Oracle 1z0-899

BrainDumps

Java EE 6 Web Component Developer Certified Expert Exam

Exam

Version: 4.2

QUESTION NO: 1

Given the element from the web application deployment descriptor:

<jsp – property – group>

<url – pattern>/main/page1.jsp</url – pattern>

<scripting – invalid>true</scripting – invalid>

</jsp—property—group>

And given that /main/page1.jsp contains:

<% int i = 12; %>

 <%= i %>

What is the result?

A.
B. l2
C. The JSP fails to execute.
D. <% int i = 12 %>
<b> <%= i % > < b>

Answer: C
Explanation:

QUESTION NO: 2

A web application allows the HTML title banner to be set using a context initialization parameter called titlestr.

Which two properly set the title in the scenario? (Choose two)

A. <title> $ {titlestr} </title>
B. <title> $ {initparam.titlestr}</title>
C. <title> $ {param [0]. titlestr} </title>
D. <title> $ {paramValues.titleStr} </title>
E. <title> $ {initParam ['titleStr'] } </title>
F. <title> $ {servletParams.titleStr} </title>

G. <title> $ {request.get ("titleStr") } </title>

Answer: B,E
Explanation:

QUESTION NO: 3

Given:

11. <%

12. request.setAttribute ("vals", new String[] {"1", "2", "3", "4"});

13. request.setAttribute ("index", "2");

14. %>

15. <% - - insert code here - - %>

Which three EL expressions, inserted at line 15, are valid and evaluate to "3"? (Choose three)

A. ${vals.2}
B. ${vals ["2"] }
C. ${vals.index}
D. ${vals[index] }
E. ${vals} [index]
F. ${vals. (vals.index) }
G. ${vals [vals[index-1]] }

Answer: B,D,G
Explanation:

QUESTION NO: 4

Given:

```
11.  <% java.util.Map map = new java.util.HashMap();
12.      request.setAttribute("map", map);
13.      map.put("a", "b");
14.      map.put("b","c");
15.      map.put("c", "d"); %>
16.  <%-- insert code here --%>
```

Which three EL expressions, inserted at line 16, are valid and evaluate to "d"? (Choose three)

A. ${map.c}

B. ${map.[c]}

C. ${map.["c"]}

D. ${map.map.b}

E. ${map.[map.b]}

F. ${map. (map.b)}

Answer: A,C,E

Explanation:

QUESTION NO: 5

You are building a dating service web site. Part of the form to submit a client's profile is a group of radio buttons for the person's hobbies:

20. <input type = 'radio' name = 'hobbyEnum' value = 'HIKING'> Hiking

21. <input type = 'radio' name = 'hobbyEnum' value = 'SKING'> Sking

22. <input type = 'radio' name = 'hobbyEnum' value = 'SCUBA'> SCUBA

23. <! - - and more options - - >>

After the user submits this form, a confirmation screen is displayed with these hobbies listed. Assume that an application-scoped hobbies, holds a map between the hobby enumerated type and the display name.

Which EL code snippet will display Nth element of the user's selected hobbles?

A. ${hobbies [hobbyEnum[N]}

B. ${hobbies [paramValues.hobbyEnum[N]]}

C. ${hobbies [paramValues @ 'hobbyEnum' @N]

D. ${hobbies.get(paramValues.hobbyEnum[N]) }

E. ${hobbies [paramValues.hobbyEnum.get(N)] }

Answer: B

Explanation:

QUESTION NO: 6

Given:

1. <% int[] nums = {42,420,4200};

2. request.setAttribute("foo", nums); %>

3. ${5 + 3 lt 6}

4. S(requestScope['foo'][0] ne 10 div0>

5. ${10 div 0}

What is the result?

A. true true
B. false true
C. false true 0
D. true true Infinity
E. false true Infinity
F. An exception is thrown
G. Compilation or translation fails

Answer: E
Explanation:

QUESTION NO: 7

You are building a web application with a scheduling component. On the JSP, you need to show the current date, the date of the previous week, and the date of the next week.

To help you present this information, you have created the following EL functions in the 'd' namespace:

- Name : curData; signature: java.util.utilDate CurrentDate()
- Name : curData; signature: java.util.utilDate addweek (java.util.Date, int)
- Name: dateString; signature: java.util.String getDataString (jave.util.Data)

Which EL code snippet will generate the string for the previous week?

A. ${d:dateString(affWeek(curDate(), -1)) }
B. ${d:dateString [addWeek[curDate[], -]] }

C. ${d:dateString [d:addWeek[d:curDate[], - 1]] }
D. ${d:dateString (d:addWeek(d:curDate(), -1)) }

Answer: D
Explanation:

QUESTION NO: 8

Given a header in an HI IP request:

X-Retries: 4

A Which two retrieve the value of the header from a given HttpServletRequest request? (Choose two)

A. request-getHeader("X-Retries")
B. request.getIntHeader("X-Retries")
C. request.getRequestHeader("X-Retries")
D. request.getHeaders{"X-Retries").get(0)
E. request.getRequestHeaders("X-Retries").get(0)

Answer: A,B
Explanation:

QUESTION NO: 9

Given an HttpServletRequestrequest and HttpResponseresponse, which sets a cookie "username" with the value "joe" in a servlet?

A. request.addCookie("username", "joe")
B. request.setCookie("username", "joe")
C. response.addCookie("username", "joe")
D. request.addHeader(new Cookie("username", "joe"))
E. request.addCookie(new Cookie("username", "joe"))
F. response.addCookie(new Cookie("username", "joe"))
G. response.addHeader(new Cookie("username", "joe"))

Answer: F
Explanation:

QUESTION NO: 10

Which annotation enables a servlet to efficiently process requests of typo multipart/form-data that involve large files being uploaded by the client?

A. @AcceptMultipart
B. @MultiPartConfig
C. @MultiPartFormData
D. @WebServlet (multipart = true)

Answer: B
Reference: http://www.scribd.com/ilinchen2008/d/38764279-Servlet3-0-Specs (page 22, last paragraph)

QUESTION NO: 11

APIs for dynamically registering servlets and filters can be called at:

A. Context initialization time
B. Any time
C. There is no way to dynamically register servlets and filters
D. Context destruction time
E. Injected into a servlet context

Answer: A
Explanation:

QUESTION NO: 12

ServletRegistration.Dynamic can be used to configure (Choose two)

A. Filter mappings
B. add init params
C. set asyncSupported to true
D. add Listeners

Answer: A,B

Reference: http://www.softwareengineeringsolutions.com/blogs/2010/08/01/ (3rd paragraph)

QUESTION NO: 13

A web application wants to register a new servlet at runtime using one of the servletContext.addServlet methods.

In the implementation of which method is it legal to perform this operation?

A. AsyncListner.oncomplete
B. Servlet.init
C. ServletContextListener.contextInitialize
D. ServletrequestListener.contextInitialize

Answer: C
Explanation:

QUESTION NO: 14

You want to create a filter for your web application and your filter will implement

.servlet.Filter.

Which two statements are true? (Choose two)

A. Your filter class must implement an init method and a destroy method.
B. Your filter class must also implement javax.sarvlat.FilterChain.
C. When your filter chains to the next filter, it should pass the same arguments it received in its doFilter method.
D. The method that your filter invokes on the object it received that implements
javax – servlet.Filterchain can invoke either another filter or a servlet.
E. Your filter class must implement a doFilter method that takes, among other things, an HttpServletRequest object and an HttpServletresponse object.

Answer: A,D
Explanation:

QUESTION NO: 15

You have a simple wpb application that has a single Front Controller servlet that dispatches to JSPs generate a variety of views. Several of these views require further database processing to retrieve the necessary order object using the orderID request parameter. To do this additional processing, you pass the request first to a servlet that is mapped to the URL pattern /WEB – INF / retrieveOrder.do. in the deployment descriptor. This servlet takes two request parameters, the ordered and the jspID and the jspURL. It handles the database calls to retrieve and build the complex order objects and then it dispatches to the jspURL.

Which code snippet in the Front Controller servlet dispatches the request to the order retrieval servlet?

A. reques.setAttribute ("orderID", orderIS);
request.setAttribute("jspURL", jspURL);
= context.getRequestDispathcher ("/WEB – INF / retrieveOrder.do");
view.forward(request, response)
B. reques.setAttribute ("orderID", orderIS);
request.setAttribute("jspURL", jspURL);
Dispatcher view
= request.getDispatcher (".WEB – INF / retrieveOrder.do");
View.forwardRequest (request, response);
C. String T= "/WEB – INF / retrieveOrder.do?orderID = %d&jspURI = %s";
String url = String.format (T, ordered, jspURL);
= context.getRequestDispatcher (url);
View.forward (request, response) ;
D. String T= "/WEB – INF / retrieveOrder.do?orderID = %d&jspURI = %s";
String url = String.format (T, ordered, jspURL);
= context.getRequestDispatcher (url);
View.forwardRequest (request, response) ;

Answer: C
Explanation:

QUESTION NO: 16

In ServletContext.getNamedDispatcher(string arg0) arg0 is the

A. url of a servlet relative to the ServletRequest
B. url of a servlet relative to the Context
C. name of a servlet

D. absolute url to any servlet in any Context

Answer: B
Explanation:

QUESTION NO: 17

When using Servlet asynchronous API if you want to dispatch the request back to a particular url - "/url" within the same Servlet Context which of the following API would you use?

