the implications of using the HTTP GET method for a form submission?  
Select three answers from the following.**

1. **You cannot pass binary data to the server.**
2. **You cannot send unlimited (or a lot of) data to the server**
3. You cannot send multiple values for one parameter to the server.
4. You can only reply with the HEADER information in the response.

**e) The parameters will be appended to the URL as a query string**

**18. Consider the following HTML page code:  
  
<html><body>  
<a href="/servlet/HelloServlet">POST</a>  
</body></html>  
Which one of the following methods of HelloServlet will be invoked when the hyperlink displayed by the above page is clicked?**

1. **doGet**
2. doPost
3. doForm
4. doHref
5. serviceGet

19. **.Consider the following URL:  
http://www.godaddy.com/servlet/HelloWorld;jsessionid=ab123f556  
The above is a correct example of \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.**

1. use of URL-rewriting for authentication
2. **use of URL-rewriting for session tracking**
3. use of html hidden parameters for session tracking
4. use of cookie based session tracking.

use of HttpSession for session tracking

20. **.<web-app>  
 ...  
 <session-config>  
 <session-timeout>15</session-timeout>  
 </session-config>  
....  
What does the above entry in the Deployment Descriptor cause?  
Select one answer from the following.**

1. Only allows users to be connected for 15 days total.
2. **Sets the session timeout interval to 15 minutes.**
3. Only allows users to be connected for 15 minutes total
4. Sets the session timeout interval to 15 days.
5. Sets the session timeout interval to 15 seconds.

21. **Which one of the following exceptions may be thrown by the \_jspService() method?**

1. javax.servlet.ServletException
2. javax.servlet.jsp.JSPException
3. javax.servlet.ServletException and javax.servlet.jsp.JSPException
4. **javax.servlet.ServletException and java.io.IOException**
5. javax.servlet.jsp.JSPException and java.io.IOException

22. **Which one of the following is a correct way to pass a parameter equivalent to the query string user=rani at request time to an included component?**

1. <jsp:include page="other.jsp" >  
   <jsp:param paramName="user" paramValue="rani" />  
   </jsp:include>
2. <jsp:include page="other.jsp" >  
   <jsp:param name="rani" value="user" />  
   </jsp:include>
3. **<jsp:include page="other.jsp" >  
   <jsp:param value="rani" name="user" />  
   </jsp:include>**
4. <jsp:include page="other.jsp" >  
   <jsp:param param="user" value="rani"/>  
   </jsp:include>
5. <jsp:include page="other.jsp" >  
   <jsp:param user="rani" />  
   </jsp:include>

23. **Consider the following scenario:  
  
Statement A: <jsp:include/> standard action inserts the response of a jsp page at rutime.  
Statement B: include directive inserts the source of a jsp page at translation time.  
  
Which one of the following is correct with respect to the above?**

1. Statement A is true and B is false
2. Statement A is false and B is false
3. **Statement A is true and B is true**
4. Statement A is false and B is true

24. **Which two of the following statements are true?**

1. **The default value of isThreadSafe attribute of page directive is true.**
2. If isThreadSafe attribute of page directive is set to true, then JSP container dispatches request for the page sequentially.
3. **When isThreadSafe attribute of page directive is set to true, a thread is created for each request for the page.**
4. Setting isThreadSage attribute to true for JSP pages, can lead to poor performance.
5. The default value of isThreadSafe attribute is false.

25.  **Given a web page that is requested with the query string ?username=yogesh, which two of the following can result in the output in the page of the string yogesh?**

1. **${param.username}**
2. ${request.yogesh}
3. ${param[yogesh]}
4. ${request["yogesh"]}
5. **${param["username"]}**

**26. The page directive is used to convey information about the page to JSP container.   
Which two of the following are legal syntax of page directive?**

1. <% page info="test page" %>
2. **<%@ page info="test page" session="false"%>**
3. **<%@ page session="true" %>**
4. <%@ page isErrorPage="errorPage.jsp" %>
5. <%@ page isThreadSafe=true %>

**27. . Consider the following scenario:  
  
An existing JSP page is modified.  
The following steps happen when the modified page is requested:  
(a) jspInit()  
(b) jspDestroy()  
(c ) translation  
(d) Load the class  
(e) \_jspService()  
(f) Create an Instance  
(g) Compile the translated code  
  
Which one of the following options gives the correct sequence in which the above steps happen?**

1. a,b,c,d,e,f,g
2. a,c,g,f,d,e,b
3. c,g,f,d,a,e,b
4. **c,g,d,f,a,e,b**

**28. Which two of the following statements are true?**

1. **<%! int i; %> is a variable declaration that has object scope within the servlet**
2. **<% int i = 0; %> is a variable declaration that has local scope within the service method of the servlet.**
3. <% int i = 0; %> is a variable declaration that has local scope within the doGet method of the servlet.
4. <%! int i; > a variable declaration that has local scope within the service method of the servlet
5. <% int i; %> is a variable declaration that is class scope to the service method.

