L1 JEE MOD 3 Practice Test

#### Section 1: Developer Workbench

**Q.** Match the following:  
  
A. version  
B. artifactId  
C. groupId  
  
  
1. This is an Id of project's group. This is generally unique amongst an organization or a project.   
For example, a banking group com.company.bank has all bank related projects.  
  
2. This is an Id of the project .This is generally name of the project. For example, consumer-banking.   
Along with the groupId, the artifactId defines the artifact's location within the repository.   
  
3. This is the version of the project.Along with the groupId, It is used within an artifact's repository   
to separate versions from each other.

**1)**. **A-1, B-2, C-3**

**2)**. **A-3, B-2, C-1**

**3)**. A-2, B-3, C-1

**4)**. A-2, B-1, C-3

##### Q. \_\_\_\_  is used to create a simple java application in Maven.

### Attempted

1. maven-archetype-quickstart

### Solution

1. maven-archetype-quickstart

2. maven-archetype-jee

3. maven-archetype-mojo

4. maven-archetype-java

**Q.** Consider the below code snippet exists in pom.xml file  
  
<project>  
  <modelVersion>4.0.0</modelVersion>  
  <groupId>com.cg.bank</groupId>  
  <artifactId>bankapp</artifactId>  
  <version>1.0</version>  
</project>  
  
What type of packaged artifacts will get created when the above mentioned artifacts is packaged   
using "mvn package" command?

**1)**. EAR

**2)**. WAR

**3)**. **JAR**

**4)**. **SAR**

**Q.** PMD helps to identify \_\_\_\_\_\_\_\_ and \_\_\_\_\_\_\_\_.

**1)**. **Empty try/catch/finally**

**2)**. **Unnecessary Object creation**

**3)**. **Logical Error**

**4)**. Solution to program bugs.

#### Section 2: JSP

**Q.**Consider the following code snippets. Which of them is correct syntax for page directive?

**1)**. <% page info="test page" %>

**2)**. **<%@page info="test page" %>**

**3)**. **<%@ page session="true" %>**

**4)**. <%@ page isErrorPage="errorPage.jsp" %>

**Q.** What following JSP snippet will produce?  
  
<%  
methodX();  
public void methodX()  
{  
      out.println("Hello ");  
}  
%>

**1)**. **It will print Hello at the output.**

**2)**. It will produce an error as function is called before defining it.

**3)**. **It will produce an error as function is defined in scriptlet.**

**4)**. It will produce an error as JSP does not allow methods to be defined

**Q.** <%= "HelloWorld" %>  
  
This is an example of:

**1)**. jsp directive

**2)**. **jsp expression**

**3)**. **jsp scriptlet**

**4)**. jsp declaration

**Q.**My.jsp is loaded and executed twice, then it is edited. After this the server is restarted   
and now when My.jsp is requested,  which of the following steps will be performed?

**1)**. **JSP translated to Servlet and compiled**

**2)**. **Servlet loaded into memory**

**3)**. **jspInit called**

**4)**. **\_jspService called**

**Q.** <%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>   
<c:set var="cnt" value="2"/>   
<c:choose>   
     <c:when test="${cnt>0}">   
          <c:out value="1"/>   
     </c:when>   
     <c:when test="${cnt==1}">   
            <c:out value="2"/>   
     </c:when>   
     <c:when test="${cnt<2}">   
               <c:out value="3"/>   
      </c:when>   
      <c:otherwise>   
              <c:out value="4"/>   
       </c:otherwise>   
</c:choose>   
  
What is the output of the above code snippet?

**1)**. The code will not compile because of incorrect syntax

**2)**. **The output cannot be determined**

**3)**. The code will print 4

**4)**. **It prints 1**

**Q.** What gets printed when the following JSTL code fragment is executed?   
  
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>   
<c:set var="item" value="2"/>   
<c:forEach var="item" begin="0" end="0" step="2">  
<c:out value="${item}" default="abc"/>  
</c:forEach>

**1)**. The JSTL code does not compile as an attribute for forEach tag is not correct.

**2)**. **Nothing gets printed as c:out statement does not get executed.**

**3)**. **0**

**4)**. 2

**5)**. ABC

**Q.** <c:set var="city" value="${user.address.city}" default="N/A" />  
  
What does the above line indicate ?