A. ASyncContext.dispatch();
B. AsyncContext.dispatch("/url");
C. AsyncContext.dispatch(servletContext, "/url");
D. AsyncContext.start(runnable);
E. RequestDispatcher.fotward("/url");
F. RequestDispatcher.forward(servletContext, "/url");
G. RequestDispatcher.include("/url");

Answer: C
Reference: http://blogs.oracle.com/enterprisetechtips/entry/asynchronous_support_in_servlet_3
(Topic: AsyncContext Class, third paragraph)

QUESTION NO: 18

Which describes a trigger that causes a call to an AsyncListener?

A. Completion of processing
B. Receipt of a new request from the same client
C. Addition of a new filter to the processing stream for this request
D. Completion of each timer interval

Answer: B
Explanation:

QUESTION NO: 19

Refer to the Exhibit.

```
 1.  package com.example;
 2.
 3.  public class Product {
 4.      private String name;
 5.      private double price;
 6.
 7.      public Product() {
 8.          this( "Default", 0.0 );
 9.      }
10.
11.      public Product( String name, double
price ) {
12.          this.name = name;
13.          this.price = price;
14.      }
15.
16.      public String getName() {
17.          return name;
18.      }
19.
20.      public void setName(String name) {
21.          this.name = name;
22.      }
23.
24.      public double getPrice() {
25.          return price;
26.      }
27.
28.      public void setPrice(double price) {
29.          this.price = price;
30.      }
31.  }
```

A servlet sets a session-scoped attribute product with an instance of com.example.product an forward to a JSP. Which two output the name of the product in the response? (Choose two)

A. <%= product.getName() %>
B. <jsp:useBean id="product" class="com.example.Product" />
<%= product.getName() %>
C. <jsp:useBean id="com.example.Product" scope="page">
<%= product.getName() %>
</jsp:useBean>
D. <jsp:useBean id="product" type="com.example.Product"
scope="page" />
<%= product.getName() %>
E. <jsp:useBean id="product" type="com.example.Product">
<%= product.getName() %>
</jsp:useBean>

Answer: B,D
Explanation:

QUESTION NO: 20

Click the Exhibit button.

Given the HTML form:

1. <html>

2. <body>

3. <form action="submit.jsp">

4. Name: <input type="text" name="i1">

5. Price: <input type="text" name="i2">

6. <input type="submit">

7. </form>

8. </body>

9. </html>

Assume the product attribute does NOT yet exist in any scope.

Which code snippet, in submit.jsp, instantiates an instance of com.example.Product that contains the results of the form submission?

```
1.  package com.example;
2.
3.  public class Product {
4.     private String name;
5.     private double price;
6.
7.     public Product() {
8.        this( "Default", 0.0 );
9.     }
10.
11.    public Product( String name, double
price ) {
12.       this.name = name;
13.       this.price = price;
14.    }
15.
16.    public String getName() {
17.       return name;
18.    }
19.
20.    public void setName(String name) {
21.       this.name = name;
22.    }
23.
24.    public double getPrice() {
25.       return price;
26.    }
27.
28.    public void setPrice(double price) {
29.       this.price = price;
30.    }
31. }
```

A. <jsp:useBean id="com.example.Product" />
<jsp:setProperty name="product" property="*" />
B. <jsp:useBean id="product" class="com.example.Product" />
${product.name = param.i1}
${product.price = param.i2}
C. <jsp:useBean id="product" class="com.example.Product">
<jsp:setProperty name="product" property="name"
param="i1" />
<jsp:setProperty name="product" property="price"
param="i2" />
</jsp:useBean>
D. <jsp:useBean id="product" type="com.example.Product">
<jsp:setProperty name="product" property="name"
value="<%= request.getParameter("i1") %>" />
<jsp:setProperty name="product" property="price"
value="<%= request.getParameter("i2") %>" />
</jsp:useBean>

Answer: C

Explanation:

QUESTION NO: 21

You need to create a JavaBean object that is used only within the current ISP page. It must NOT be accessible to any other page including those that this page might import.

Which JSP standard action < .in accomplish this goal?

A. <jsp:useBean id = 'pageBean' type = 'com.example.MyBean' />
B. <jsp:useBean id = 'pageBean' class = 'com.example.MyBean' />
C. <jsp:makeBean id = 'pageBean' type = 'com.example.MyBean' />
D. <jsp:makeBean id = 'pageBean' class = 'com.example.MyBean' />
E. <jsp:useBean id = 'pageBean' class = 'com.example.MyBean' />
F. <jsp:makeBean id = 'pageBean' class = 'com.example.MyBean' />

Answer: B

Explanation:

QUESTION NO: 22

Given the JavaBean class:

public class MyBean {

private Boolean roasted = false;

public MyBean () {}

public Boolean isRoasted () { returned roasted }

public void setRoasted (Boolean roasted) { this.roasted = roasted;]

}

Assume a controller has created an instance of this bean, called setRoasted (true), and inserted the bean into the application scope using the name "myBean".

A JSP contains these two tags:

<jsp: useBean id = "aBean" class = "MyBean" scope = "page" \ />

<jsp: getProprty name = "aBean" property = "roasted" \ />

Which is true?

A. the page will include the output false
B. the page will include the output
C. the page will report that the property roasted does not exist
D. the page will report that the syntax of the useBean tag is incorrect
E. the page will report that the syntax of the getProperty tag Is incorrect

Answer: E
Reference: http://www.emacao.gov.mo/documents/18/06/exam.pdf (4th table from the top)

QUESTION NO: 23

A developer chooses to avoid using singieThreadModel but wants to ensure that data If; updated in a thread safe manner.

Which two can support this design goal? (Choose two)

A. Store the data in a local variable.
B. Store the data in an instance variable.
C. Store the data in the Httpsession object.
D. Store the data in the sarvletContext object.
E. Store the data in the ServletRequest object.

Answer: A,E
Explanation:

QUESTION NO: 24

You have a use in your web application that adds several session-scoped attributes. At the end of the use case, one of these objects, the manage attribute, is removed and then it needs to decide which of the other session-scoped attributes to remove.

How can this goal be accomplished?

A. The object of the manager attribute should implement the HttpSessionBindingListener and it should call the removeAttribute method on the appropriate session attributes.
B. The object of the manager attribute should implement the HttpSessionListenar and it should call the removeAttribute method on the appropriate session attributes.
C. The object of the manager attribute should implement the HttpSessionBindingListener rind it should call the deleteAttribute method on the appropriate session attributes.
D. The object of the manager attribute should implement the HttpsessionListener and it should call the deleteAttribute method on the appropriate session attributes.

Answer: A
Explanation:

QUESTION NO: 25

The Squeaky Beans Inc. shopping application was initially developed for a non-distributed environment. The company recently purchased the Acme Application Server, which supports distributed Httpsession objects. When deploying the application to the server, the doployer marks it as distributable in the web application deployment descriptor to take advantage of this feature.

Given this scenario, which two must be true? (Choose two)

A. The J2EE web container must support migration of objects that implement serializable.
B. The J2EE web container must use the native JVM Serialization mechanism for distributing HttpSesaion objects.
C. As per the specification, the J2EE web container ensures that distributed Httpsassion objects will be stored in a database.
D. Storing references to Enterprise JavaBeans components in the HttpSession object might NOT supported by J2EE web contain.

Answer: A,D
Explanation:

QUESTION NO: 26

Given an HttpServletRequest request and an httpServletResponse response:

41. HttpSession session = null;

42. / / insert code here

43. if (session = = null) {

44. // do something if session does not exist

45. } else {

46. // do something if session exists

47. }

To implement the design intent, which statement must be inserted at line 42?

A. session = response.getSession ();
B. session = request.getSession ();
C. session = request.getSession (true);
D. session = request.getSession (false);
E. session = request.getSession ("jsessionid");

Answer: D
Explanation:

QUESTION NO: 27

Users of your web application have requested that they should be able to set the duration of their sessions. So for example, one user might want a webapp to stay connected for an hour rather than the webapp's default of fifteen minutes; another user might want to stay connected for a whole day.

Furthermore, you have a special login servlet that performs user authentication and retrieves the object from the database. You want to augment this code to set up the user's specified session duration.

Which code snippet in the login servlet will accomplish this goal?

A. User user = // retrieves the user objects object from the database
session.setDurationInterval (user.getSessionDuration());
B. User user = // retrieves the User objects from the database
session.setInactiveInterval (user.getSessionDuration());
C. User user = // retrieves the user objects object from the database
session.setInactiveInterval (user.get.SessionDuration());
D. User user = // retrieves the user objects object from the database
session.setDuration (user.getSessionDuration());
E. User user = // retrieves the user objects object from the database
session.setMaxInactiveInterval (user.getSessionDuration());
F. User user = // retrieves the user objects object from the database

session.setMaxDurationInterval (user.getSessionDuration());

Answer: E
Explanation:

QUESTION NO: 28

Which interface must a class implement so that instances of the class are notified after any object added to a session?

A. javax.servlet.http.HttpSessionListener
B. javax.servlet.http.HttpSessionValueListener
C. javax.servlet.http.HttpSessionBindingListener
D. javax.servlet.http.HttpSessionAttributeListener

Answer: D
Explanation:

QUESTION NO: 29

Which statement is true about web container session management?