**29. Which one of the following gives the correct sequence of steps involved in JSP lifecycle?**

1. jspinit(),jspdestory(),\_jspservice(),translation,compilation,load class,create an instance
2. jspinit(),\_jspservice(),jspdestory(),translation,compilation,load class,create an nstance
3. **translation,compilation,load class,create an instance,jspinit(),\_jspservice(),jspdestory()**
4. translation,compilation,create an instance,load class jspinit(),jspdestory(),\_jspservice()

**30. Which one of the following comments used in JSP page will be displayed in the response to the user?**

1. **<! --this is html comment-->**
2. <%-- this is jsp comment--%>
3. <%/\* this is java comment \*/%>

<% this is jspscriplet comment%>

31. **Consider the following JSP code:   
  
<html><body>  
  
<%  
  
 Integer count = (Integer) request.getSession(false).getAttribute("count");  
  
 if(count != null )  
  
 {  
  
 out.println(count);  
  
 }  
  
 else request.getSession(false).setAttribute("count", new Integer(1));  
  
%>  
  
Hello!  
  
</body></html>  
  
Which one of the following will be printed for the very first request to the above page as well as the web application that contains this page?**

1. It will print Hello!
2. **It will print Hello and will set the count attribute in the session.**
3. It will throw a NullPointerException at request time.
4. It will not compile.

**32**. **. What will be the output of the following JSP code snippet at run-time when it is accessed the third time?  
  
<% int test = 0; %>  
<% ++test; %>  
The value of test is   
<%= test %>  
(select one answer)**

1. The value of the 'test' variable will be 2.
2. The value of the 'test' variable will be 0.
3. **The value of the 'test' variable will be 1.**
4. The variable 'test' must be declared at global scope, else a run-time error will occur.

33. **Which three of the following are implicit objects available to every JSP page by default?**

1. **session**
2. context
3. system
4. **out**
5. **request**

**34.** **Consider the following JSP code:  
  
  
<%@page language=”java”%>  
<html>  
<body>  
  
<%! int x=0;%>  
  
<%! public void jspinit()  
{  
System.out.println(x++);  
}%>  
  
<%=x++%>  
  
<%System.out.println(x);%>  
  
<%jspInit();%>  
</body>  
</html>  
  
What is output to the web page on the second access to the same instance of the above JSP?  
Select one answer from the following.**

1. 0
2. 1
3. 2
4. **3**
5. 4

**35.** **Consider the following code:  
<html><body>  
<jsp:useBean id="address" class="AddressBean" scope="session" />  
state = <jsp:getProperty name="address" property="state" />  
</body></html>  
  
Which three of the following are equivalent to the third line above?**

1. <% state = address.getState(); %>
2. **<% out.write("state = "); out.print(address.getState()); %>**
3. <% out.write("state = "); out.print(address.getstate()); %>
4. **<% out.print("state = " + address.getState()); %>**
5. **state = <%= address.getState() %>**

36. **Which two of the following statements are true?**

1. **The jspInit method has access to the ServletConfig and ServletContext objects.**
2. The jspInit method is called before each time \_jspService is called
3. **The jspInit method is called once and only once for a servlet instance**
4. An overridden jspInit method will not be called by the container

**37**. **Consider the following code for address.jsp:  
  
<html><body>  
<jsp:useBean id="address" class="AddressBean" />  
<jsp:setProperty name="address" property="city" value="LosAngeles" />  
<jsp:setProperty name="address" property="city" />  
<jsp:getProperty name="address" property="city" />  
</body></html>  
  
Also consider the URL - http://localhost:8080/guest/address.jsp?city=Chicago&city=Miami  
  
(Assume that the city property is not an indexed property.)   
  