**1)**. **Variable city is assigned value of user bean's property**

**2)**. **Variable called 'city' is set with the default value of 'N/A'**

**3)**. **Bean property address is set to 'city'**

**4)**. This tag will throw null pointer exception if user object is not in any of the scopes.

**Q.**Which of the following code snippets shows the most appropriate use of <jsp:forward> ?

**1)**. **<jsp:forward page="/pages/register.jsp">  
     <jsp:param name="user" value="John"/>  
   </jsp:forward>**

**2)**. **<jsp:forward page="/pages/register.html">  
     <jsp:param name="user" value="John"/>  
     </jsp:forward>**

**3)**. **<jsp:forward page="/registerServlet">  
     <jsp:param name="user" value="John"/>  
     </jsp:forward>**

**4)**. <jsp:forward page="/registerServlet"/>  
     <jsp:param name="user" value="John"/>

**Q.** What is the default scope  attribute value in<jsp:useBean> action tag?

**1)**. session

**2)**. **page**

**3)**. application

**4)**. **request**

**5)**. context

**Q.** A \_\_\_\_\_\_\_\_\_is the one that documents the JSP page but is not sent to the client.

**1)**. **Hidden comment**

**2)**. Comment

**3)**. **Compiler comment**

**4)**. Runtime comment

**Q**. Consider the following code snippet. What will be the output?  
  
<%= if(Math.random() < 0.5) %>  
  hello  
<%= } else { %>  
  hi  
<%= } %>

**1)**. **It will always print hello**

**2)**. It will always print hi

**3)**. It will print either hello or hi depending on the return value of random

**4)**. **The JSP file has errors and will not compile**

**Q.** <%@ page implements = "java.lang.Runnable" %>  
  
Is the above statement valid ?

**1)**. True

**2)**. **False**

**Q.**What is the effect of executing the following JSP statement, assuming a class with name   
Employee exists in classes package?  
  
<%@ page import = "classes.Employee" %>   
<jsp:useBean id="employee" class="classes.Employee" scope="session"/>   
<jsp:setProperty name="employee" property="\*"/>

**1)**. The code does not compile as there is no property attribute of setProperty tag.

**2)**. **The code does not compile as property attribute cannot take \* as a value.**

**3)**. The code sets value of all properties of employee bean to "\*".

**4)**. **The code sets the values of all properties of employee bean to matching parameters in request object.**

**Q.** Given:  
  
//Code for error.jsp  
This is an Error Page  
  
//Code for test.jsp  
<%@page errorPage = "error.jsp" %>  
<%!  
 int arr[]= {5,6,7,8};  
 %>  
 <%  
  for(int i=0;i<4;i++) {  
   int val = arr[i];  
 %>  
  <%=val%>  
 <%  
  }  
 %>  
  
What will be the output of the above code fragment?

**1)**. This is an Error Page

**2)**. 5 6 7

**3)**. **5 6 7 8**

**4)**. **Some Error raised by JSP Engine**

**Q.** What following code will produce?   
  
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>  
<br>  
<hr>  
<c:forEach var="i" begin="10" end="1" step="-2">  
      <c:out value="${i}" />  
      <br />  
</c:forEach>

**1)**. Error, since begin is greater than end

**2)**. **Error, since step is < 0**

**3)**. It will print 10 8 6 4 2

**4)**. It will produc error: Invalid use of EL

**Q.** What will be printed out if this is the jsp code?   
  
<% java.util.Locale s=request.getLocale() %>  
<%= s %>

**1)**. **It will not compile.**

**2)**. **It will print the default client's Web browser locale.**

**3)**. It will print the default web server Locale.

**4)**. It won't print the default Locale but the first language specified in the Accept-Language request's header

#### Section 3: Servlets

**Q.**To create a war file testApp.war  from the folder testApp, the jar command to be used is:

**1)**. **jar cvf testApp.war \***

**2)**. jar xvf testApp.war \*

**3)**. **jar xvf testApp.war**

**4)**. jar c testApp.war \*

**Q.** Point out error(s) if any in the following code snippet:  
  
import javax.servlet.\*;  
import javax.servlet.http.\*;  
class MyServlet extends HttpServlet {  
   public void DOGET(){  
     System.out.println("Inside DOGET");  
  }  
   public static void main(String args[]) {  
     System.out.println("Inside DOGET");  
   }  
}

**1)**. **Compilation Error indicating invalid signature and name of method DOGET**

**2)**. **No Compilation Error**

**3)**. Compilation Error indicating Servlet class must not contain main() method

**4)**. Compilation Error indicating MyServlet class should be declared as public

**Q.** Which of the following statements are TRUE for the code given below?  
  
int MAX\_AGE;   
Cookie cookie = new Cookie("user", user);   
cookie.setMaxAge(MAX\_AGE);   
response.addCookie(cookie);

**1)**. **If MAX\_AGE = 10 the cookie will expire after 10 seconds.**

**2)**. **If MAX\_AGE = 10 the cookie will expire after 600 seconds.**

**3)**. **If MAX\_AGE = 0 the cookie will be deleted.**

**Q.**Select the correct statement about the attribute fileSizeThreshold of the annotation @MultipartConfig:

**1)**. Default size is 10 bytes which means files will be immediately written without caching.

**2)**. **The fileSizeThreshold is a byte limit above which the file will be saved to cached in RAM.**

**3)**. **The file size in bytes after which the file will be temporarily stored on disk. The default size is 0 bytes.**

**4)**. Unless we intend to save all the uploads to the hard drive, you may wish to set this larger than   
the  default to avoid using slower disk -based access when reading the file contents.