A. Access to session-scoped attributes is guaranteed to be thread-safe by the web container.
B. To activate URL rewriting, the developer must use the HttpServletResponse.setURLRewriting method.
C. If the web application uses HTTPS, then the web container may use the data on the HTTPS request stream to identify the client.
D. The JSESSIONID cookie is stored permanently on the client so that a user may return to the web application and the web container will rejoin that session.

Answer: C
Explanation:

QUESTION NO: 30

A method call inside your servlet has thrown an IOException resulting from a lost network connection to a necessary resource. The servlet has determined that it will be unable to operate successfully fen few minutes, but expects to be able to resume operation after that period. How should this situation handled?

A. The servlet should re-throw the exception to the web container.
B. The servlet should throw a servletException to the web container.
C. The servlet should throw an unavailableException to the web container.
D. The servlet should redirect to an error page that reports the exception message.
E. The servlet should retry the request until the method returns successfully.

Answer: C
Reference: http://jcp.org/aboutJava/communityprocess/first/jsr053/servlet23_PFD.pdf (page 26, topic: 2.3.3.2, first paragraph)

QUESTION NO: 31

During initialization, a servlet finds that the database connection it requires is unavailable. As the system designer, you know that the database start up completes several minutes after the web-container initializes the servlet. How should this be handled?

A. Retry the connection until it is successful, then allow the init() method to complete.
B. Throw a ServletException
C. Throw the IOException
D. Throw an UnavailableException

Answer: D
Reference: http://java.sun.com/j2ee/tutorial/1_3-fcs/doc/Servlets6.html

QUESTION NO: 32

A servlet wishes to indicate that it is unable to initialize at the present time, but that the initialization might succeed at some future time. Which is true?

A. This cannot be expressed. A servlet either initializes correctly or fails.
B. This expression is not necessary. If a servlet fails to initialize, the container will try again later each time a request is received that attempts to invoke that servlet.
C. The servlet should delay until it is ready to complete initialization successfully.
D. The servlet should throw an UnavailableException
E. The servlet should throw a servletException

Answer: E
Explanation:

QUESTION NO: 33

You are implementing a model component. You realize that an IOException might arise if you lose connection to the database. How should you address this?

A. Implement multipathing to provide redundant connectivity to the database, thereby avoiding that risk of connection failure.
B. Provide an error handler page, and use the page directive in the invoking ISP to redirect to that page if the error arises.
C. Use the JSTL <c:catch> tag to take control if the exception arises.
D. Surround the problem area with a try/catch block and implement appropriate recovery or fallback behavior.

Answer: C
Reference: http://www.ibm.com/developerworks/java/library/j-jstl0318/ (topic: exception handling)

QUESTION NO: 34

Which is a valid value for the enum EmptyRoleSemantic in
Javax.servlet.annotation.ServletSecurity?

A. ADMIT
B. PERMIT
C. EXCLUDE
D. DENYALL
E. ALLOWALL

Answer: B
Reference: http://tomcat.apache.org/tomcat-7.0-
doc/servletapi/javax/servlet/annotation/ServletSecurity.EmptyRoleSemantic.html

QUESTION NO: 35

Given the annotation for MyServlet:

@WebServlet{"/abc")

@ServletSecurity (value=@HttpConstraint)

public class Myservlet extends Httpservlet {

Which two are true? (Choose two)

A. No protection of user data must be performed by the transport.
B. All user data must be encrypted by the transport.
C. Access is to be denied independent of authentication state and identity.
D. Only authenticated users are to be permitted.
E. Access is to be permitted independent of authentication state and identity.

Answer: B,C
Explanation:

QUESTION NO: 36

Which of the following annotations relate to security in a servlet?

(i) @WebSecurity

(ii) @RolesAllowed

(iii) @WebConstraint

(iv) @HttpConstraint

(v) @Servletsecurity

A. (i) only
B. (i) and (iii)
C. (iv) and (v)
D. (iii) and (v)
E. (ii) and (iv)

Answer: C
Explanation:

QUESTION NO: 37

Which element of a web application deployment descriptor <web-resource-collection> element is

used to specify a HTTP method to which the corresponding <security-constraint> will not apply?

A. <exclude-http-method>
B. <neglect-http-method>
C. <http-method-omission>
D. <http-method-excluded>
E. <exceptional-http-method>

Answer: C
Reference: http://java.sun.com/xml/ns/javaee/web-common_3_0.xsd (search http-method-omission)

QUESTION NO: 38

Which two statements are true about the security-related tags in a valid Java EE deployment descriptor? (Choose two)

A. Every <security-constraint> tag must have at least one <http-method> tag.
B. A <security-constraint> tag can have many <web-resource-collection> tags.
C. A given <auth-constraint> tag can apply to only one <web-resource-collection> tag.
D. A given <web-resource-collection> tag can contain from zero to many <url-pattern> tags.
E. It is possible to construct a valid <security-constraint> tag such that, for a given resource user roles can access that resource.

Answer: B,E
Explanation:

QUESTION NO: 39

A cookie may be set to be an HttpOnly cookie. Setting a cookie to be HttpOnly results in (Choose two)

A. Client to not expose the cookie to client side scripting code
B. Does not work with https protocol
C. prevent certain types of cross-site scripting attacks
D. There is no such thing as an HttpOnly cookie

Answer: A,C
Reference: http://docs.oracle.com/javaee/6/api/javax/servlet/http/Cookie.html

QUESTION NO: 40

A popular Ajax framework and its companion widget library contain several hundreds of files of different types (.js, .ess, .html). Your company has mandated that all its web applications use only specific versions of this framework approved by IT and repackaged internally as jar files. Furthermore, web applications should only include the entire jar, without subsetting or modification.

Where should the framework's files be placed inside a jar file to enable this kind of reuse?

A. under resources
B. under META-INF/resources
C. under META-INF/web-contents
D. under WEB-INF/resources

Answer: B
Reference: http://ocpsoft.com/opensource/create-common-facelets-jar/ (check the box with the update)

QUESTION NO: 41

Which two actions protect a resource file from direct HTTP access within a web application? (Choose two)

A. placing it in the /secure directory
B. placing it in the /WEB-INF directory
C. placing it in the /META-INF/secure directory
D. creating a <web-resource> element within the deployment descriptor
E. creating a <secure-resource> element within the deployment descriptor

Answer: B,C
Explanation:

QUESTION NO: 42

In which two locations can library dependencies be defined for a web application? (Choose two)

A. the web application deployment descriptor

B. the /META-INF/dependencies.xml file

C. the /META-INF/MANIFEST.MF manifest file

D. the /META-INF/MANIFEST.MF manifest of a JAR in the web application classpath

Answer: C,D

Explanation:

QUESTION NO: 43

The jquery_1_3_2.jar file contains the JQuery Ajax framework in its META-INF/ resources directory. Where should the jar file be placed inside the web application to ensure the resources it contains are accessible by clients?

A. WEB-INF/classes

B. WEB-INF/jar

C. WEB-INF/lib

D. WEB INF/resources

Answer: C

Reference:

http://www.ibm.com/developerworks/websphere/library/techarticles/0112_deboer/deboer.html (topic: web modules, third paragraph)

QUESTION NO: 44

Given a jar file packaged with three web fragments with names X, Y and Z respectively.

Which of the following deployment descriptor, web.xml, snippets correspond to the web fragment processing orders of X, Y, Z?

(i) <absolute-ordering>

<name>X</name>

<name>Y</name>

<name>Z</name>

</absolute-ordering>

(ii) <absolute-ordering>

<name>X</name>

<name>Y</name>

</absolute-ordering>

(iii) <absolute-ordering>

</others>

<name>Y</name>

<name>Z</name>

</absolute-ordering>

(iv) <absolute-ordering>

<name>X</name>

</others>

<name>Z</name>

</absolute-ordering>

(v) <absolute-ordering>

<name>X</name>

<name>Y</name>

</others>

</absolute-ordering>

A. (i) only
B. (i) and (ii)
C. (i) and (iv)
D. (i), (iii), (iv) and (v)

Answer: C
Reference: http://blogs.oracle.com/swchan/entry/servlet_3_0_web_fragment

QUESTION NO: 45

Which defines the welcome files in a web application deployment descriptor?

A. <welcome>
<welcome- file >/ welcome - jsp</welcome-file>
</welcome>
<welcome>
<welcome-file>/index-html</welcome-file>
</welcome>
B. <we1come-file-list>
<welcome-file> welcome .jsp</welcome-file>
<welcome-file>index.html</welcome-file>
</welcome-file-list>
C. <welcome>
<welcome-file>welcome.jsp</welcome </we1come >
</welcome >
<welcome-file>index.html<//welcome-file>
</we1come >
D. <we1come-file-list>
<welcome-file> welcome .jsp</welcome-file>
<welcome-file>/index.html</welcome-file>
</welcome-file-list>
E. <welcome>
<welcome-file>
<welcome-name> welcome </welcome-name>
<location> welcome.jsp </location>
</welcome-file>
<welcome-file>
<welcome-name>index</welcome-name>
<location> index </location>
</welcome-file>
</welcome>

Answer: B
Explanation:

QUESTION NO: 46

Which code snippet specifies the metadata for a servlet with:

(i) url mapping "/mytest"'

(ii) Initial parameter with name "debug" and value "10"'

(iii) Support asynchronous operation?