Which one of the following is the output if the above page is accessed via the mentioned URL?**

1. LosAngeles
2. **Chicago**
3. Miami
4. ChicagoMiami
5. LosAngelesChicagoMaimi

38. **. Given the following code:  
  
package com.smilejava;  
public class JobBean {  
 private String chef,waiter,cleaner;  
 public JobBean() {  
 chef="jim";  
 waiter="bob";  
 cleaner="fred";  
 }  
 public void setChef(String chef){  
 this.chef=chef;  
 }  
 public void setWaiter(String waiter){  
 this.waiter=waiter;  
 }  
 public void setCleaner(String cleaner){  
 this.cleaner=cleaner;  
 }  
 public String getJobs(){  
 return chef+waiter+cleaner;  
 }  
}  
  
And the following JSP page that uses the code:  
  
<jsp:useBean id="j" class="com.smilejava.JobBean"/>  
<jsp:setProperty name="j" property ="\*"/>  
<jsp:getProperty name="j" property="jobs" />  
<jsp:setProperty name="j" property="waiter" param="a"/>  
<jsp:setProperty name="j" property="cleaner" param="b"/>  
<jsp:setProperty name="j" property="chef" param="c"/>  
<jsp:getProperty name="j" property="jobs" />  
  
What will be output by that JSP code when invoked with the query string  
?a=tom&b=dick&c=harry  
  
Select one answer from the following.**

1. fredjimbob harrytomdick
2. jimbobfred tomdickharry
3. harrytomdick jimbobfred

**d) jimbobfred harrytomdick**

39.  **Consider the following scenario:  
  
Ajay is using JSP pages to create his application.  
He is using useBean tag(to include a student bean with attribute such as name and age) in JSP.  
After using the attributes of the bean in page1.jsp, request is forwarded to page2.jsp.  
The tag he used in page1 is <jsp:useBean id=”stu” class=”student” />   
  
Is the bean accessible in page2.jsp?  
  
State Yes or No.**

1. Yes
2. **No**

40. **Which one of the following gets or sets the bean in the ServletContext container object?**

1. <jsp:useBean id="address" class="AddressBean" />
2. **<jsp:useBean id="address" class="AddressBean" scope="application" />**
3. <jsp:useBean id="address" class="AddressBean" scope="servlet" />
4. <jsp:useBean id="address" class="AddressBean" scope="session" />
5. None of the listed options

41.  **When <jsp:useBean id="objectName" scope="scopeAttribute" class="ClassName" /> is processed, the JSP engine first searches for the object of the class with the same id and scope. If found, the preexisting bean is used; otherwise, a new bean is created.  
State True or False.**

1. **True**
2. False

42.  **Consider the following code:  
public class MyTagHandler extends TagSupport   
  
{  
  
 public int doStartTag() throws JspException  
  
 {  
  
 try  
  
 {  
  
 //insert code here  
  
 }  
  
 catch(Exception e){ }  
  
 return super.doStartTag();  
  
 }  
  
}  
  
Which one of the following options, when inserted in the above code, causes the value "hello" to be output?**

1. **JspWriter out = pageContext.getOut();  
   out.print("hello");**
2. JspWriter out = pageContext.getWriter();  
   out.print("hello");
3. JspWriter out = getPageContext().getWriter();  
   out.print("hello");
4. JspWriter out = new JspWriter(pageContext.getWriter());  
   out.print("hello");
5. JspWriter out = getPageContext().getOut();  
   out.print("hello");

43.  **Which two of the following are valid custom tag usages, assuming that the correct taglib directives have been set up?**

1. **<mg:claim>amount</mg:claim>**
2. <claim>amount</claim>
3. <mg:claim>amount<mg:claim/>
4. **<mg:claim></mg:claim>**

44. **Which one of the following statements is true?**

1. TLD files must be stored under the WEB-INF/tld directory
2. TLD files must be stored under the WEB-INF/lib directory
3. **TLD files must be stored under the WEB-INF/ directory or a directory under WEB-INF**
4. TLD files can be stored anywhere within the application path

45. **The following is an entry from Avarice's DateTime Tag-lib Descriptor:   
  
 <tag>  
 <name>currenttime</name>   
 <tag-class>org.apache.taglibs.datetime.CurrentTimeTag</tag-class>   
 <body-content>empty</body-content>   
 <description>Gets the current time in milliseconds since Jan 1, 1970 GMT.</description>  
 </tag>  
  
Which two of the following are true statements about the above tag?**

1. **This is an example of an empty custom tag.**
2. This is a nested tag.
3. **This tag has no attributes**
4. This tag can surround other JSP code.

46. Consider the following JSP code:

<html>

<head>

</head>

<body>

<% for (int i = 0; i < 5 ; i++)

{ %>

out.print(i);

<% } %>

</body>

</html>

Which one of the following will be printed in the generated response?

a) This page will not compile.

b) It will compile but will not generate any output.

c) It will print 0 to 4.

d) It will throw an exception at runtime

**e) out.print(i); out.print(i); out.print(i); out.print(i); out.print(i);**