**Q.** import java.io.\*;  
import javax.servlet.\*;  
import javax.servlet.http.\*;  
public class HelloWorld extends HttpServlet   
{  
   public void doGet(HttpServletRequest request,HttpServletResponse response)   
    {  
          PrintWriter print = response.getWriter(); print.println("Hello World");  
    }  
}  
  
What is true about the above code ?

**1)**. **Program executes and displays a message Hello World**

**2)**. Program does not compile as the implicit object out is not specified for printing the message

**3)**. **Program does not execute as the doGet() does not throw Servlet and IO Exceptions**

**4)**. Program may execute, but instead of the doGet(), the doPost should be used if printing is to be done.

**Q.** We have following methods declared in our servlet:  
  
public void service(HttpServletRequest request, HttpServletResponse response)   
{ System.out.println("Service Method"); }  
  
public void doGet(HttpServletRequest request, HttpServletResponse response)   
{ System.out.println("Get Method"); }  
  
public void doPost(HttpServletRequest request, HttpServletResponse response)   
{ System.out.println("Post Method"); }  
This servlet is invoked by an HTML in which we have specified method="post"   
  
What output will it show on the console?

**1)**. **Service Method   
Get Method  
Post Method**

**2)**. Service Method  
Post Method

**3)**. **Service Method**

**4)**. Post Method

**Q.**In the deployment descriptor, if we have following entries for our servlet "MyServlet"  
  
<servlet>  
 <servlet-name> MyServlet1 </servlet-name>  
 <servlet-class> MyServlet </servlet-class>  
</servlet>  
<servlet-mapping>  
 <servlet-name> MyServlet1 </servlet-name>  
 <url-pattern>/1 </url-pattern>  
 <url-pattern>/2 </url-pattern>  
</servlet-mapping>  
  
Which of the following statements are correct?

**1)**. **It will create one servlet object**

**2)**. **It will create two servlet objects**

**3)**. **Above entries in deployment descriptor are valid**

**4)**. Above entries in deployment descriptor are invalid as we can not have multiple   
<url-pattern> tags in <servlet-mapping>

**Q.**Following is the code for doGet() method of TestServlet. Which of the given statements   
about it are correct?  
  
public void doGet(HttpServletRequest req, HttpServletResponse res)  
{  
    try  
    {  
        RequestDispatcher rd = this.getServletContext().getRequestDispatcher("Login.jsp"); // 1  
        rd.forward(req, res); // 2  
    }  
    catch(Exception e)  
    {  
        e.printStackTrace();  
    }  
}

**1)**. This will not compile.

**2)**. **This will compile but will give runtime error.**

**3)**. **This will compile and work properly.**

**4)**. **This will compile. However, it will work properly only if //1 is replaced with:  
RequestDispatcher rd = req.getRequestDispatcher("Login.jsp");**

**Q.** @MultipartConfig defines \_\_\_\_\_\_\_\_\_\_.

**1)**. security constraints

**2)**. **Listener**

**3)**. **file upload properties**

**4)**. Servlet

**Q.**Which of the following is the correct function  in a servlet  to get  all the cookie from client?

**1)**. **request.getCookies()**

**2)**. response.getCookies()

**3)**. **request.getCookie()**

**4)**. response.getCookie()

**Q.** @WebServlet("/FileUploadServlet")  
@MultipartConfig(fileSizeThreshold=1024\*1024\*10,    // 10 MB   
                 maxFileSize=1024\*1024\*50,          // 50 MB  
                 maxRequestSize=1024\*1024\*100)      // 100 MB  
public class FileUploadServlet extends HttpServlet {  
  
   protected void doGet(HttpServletRequest request,  
            HttpServletResponse response) throws ServletException, IOException {  
  
             }  
}  
  
Which exception above servlet will generate if we upload the file of size 150MB?

**1)**. FileSizeException

**2)**. **FileNotFoundException**

**3)**. All of these

**4)**. **IOException**

**Q.** Select the correct order of the invocation of service functions in servlet.

**1)**. **public void  service(..)    
    either void doGet(..) Or doPost(..)**

**2)**.        either void doGet(..) Or doPost(..)  
private void service(..)

**3)**. **either void doGet(..) Or void  doPost(..)  
public  void service(..)**

**4)**. None of the above

**Q.** Assumption:  
  
5 web applications are deployed under Jboss application server and each of them   
consists of 2 servlets. How many instances of type ServletContext will be created   
when the server gets started?

**1)**. 10

**2)**. **5**

**3)**. 1

**4)**. **25**

**Q.** In a J2EE enabled application, \_\_\_\_\_\_\_\_\_\_ is a component that acts as an interceptor.

**1)**. Servlet

**2)**. **Listener**

**3)**. **Filter**

**4)**. JSP Page

**Q.** @WebServlet("/my-address")  
public class MyServlet extends HttpServlet { ... }  
  
Choose the correct URL.

**1)**. **http://hostname/appname/my-address**

**2)**. http://hostname/my-address

**3)**. http://appname/my-address

**4)**. http://hostname/appname