A. @WebServlet("/mytest")
@InitParam(name-"debug" va1ue = "10")
@Asynchronous
public class TestServlet extends HttpServlet {}
B. @WebServlet ("/myset")
@WebParam(name = "debug" value = "10")
Public class testServlet {}
C. @WebServlet (urlPatterns = { / "mytest"},
Initparams = {@InitParam (name = "debug", value = "10") }, supportAsync = true)
public class TestServlet extends HttpServlet {}
D. @WebServlet (urlPatterns = { / "mytest"},
Initparams = {@InitParam (name = "debug", value = "10") }, asyncSupported = true)
public class TestServlet extends HttpServlet {}
E. @WebServlet (urlPatterns = { / "mytest"},
Initparam = @WebInitParam (name = "debug", value = "10"), supportAsync = true)
public class testServlet {}

Answer: D
Explanation:

QUESTION NO: 47

Which annotation specifies the metadata as a javax.servlet.http.HttpSessionListener?

A. @Listener
B. @WebListener
C. @WebSessionListener
D. @HttpSessionListener

Answer: A
Reference: http://docs.oracle.com/cd/E12840_01/wls/docs103/webapp/annotateservlet.html (table 8-1, third row)

QUESTION NO: 48

You are creating a new ISP page and you need to execute some code that acts when the- page is first executed, but only once. Which three are possible mechanisms for performing this initialization code? (Choose three)

A. In the init method.
B. In the jspInit method.
C. In the constructor of the JSP's Java code.
D. In a JSP declaration, which includes an initializer block.
E. In a JSP declaration, which includes a static initializer block.

Answer: B,D,E
Explanation:

QUESTION NO: 49

You are creating a web form with this HTML:

11. <form action="sendOrder.jsp">

12. <input type="text" name="creditCard">

13. <input type="text" name="expirationDate">

14. <input type="submit">

15. </form>

Which HTTP method is used when sending this request from the browser?

A. GET
B. PUT
C. POST
D. SEND
E. FORM

Answer: A
Explanation:

QUESTION NO: 50

Your web application requires the ability to load and remove web files dynamically to the web container's file system? Which two HTTP methods are used to perform these actions? (Choose two)

A. PUT

B. POST
C. SEND
D. DELETE
E. REMOVE
F. DESTROY

Answer: A,D
Explanation:

QUESTION NO: 51

A web application wants to expose to its clients an operation that results in a transaction being performed across several systems, for example the transfer of money from one bank account to another

Which HTTP method should it use?

A. GET
B. HEAD
C. OPT IONS
D. POST

Answer: D
Reference: http://en.wikipedia.org/wiki/POST_(HTTP)

QUESTION NO: 52

A Client wants to retrieve a representation of a resource, e.g. an XML document, localized according to the preferences of the user. Each user is allowed to specify one or more languages, in order of preference. Such preferences are application independent. How can the Client send language preference Information to the Server using a standard HTTP facility?

A. In the request body
B. In a request header
C. In the request line
D. In the request URL

Answer: B
Reference: http://www.w3.org/International/questions/qa-lang-priorities.en.php (Check the answer,

first paragraph)

QUESTION NO: 53

Which JSTL code snippet can be used to import content from another web resource?

A. <c:import url = "foo.jsp"/>
B. <c:import page = "foo.jsp"/>
C. <c:include url = "foo.jsp"/>
D. <c:include page = "foo.jsp"/>
E. Importing cannot be done in JSTL. A standard action must be used instead.

Answer: A
Explanation:

QUESTION NO: 54

In a jsp-centric shopping cart application cart application to move a client's home address of Customer object into the shipping address of the order object. The address data is stored in a value object class called Address with properties for: street address, city, province, country, and postal code.

Which two code snippets can be used to accomplish this goal?

A. <c:set var='order' property='shipAddress'
value='${client.homeAddress}' />
B. <c:set target='${order}' property='shipAddress'
value='${client.homeAddress}' />
C. <jsp:setProperty name='${order}' property='shipAddress'
value='${client.homeAddress}' />
D. <c:set var='order' property='shipAddress'>
<jsp:getProperty name='client' property='homeAddress' />
</c:store>
E. <c:set target='${order}' property='shipAddress'>
<jsp:getProperty name='client' property='homeAddress' />
</c:set>
F. <c:setProperty name='${order}' property='shipAddress'>
<jsp:getProperty name='client' property='homeAddress' />
</c:setProperty>

Answer: B,E
Explanation:

QUESTION NO: 55

You are creating a JSP page to display a collection of data. This data can be displayed in several different ways so the architect on your project decided to create a generic servlet that generates a comma delimited string so that various pages can render the data in different ways. This servlet takes on request parameter: objoctID. Assume that this servlet is mapped to the URL pattern: /WEB-INF/data.

In the JSP you are creating, you need to split this string into its elements separated by commas and generate an HTML <ul> list from the data.

Which JSTL code snippet will accomplish this goal?

A. <c:import varReader='dataString' url='/WEB-INF/data'>
<c:param name='objectID' value='${currentOID}' />
</c:import>

<c:forTokens items'${dataString.split(",")}' var='item'>
${item}
</c:forTokens>

B. <c:import varReader='dataString' url='/WEB-INF/data'>
<c:param name='objectID' value='${currentOID}' />
</c:import>

<c:forTokens items'${dataString}' delims=',' var='item'>
${item}
</c:forTokens>

C. <c:import var='dataString' url='/WEB-INF/data'>
<c:param name='objectID' value='${currentOID}' />
</c:import>

<c:forTokens items'${dataString.split(",")}' var='item'>
${item}
</c:forTokens>

D. <c:import var='dataString' url='/WEB-INF/data'>
<c:param name='objectID' value='${currentOID}' />
</c:import>

```
<ul>
<c:forTokens items'${dataString}' delims=',' var='item'>
<li>${item}</li>
</c:forTokens>
</ul>
```
                                                                                    32

Answer: D

Explanation:

QUESTION NO: 56

Which JSP standard action can be used to import content from a resource called foo.jsp?

A. <jsp:import file='foo.jsp' />
B. <jsp:import page='foo.jsp' />
C. <jsp:include page='foo.jsp' />
D. <jsp:include file='foo.jsp' />
E. <jsp:import>foo.jsp</jsp:import>
F. <jsp:include>foo.jsp</jsp:include>

Answer: C

Explanation:

QUESTION NO: 57

Refer to the Exhibit.

```
  1.  <?xml version="1.0" encoding="UTF-8" ?>
  2.
  3.  <taglib
xmlns="http://java.sun.com/xml/ns/j2ee"
  4.     xmlns:xsi="http://www.w3.org/2001/XMLSchem
a-instance"
  5.     xsi:schemaLocation="http://java.sun.com/xm
l/ns/j2ee web-jsptaglibrary_2_0.xsd"
  6.     version="2.0">
  7.     <tlib-version>1.0</tlib-version>
  8.     <short-name>forum</short-name>
  9.     <uri>http://example.com/tld/forum</uri>
 10.     <tag>
 11.        <name>message</name>
 12.
<tag-class>com.example.MessageTag</tag-class>
 13.
<body-content>scriptless</body-content>
 14.        <attribute>
 15.           <name>from</name>
 16.           <rtexprvalue>true</rtexprvalue>
 17.        </attribute>
 18.        <attribute>
 19.           <name>subject</name>
 20.           <required>false</required>
 21.           <rtexprvalue>true</rtexprvalue>
 22.        </attribute>
 23.     </tag>
 24.  </taglib>
```

Assuming the tag library in the exhibit is imported with the prefix forum, which custom tag invocation procedures a transaction error in a jsp page?

A. <forum:message from="My Name" subject="My Subject" />
B. <forum:message subject="My Subject">
My message body.
</forum:message>
C. <forum:message from="My Name" subject="${param.subject}">
${param.body}
</forum:message>
D. <forum:message from="My Name" subject="My Subject">
<%= request.getParameter("body") %>
</forum:message>
E. <forum:message from="My Name"
subject="<%= request.getParameter("subject") %>">
My message body.
</forum:message>

Answer: D
Explanation:
New Questions

QUESTION NO: 58

Which is a benefit of precompiling a JSP page?

A. It avoids initialization on the first request.
B. It provides the ability to debug runtime errors in the application.
C. It provides better performance on the first request for the JSP page.
D. It avoids execution of the _jspService method on the first request.

Answer: C
Explanation:

QUESTION NO: 59

A web browser need NOT always perform a complete request for a particular page that it suspects might NOT have changed. The HTTP specification provides a mechanism for the browser to retrieve only a partial response from the web server; this response includes information, such as the Last-Modified date but NOT the body of the page.

Which HTTP method will the browser use to retrieve such a partial response?

A. GET
B. ASK
C. SEND
D. HEAD
E. TRACE
F. OPTIONS

Answer: D
Explanation:

QUESTION NO: 60

A Client wants to retrieve a representation of a resource, for example an HTML page, allowing a cached version to be returning by an intermediary such a proxy server.

Which HTTP method should it use?

A. GET
B. PUT
C. POST
D. DELETE

Answer: A
Explanation: The first version of the HTTP protocol had only one method, namely GET, which would request a page from a server. The response from the server was always an HTML page

QUESTION NO: 61

Which two are true about the JSTL core iteration custom tags? (Choose two.)

A. It may iterate over arrays, collections, maps, and strings.
B. The body of the tag may contain EL code, but not scripting code.
C. When looping over collections, a loop status object may be used in the tag body.
D. It may iterate over map, but only the key of the mapping may be used in the tag body.
E. When looping over integers (for example begin= '1' end= '10'), a loop status object may not be used in the tag body.

Answer: A,C
Explanation:

QUESTION NO: 62

Assume a JavaBean com.example.GradedTestBean exists and has two attributes. The attribute name of type java.lang.string and the attribute score is of type jave.lang.Integer.

An array of com.example.GradedTestBean objects is exposed to the page in a request-scoped attribute called results. Additionally, an empty java.util.HashMap called resultMap is placed in the page scope.

A JSP page needs to add the first entry in results to resultMap, storing the name attribute of the

bean as the key and the score attribute of the bean as the value.

Which code snippet of JSTL code satisfies this requirement?

A. ${resultMap[results[0].name] = results[0].score}
B. <c:set var="${resultMap}" key="{results[0].name}"
value="${results[0].score}" />
C. <c:set var="resultMap" property="${results[0].name}">
${results[0].value}
</c:set>
D. <c:set var="resultMap" property="${results[0].name}">
value="${results[0].score}" />
E. <c:set target="${resultMap}" property="${results[0].name}">
value="${results[0].score}" />

Answer: E
Explanation:

QUESTION NO: 63

For manageability purposes, you have been told to add a "count" instance variable to a critical JSP
Document so that a JMX MBean can track how frequent this JSP is being invoked.

Which JSP code snippet must you use to declare this instance variable in the JSP Document?

A. <jsp:declaration>
int count = 0;
<jsp:declaration>
B. <%! int count = 0; %>
C. <jsp:declaration.instance>
int count = 0;
<jsp:declaration.instance>
D. <jsp:scriptlet.declaration>
int count = 0;
<jsp:scriptlet.declaration>

Answer: A
Explanation:

QUESTION NO: 64

To take advantage of the capabilities of modern browsers that use web standards, such as XHTML and CSS, your web application is being converted from simple JSP pages to JSP Document format. However, one of your JSPs, /scripts/screenFunctions.jsp, generates a JavaScript file. This file is included in several web forms to create screen-specific validation functions and are included in these pages with the following statement:

```
10. <head>
11.    <script> src='/scripts/screenFunctions.jsp'
12.       language='javascript'
13.       type='application/javascript'> </script>
14. </head>
15. <!—body of the web form -->
```

Which JSP code snippet declares that this JSP Document is a JavaScript file?

A. <%@ page contentType='application/javascript' %>
B. <jsp:page contentType='application/javascript' />
C. <jsp:document contentType='application/javascript' />
D. <jsp:directive.page contentType='application/javascript' />
E. No declaration is needed because the web form XHTML page already declares the MIME type of the /scripts/screenFunctions.jsp file in the <script> tag.

Answer: D
Explanation:

QUESTION NO: 65

You are building your own layout mechanism by including dynamic content for the page's header and footer sections. The footer is always static, but the header generates the <title> tag that requests the page name to be specified dynamically when the header is imported.

Which JSP code snippet performs the import of the header content?

A. <jsp:include page='/WEB-INF/jsp/header.jsp'>
<jsp:param name='pageName' value='Welcome Page' />
</jsp:include>
B. <jsp:import page='/WEB-INF/jsp/header.jsp'>
<jsp:param name='pageName' value='Welcome Page' />
</jsp:import>
C. <jsp:include page='/WEB-INF/jsp/header.jsp'>

<jsp:attribute name='pageName' value='Welcome Page' />
</jsp:include>
D. <jsp:import page='/WEB-INF/jsp/header.jsp'>
<jsp:attribute name='pageName' value='Welcome Page' />
</jsp:import>

Answer: A
Explanation:

QUESTION NO: 66

View the exhibit.

```
1. <?xml version="1.0" encoding="UTF-8" ?>
2.
3. <taglib
xmlns="http://java.sun.com/xml/ns/j2ee"
4.    xmlns:xsi="http://www.w3.org/2001/XMLSchem
a-instance"
5.    xsi:schemaLocation="http://java.sun.com/xm
l/ns/j2ee web-jsptaglibrary_2_0.xsd"
6.    version="2.0">
7.    <tlib-version>1.0</tlib-version>
8.    <short-name>ex</short-name>
9.
<uri>http://example.com/tld/example</uri>
10.    <tag>
11.      <name>hello</name>
12.
<tag-class>com.example.HelloTag</tag-class>
13.
<body-content>scriptless</body-content>
14.    </tag>
15. </taglib>
```

Assume the tag library in the exhibit is placed in a web application in the path /WEB-
INF/tld/example.tld.

1.

2. <ex:hello />

Which JSP code, inserted at line 1, completes the JSP code to invoke the hello tag?

A. <%@ taglib prefix="ex" uri="/WEB-INF/tld" %>
B. <%@ taglib uri="/WEB-INF/tld/example.tld" %>
C. <%@ taglib prefix="ex"
uri:="http://localhost:8080/tld/example.tld" %>
D. <%@ taglib prefix="ex"
uri:="http://example.com/tld/example" %>

Answer: D
Explanation:

QUESTION NO: 67

You are working on a JSP that is intended to inform users about critical errors in the system. The JSP code is attempting to access the exception that represents the cause of the problem, but your IDE is telling you that the variable does not exist. What should you do to address this problem?

A. Add a page directive stating that this page is an error handler
B. Add scriptlet code to create a variable that refer to the exception
C. Add a <jsp:useBean tag to declare the and access the exception>
D. Perform the error handling in a servlet rather than in the JSP
E. Edit the page that caused the error to ensure that it specifies this page as its error handler

Answer: A
Explanation: Exception is a JSP implicit variable

The exception variable contains any Exception thrown on the previous JSP page with an errorPage directive that forwards to a page with an isErrorPage directive.

Example:
If you had a JSP (index.jsp) which throws an exception (I have deliberately thrown a NumberFormatException by parsing a String, obviously you wouldn't write a page that does this, its just an example)

<%@ page errorPage="error.jsp" %>
<% Integer.parseInt("foo"); //throws an exception %>
This will forward to error.jsp,

If error.jsp was

<%@ page isErrorPage = "true"%>
<body>

<h2>Your application has generated an error</h2>
<h3>Please check for the error given below</h3>
Exception:
<%= exception.toString() %>
</body>

Because it has the

<%@ page isErrorPage = "true"%>
page directive, the implicit variable exception will contain the Exception thrown in the previous jsp

So when you request index.jsp, the Exception will be thrown, and forwarded to error.jsp which will output html like this

<body>
<h2>Your application has generated an error</h2>
<h3>Please check for the error given below</h3>
Exception:
java.lang.NumberFormatException: For input string: "foo"
</body>

As @JB Nizet mentions exception is an instanceof Throwable calling exception.getMessage() For input string: "foo" instead of java.lang.NumberFormatException: For input string: "foo"

QUESTION NO: 68

All of your JSPs need to have a link that permits users to email the web master. This web application is licensed to many small businesses, each of which have a different email address for the web master. You have decided to use a context parameter that you specify in the deployment descriptor, like this:

42. <context-param>

43. <param-name>webmasterEmail</param-name>

44. <param-value>master@example.com</param-value>

45. </context-param>

Which JSP code snippet creates this email link?

A. contact us
B. contact us
C. contact us
D. contact us

Answer: D
Explanation:

QUESTION NO: 69

Which EL expression evaluates to the request URI?

A. ${requestURI}
B. ${request.URI}
C. ${request.getURI}
D. ${request.requestURI}
E. ${requestScope.requestURI}
F. ${pageContext.request.requestURI}
G. ${requestScope.request.requestDRI}

Answer: F
Explanation:

QUESTION NO: 70

You are building a dating web site. The client's date of birth is collected along with lots of other information. The Person class has a derived method, getAge() :int, which returns the person's age calculated from the date of birth and today's date. In one of your JSPs you need to print a special message to clients within the age group of 25 through 35.

Which two EL code snippets will return true for this condition? (Choose two.)

A. ${client.age in [25,35]}
B. ${client.age between [25,35]}
C. ${client.age between 25,35}
D. ${client.age <= 35 && client.age >= 25}
E. ${client.age le 35 and client.age ge 25}
F. ${client.age > 35 && client.age < 25}

Answer: D,E

Explanation:

QUESTION NO: 71

Given an EL function declared with:

```
11. <function>
12.    <name>spin</name>
13.    <function-class>com.example.Spinner</function-class>
14.    <function-signature>
15.       java.lang.String spinlt ()
16.    </ function-signature>
17. </function>
```

Which two are true? (Choose two)

A. The function method must have the signature:
public String spin().
B. The method must be mapped to the logical name "spin" in the web.xml file.
C. The function method must have the signature:
public String spinIt().
D. The function method must have the signature:
public static String spin().
E. The function method must have the signature:
public static String spinIt().
F. The function class must be named Spinner, and must be in the package com.example.

Answer: E,F
Explanation:

QUESTION NO: 72

Which EL expression returns true if no session has been established with current client?

A. ${not(pageContext.session)}
B. ${not(requestScope.session)}
C. ${requestScope.sessions.this}
D. ${sessionScope.empty}

Answer: A

Explanation:

Note:

* A session is never null. The session is always present in JSP EL, unless you add

<%@page session="false" %>

* If you'd like to check if the session is new or has already been created, use HttpSession#isNew() instead.

<c:if test="${not pageContext.session['new']}">

<p>You've already visited this site before.</p>

</c:if>

<c:if test="${pageContext.session['new']}">

<p>You've just started the session with this request!</p>

</c:if>

QUESTION NO: 73

Your web application uses a simple architecture in which servlets handle requests and then forward to a JSP using a request dispatcher. You need to pass information calculated in the servlet to the JSP for view generation. This information must NOT be accessible to any other servlet, JSP or session in the webapp. Which two techniques can you use to accomplish this goal? (Choose two)

A. Add attributes to the session object.
B. Add attributes on the request object.
C. Add parameters on the request object.
D. Use the pageContext object to add request attributes.
E. Add parameters to the JSP's URL when generating the request dispatcher.

Answer: B,E
Explanation:

QUESTION NO: 74

A web application for business expense reporting allows uploading expense receipts. Multiple receipts can be uploaded single step using one HTTP request. The servlet that processes the

request has been marked with the @MultipartConfig annotation.

Which method should the servlet use to access the uploaded files?

A. HttpServletRequest.getParts()
B. HttpServletRequest.getData()
C. servletRequest.getParts()
D. servletRequest.getAllParts()

Answer: A
Explanation: The request.getParts() method returns collections of all Part objects. If you have more than one input of type file, multiple Part objects are returned. Since Part objects are named, the getPart(String name) method can be used to access a particular Part. Alternatively, the getParts() method, which returns an Iterable<Part>, can be used to get an Iterator over all the Part objects.

QUESTION NO: 75

For which three events can web application event listeners be registered? (Choose three.)

A. when a session is created
B. after a servlet is destroyed
C. when a session has timed out
D. when a cookie has been created
E. when a servlet has forwarded a request
F. when a session attribute value is changed

Answer: A,C,F
Explanation:

QUESTION NO: 76

To add a servlet to a context that has not been declared either via annotation or via the descriptor, during context initialization time the following API can be used (Choose three.)

A. servletContext.addServlet("myServletName", "MyServlet");
B. servletContext.addServlet((<Class extends
Servlet>)getClass().getClassLoader().getClassLoader().loaderClass("MyServlet"));

C. servletContext.addServlet(myServlet);
D. servletContext.addServlet(("myServletName", <Class extends
Servlet>)gerClass().getClassLoader().loaderClass("myServlet"));
E. servletContext.addServlet("myServletName", MyServlet);

Answer: A,D,E
Explanation: addServlet

ServletRegistration.Dynamic addServlet(java.lang.String servletName,
java.lang.Class<? extends Servlet> servletClass)
Adds the servlet with the given name and class type to this servlet context.
The registered servlet may be further configured via the returned ServletRegistration object.

Parameters:
servletName - the name of the servlet
servletClass - the class object from which the servlet will be instantiated

QUESTION NO: 77

ServletContextListeners are invoked in

A. Random order
B. contextInitialized and contextDestroyed are invoked in the order in which they are declared in
the web.xml
C. contextInitialized method are invoked in the order in which they are declared in the web.xml and
the contextDestroyed method is invoked in the reverse order in which they are declared in the
web.xml
D. contextInitialized and contextDestroyed are invoked in the reverse order of which they are
declared in the web.xml

Answer: C
Explanation:
public interface ServletContextListener
extends java.util.EventListener
Interface for receiving notification events about ServletContext lifecycle changes.

In order to receive these notification events, the implementation class must be either declared in
the deployment descriptor of the web application, annotated with WebListener, or registered via
one of the addListener methods defined on ServletContext.

Implementations of this interface are invoked at their contextInitialized(javax.servlet.ServletContextEvent) method in the order in which they have been declared, and at their contextDestroyed(javax.servlet.ServletContextEvent) method in reverse order.

QUESTION NO: 78

A servlet class is injected with a JDBC data source. After injection has occurred, the servlet needs to create a cache out of some of the data in the database, so as to improve responsiveness.

Which two methods can host the cache creation code? (Choose two)

A. Servlet.init()
B. Servlet.destroy()
C. A method annotated with @Init
D. A method annotated with @PostConstruct
E. A method annotated with @PreDestroy
F. A method annotated with @Resource

Answer: A,D
Explanation: A: Because the Servlet init() method is invoked when the servlet instance is loaded, it is the perfect location to carry out expensive operations that need only be performed during initialization. By definition, the init() method is thread-safe. The results of operations in the HttpServlet.init() method can be cached safely in servlet instance variables, which become read-only in the servlet service method.
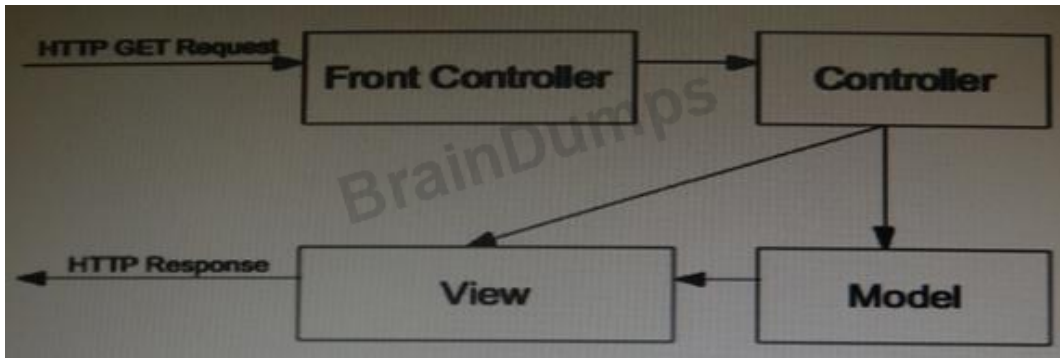
D: Example:
@PostConstruct
private void init() {
cached = (Cached) ctx.lookup(EJB_PATH + Cached.class.getSimpleName());

QUESTION NO: 79

View the Exhibit.

Which two technologies would be suitable for use as Front Controller elements? (Choose two)

A. JSP
B. Servlet
C. Filter
D. POJO
E. Custom Tag

Answer: A,B

Explanation: * Servlet only. Works well when:

– Output is a binary type. E.g.: an image

– There is no output. E.g.: you are doing forwarding or redirection as

in Search Engine example.

– Format/layout of page is highly variable. E.g.: portal.

* JSP only. Works well when:

– Output is mostly character data. E.g.: HTML

– Format/layout mostly fixed.

* Combination (MVC architecture). Needed when:

– A single request will result in multiple substantially differentlooking results.

– You have a large development team with different team members

doing the Web development and the business logic.

– You perform complicated data processing, but have a relatively

fixed layout

Incorrect:

Not D: In computing software, POJO is an acronym for Plain Old Java Object. The name is used

to emphasize that a given object is an ordinary Java Object, not a special object.

The term "POJO" is mainly used to denote a Java object which does not follow any of the major

Java object models, conventions, or frameworks. The term continues the pattern of older terms for

technologies that do not use fancy new features.

QUESTION NO: 80

View the Exhibit.

```
// Source Servlet : Source.java
10.   public class Source extends HttpServlet {
11.      public void service(HttpServletRequest request,
12.                          HttpServletResponse response)
13.               throws ServletException, IOException {
14.       RequestDispatcher rd =
15.         request.getRequestDispatcher("/dest/Destination");
16.       rd.forward(request, response);
17.      }
18.   }

// Filter : ParamAdder.java
12.   public class ParamAdder implements Filter {
        // ...
23.      public void doFilter(ServletRequest request,
24.                          ServletResponse response,
25.                          FilterChain chain)
26.          throws ServletException, IOException {
27.       request.setAttribute("filterAdded", "addedByFilter");
28.       chain.doFilter(request, response);
29.      }
        // ...
50.   }

//   Destination Servlet Destination.java
10.   public class Destination extends HttpServlet {
11.      public void service(HttpServletRequest request,
12.                          HttpServletResponse response)
13.               throws ServletException, IOException {
14.       String filterParam =
15.         (String) request.getAttribute("filterAdded");
16.       response.getWriter().println("filterAdded = "
17.                          + filterParam);
18.      }
19.   }
```

Given the web application deployment descriptor elements:

11. <filter>

12. <filter-name>ParamAdder</filter-name>

13. <filter-class>com.example.ParamAdder</filter-class>

14. </filter>

...

31. <filter-mapping>

32. <filter-name>ParamAdder</filter-name>

33. <servlet-name>Destination</servlet-name>

34. </filter-mapping>

...

55. <servlet-mapping>

56. <servlet-name>Destination</servlet-name>

57. <url-pattern>/dest/Destination</url-pattern>

58. </servlet-mapping>

What is the result of a client request of the Source servlet with no query string?

A. The output "filterAdded = null" is written to the response stream.
B. The output "filterAdded = addedByFilter" is written to the response stream.
C. An exception is thrown at runtime within the service method of the Source servlet.
D. An exception is thrown at runtime within the service method of the Destination servlet.

Answer: A
Explanation:

QUESTION NO: 81

Which three are true about the HttpServletRequestWrapper class? (Choose three.)

A. The HttpServletRequestWrapper is an example of the Decorator pattern.
B. The HttpServletRequestWrapper can be used to extend the functionality of a servlet request.
C. A subclass of HttpServletRequestWrapper CANNOT modify the behavior of the getReader method.
D. An HttpServletRequestWrapper may be used only by a class implement the javax.servlet.Filter interface.
E. An HttpServletRequestWrapper CANNOT be used on the request passed to the RequestDispatcher.interface.
F. An HttpServletRequestWrapper may modify the header of a request within an object implementing the javax.servlet.Filter interface.

Answer: A,B,F
Explanation:

QUESTION NO: 82

Given a war file with the following structure

| - WEB-INF/classes/Myservlet.class

| - WEB-INF/lib/wf.jaf

Where wf.jar contains a valid web-fragment.xml and the following two classes: MyFilter1.class and MyFiler2.class.

The web-fragment.xml is as follows:

```
<web-fragment metadata-complete="true">
    <filter>
        <filter-name>MyFilter2</filter-name>
        <filter-class>MyFilter2</filter-class>
    </filter>
    <filter-mapping>
        <filter-name>MyFilter2</filter-name>
        <url-pattern>/</url-pattern>
    </filter-mapping>
</web-fragment>
```

The following are some code snippets:

```
(i)   @WebServlet("/")
      public class MyServlet extends HttpServlet {}
(ii)  @WebFilter(urlPatterns={"/"})
      public class MyFilter1 implements Filter {}
(iii) @WebFilter(urlPatterns={"/abc"})
      public class MyFilter2 implements Filter {}
```

When one access "/" of the above web application, which filters will be executed?

A. No filters will be executed.
B. MyFilter1
C. MyFilter2
D. MyFilter1 and MyFilter2

Answer: C
Explanation:
Note:
* <filter-mapping>
This tag specifies a filter name, and either a URL mapping or servlet name, for a filter that has
been defined with the <filter> tag.

Multiple <filter-mapping> tags can be specified for a single <filter>, providing different URL patterns. See the <url-pattern> tag for examples.

The <filter-mapping> has two required elements:

<filter-name> - the filter name, as specified in the <filter-name> element of the <filter> tag
Either a <url-pattern> or a <servlet-name>.
If a servlet name is specified, the filter will be called whenever the specific servlet is called.

QUESTION NO: 83

Servlet A forwarded a request to servlet B using the forward method of RequestDispatcher. What attribute in B's request object contains the URI of the original request received by servlet A?

A. REQUEST_URI
B. javax.servlet.forward.request_uri
C. javax.servlet.forward.REQUEST_URI
D. javax.servlet.request_dispatcher.request_uri
E. javax.servlet.request_dispatcher.REQUEST_URI

Answer: B
Explanation:

QUESTION NO: 84

View the Exhibit.

```
1.  package com.example;
2.
3.  public class Product {
4.     private String name;
5.     private double price;
6.
7.     public Product() {
8.        this( "Default", 0.0 );
9.     }
10.
11.     public Product( String name, double price ) {
12.        this.name = name;
13.        this.price = price;
14.     }
15.
16.     public String getName() {
17.        return name;
18.     }
19.
20.     public void setName(String name) {
21.        this.name = name;
22.     }
23.
24.     public double getPrice() {
25.        return price;
26.     }
27.
28.     public void setPrice(double price) {
29.        this.price = price;
30.     }
31. }
```

Assume the product attribute does NOT yet exist in any scope.

Which two create an instance of com.example.Product and initialize the name and price properties to the name and price request parameters? (Choose two)

A. <jsp:useBean id="product" class="com.example.Product" />
<jsp:setProperty name="product" property="*" />
B. <jsp:useBean id="product" class="com.example.Product" />
<% product.setName(request.getParameter("name")); %>
<% product.setPrice(request.getParameter("price")); %>
C. <jsp:useBean id="product" class="com.example.Product" />
<jsp:setProperty name="product" property="name"
value="${param.name}" />
<jsp:setProperty name="product" property="price"
value="${param.price}" />
D. <jsp:useBean id="product" class="com.example.Product">
<jsp:setProperty name="product" property="name"
value="${name}" />
<jsp:setProperty name="product" property="price"
value="${price}" />
</jsp:useBean>

Answer: A,C
Explanation:

QUESTION NO: 85

A session-scoped attribute is stored by a servlet, and then servlet forwards to a JSP page. Which three jsp:useBean attributes must be used to access this attribute in the JSP page? (Choose three.)

A. id
B. name
C. bean
D. type
E. scope
F. beanName

Answer: A,D,E
Explanation:

QUESTION NO: 86

View the Exhibit.

```
 1.  package com.example;
 2.
 3.  public class Product {
 4.     private String name;
 5.     private double price;
 6.
 7.     public Product() {
 8.        this( "Default", 0.0 );
 9.     }
10.
11.     public Product( String name, double
price ) {
12.        this.name = name;
13.        this.price = price;
14.     }
15.
16.     public String getName() {
17.        return name;
18.     }
19.
20.     public void setName(String name) {
21.        this.name = name;
22.     }
23.
24.     public double getPrice() {
25.        return price;
26.     }
27.
28.     public void setPrice(double price) {
29.        this.price = price;
30.     }
31.  }
```

54

Given:

10. <form action='create_product.jsp'>

11. Product Name: <input type= 'text' name='productName'/>

12. Product Price: <input type= 'text' name='productPrice'/>

13. </form>

For a given product instance, which three jsp:setProperty attributes must be used to initialize its properties from the HTML form? (Choose three.)

A. id
B. name
C. type

D. param
E. property
F. reqParam
G. attribute

Answer: B,D,E
Explanation:

QUESTION NO: 87

A developer wants to make a name attribute available to all servlets associated with a particular user, across multiple requests from that user, from the same browser instance.

Which two provide this capability from within a tag handler? (Choose two)

A. pageContext.setAttribute("name", theValue);
B. pageContext.setAttribute("name", getSession());
C. pageContext.getRequest().setAttribute("name", theValue);
D. pageContext.getSession().setAttribute("name", theValue);
E. pageContext.setAttribute("name", theValue, PageContext.PAGE_SCOPE);
F. pageContext.setAttribute("name", theValue, PageContext.SESSION_SCOPE);

Answer: D,F
Explanation:

QUESTION NO: 88

Given that a web application consists of two HttpServlet classes, ServletA and ServletB, and the ServerletA.service method:

20. String key = "com.example.data";

21. session.setAttribute(key, "Hello");

22. object value = session.getAttribute(key);

23.

Assume session is an HttpSession, and is not referenced anywhere else in ServletA.

Which two changes, taken together, ensure that value is equal to "Hello" on line 23? (Choose two)

A. ensure that the ServletB.service method is synchronized

B. ensure that the ServletA.service method is synchronized

C. ensure that ServletB synchronizes on the session object when setting session attributes

D. enclose lines 21-22 in synchronized block:

synchronized(this) (

session.setAttribute(key, "Hello");

value = session.getAttribute(key);

)

E. enclose lines 21-22 in synchronized block:

synchronized(session) (

session.setAttribute(key, "Hello");

value = session.getAttribute(key);

)

Answer: C,E

Explanation:

QUESTION NO: 89

Given an HttpServletRequest request:

22. String id = request.getParameter("jsessionid");

23. // insert code here

24. String name = (String) session.getAttribute("name")

Which three can be placed at line 23 to retrieve an existing HttpSession object? (Choose three)

A. HttpSession session = request.getSession();

B. HttpSession session = request.getSession(id);

C. HttpSession session = request.getSession(true);

D. HttpSession session = request.getSession(false);

E. HttpSession session = request.getSession("jsessionid");

Answer: A,C,D

Explanation:

QUESTION NO: 90

As a convenience feature, your web pages include an Ajax request every five minutes to a special

servlet that monitors the age of the user's session. The client-side JavaScript that handles the Ajax callback displays a message on the screen as the session ages. The Ajax call does NOT pass any cookies, but it passes the session ID in a request parameter called sessionID. In addition, assume that your webapp keeps a hasmap of session objects by the ID. Here is a partial implementation of this servlet:

10. public class SessionAgeServlet extends HttpServlet {

11. public void service(HttpServletRequest request, HttpServletResponse) throws IOException {

12. String sessionID = request.getParameter("sessionID");

13. HttpSession session = getSession(sessionID);

14. long age = // your code here

15. response.getWriter().print(age);

16. }

... // more code here

47. }

Which code snippet on line 14, will determine the age of the session?

A. session.getMaxInactiveInterval();
B. session.getLastAccessed().getTime() – session.getCreationTime().getTime();
C. session.getLastAccessedTime().getTime() – session.getCreationTime().getTime();
D. session.getLastAccessed() - session.getCreationTime();
E. session.getMaxInactiveInterval() – session.getCreationTime();
F. session.getLastAccessedTime() – session.getCreationTime();

Answer: F
Explanation:

QUESTION NO: 91

Given the definition of Myobject and that an instance of Myobject is bound as a session attribute:

8. package com.example;

9. public class Myobject implements

10. javax.servlet.http.HttpSessionBindingListener {

11. // class body code here

12. }

Which is true?

A. Only a single instance of Myobject may exist within a session
B. The unbound method of the Myobject instance is called when the session to which it is bound times out
C. The com.example.MyObject must be declared as a servlet event listener in the web application deployment descriptor
D. The valueUnbound method of the Myobject instance is called when the session to which it is bound times out

Answer: D
Explanation:

QUESTION NO: 92

Your web application requires the adding and deleting of many session attributes during a complex use case. A bug report has come in that indicates that an important session attribute is being deleted too soon and a NullPointerException is being thrown several interactions after the fact. You have decided to create a session event listener that will log when attributes are being deleted so you can track down when the attribute is erroneously being deleted.

Which listener class will accomplish this debugging goal?

A. Create an HttpSessionAttributeListener class and implement the attributeDeleted method and log the attribute name using the getName method on the event object.
B. Create an HttpSessionAttributeListener class and implement the attributeRemoved method and log the attribute name using the getName method on the event object.
C. Create an SessionAttributeListener class and implement the attributeRemoved method and log the attribute name using the getAttributeName method on the event object.
D. Create an SessionAttributeListener class and implement the attributeDeleted method and log the attribute name using the getAttributeName method on the event object.

Answer: B
Explanation:

QUESTION NO: 93
Which method must be used to encode a URL passed as an argument to HttpServletResponse.

sendRedirect when using URL rewriting for session tracking?

A. ServletResponse.encodeURL
B. HttpServletResponse.encodeURL
C. ServletResponse.encodeRedirectURL
D. HttpServletResponse.encodeRedirectURL

Answer: D
Explanation:

QUESTION NO: 94

Which Java expression can be used to check whether the web container is currently configured to track sessions via URL rewriting?

A. servletContext.getSessionCookiesConfiq() .isHttpOnly()
B. servletContext.getSessionCookiesConfiq() .isSecure()
C. servletContext.getDefaultSessionTrackingModes() .contains(SessionTrackingMode.URL)
D. servletContext.getEffectiveSessionTrackingModes() .contains (SessionTrackingMode.URL)

Answer: D
Explanation: Code example:
String sessionID = null;
if (request.getServletContext().getEffectiveSessionTrackingModes()
contains(SessionTrackingMode.URL)) {

// Get the session ID if there was one
sessionID = request.getPathParameter(
SessionConfig.getSessionUriParamName(
request.getContext()));
if (sessionID != null) {
request.setRequestedSessionId(sessionID);
request.setRequestedSessionURL(true);
}
}

C: getDefaultSessionTrackingModes

java.util.Set<SessionTrackingMode> getDefaultSessionTrackingModes()
Gets the session tracking modes that are supported by default for this ServletContext.

QUESTION NO: 95

Which security mechanism uses the concept of a realm?

A. authorization
B. data integrity
C. confidentiality
D. authentication

Answer: D
Explanation:

QUESTION NO: 96

Which java code snippet roles "MANAGER" and "EMPLOYEE" in a given application?

@DeclareRole("EMPLOYEE")

public class MyServlet extends HttpServlet {}

A. @DeclareRoles({ "MANAGER", "EMPLOYEE" })public class MyServlet extends HttpServlet {}
B. @SecurityRoles({ "MANAGER", "EMPLOYEE" })public class MyServlet extends HttpServlet {}
C. @DeclareRoles("MANAGER")@DeclareRoles("EMPLOYEE")public class MyServlet extends HttpServlet {}
D. @DeclareRole("MANAGER")

Answer: A
Explanation: * The syntax for declaring more than one role is as shown in the following example:

@DeclareRoles({"Administrator", "Manager", "Employee"})

* @DeclareRoles
This annotation declares the security roles defined by the application.

* javax.annotation.security
Annotation Type DeclareRoles

@Documented
@Retention(value=RUNTIME)
@Target(value=TYPE)
public @interface DeclareRoles
Used by application to declare roles. It can be specified on a class.

QUESTION NO: 97

Given the java code snippet in contextInitialized method of a ServletContextListner:

ServletRegistration.Dynamic sr = (ServletRegistration.Dynamic)sc.addServlet

("myServlet", myServletClass);

sr.addMapping("/abc");

sr.setServletSecurityElement(servletSecurityElement);

sr.addMapping("/def");

Which statement is true?

A. "/abc" is mapped to "myservlet". The servletSecurityElementy applies to both "/abc" and "/def".
B. Both "/abc" and "/def" are mapped to "myservlet". The servletSecurityElementy applies to "/abc".
C. Both "/abc" and "/def" are mapped to "myservlet". The servletSecurityElementy applies to "/def".
D. Both "/abc" and "/def" are mapped to "myservlet". The servletSecurityElementy applies to both "/abc" and "/def".
E. Both "/abc" and "/def" are mapped to "myservlet". The servletSecurityElementy applies to "/abc", but the behavior for "/def" is not specified.

Answer: E
Explanation:
Note:
* addMapping

public void addMapping(java.lang.String namespace,
java.lang.String name,
java.lang.Class clazz)

Defines a direct mapping from a namespace and name to a java class (and vice versa)

QUESTION NO: 98

Given:

3. class MyServlet extends HttpServlet {

4. public void doPut(HttpServletRequest req,

  HttpServletResponse resp)

throws ServletException, IOException {

5. // servlet code here

…

26. }

27. }

If the DD contains a single security constraint associated with MyServlet and its only <http-method> tags and <auth-constraint> tags are:

<http-method>GET</http-method>

<http-method>PUT</http-method>

<auth-constrain>Admin</auth-constrain>

Which four requests would be allowed by the container? (Choose four)

A. A user whose role is Admin can perform a PUT.
B. A user whose role is Admin can perform a GET.
C. A user whose role is Admin can perform a POST.
D. A user whose role is Member can perform a PUT.
E. A user whose role is Member can perform a POST.
F. A user whose role is Member can perform a GET.

Answer: A,B,C,E
Explanation:

QUESTION NO: 99

Given this fragment in a servlet:

23. if(reg.isUserInRole("Admin")) {

24. // do stuff

25. }

And the following fragment from the related Java EE deployment descriptor:

812. <security-role-ref>

813. <role-name>Admin</role-name>

814. <role-link>Administrator</role-link>

815. </security-role-ref>

900. <security-role>

901. <role-name>Admin</role-name>

902. <role-name>Administrator</role-name>

903. </security-role>

What is the result?

A. Line 24 can never be reached.
B. The deployment descriptor is NOT valid.
C. If line 24 executes, the user's role will be Admin.
D. If line 24 executes, the user's role will be Administrator.
E. If line 24 executes, the user's role will NOT be predictable.

Answer: D
Explanation:

QUESTION NO: 100

Which java code snippet checks whether the user is of the role "MANAGER" for a given HttpServRequest, httpServletRequest?

A. httpServletRequest.isUserInRole("MANAGER");
B. httpServletRequest.isCallerInRole("MANAGER");
C. httpServletRequest.isPrincipalInRole("MANAGER");
D. httpServletRequest.isAuthnticatedUserInRole("MANAGER");

Answer: A
Explanation: isUserInRole

public boolean isUserInRole(java.lang.String role)
Returns a boolean indicating whether the authenticated user is included in the specified logical "role". Roles and role membership can be defined using deployment descriptors. If the user has not been authenticated, the method returns false.
Parameters:
role - a String specifying the name of the role
Returns:
a boolean indicating whether the user making this request belongs to a given role; false if the user has not been authenticated

Incorrect:
Not B: isCallerInRole is depreciated.

QUESTION NO: 101

A web application uses a cookies to track a client as it navigates through the pages that constitutes the application. Which code snippet can be used by the web application to reduce the chance of a cross-site scripting attack by setting some property of the cookie before returning it to the client?

A. cookie.setHttpOnly(true)
B. cookie.setMaxAge(3600)
C. cookie.setPath("/")
D. cookie.setSecure(true)

Answer: A
Explanation: When HTTPOnly flag is assigned to a cookie, the browser will restrict the access to such Cookie from Java Script code hence the cookie would only be sent to the subsequent
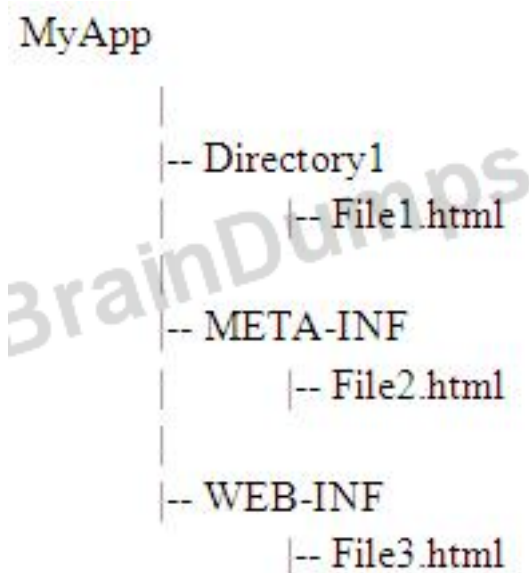
request to server but cannot be accessed using client side script. In such a case even if website is vulnerable to Cross Site Scripting (XSS) attacks, still the browser would safeguard the data stored into cookies flagged as HTTPOnly.

Incorrect:

Not D: If Secure flag is set for Cookie then it may only be transmitted over secure channel (SSL/HTTPS) ensuring that data is always encrypted while transmitting from client to server.

QUESTION NO: 102

Given the portion of a valid Java EE web application's directory structure:

```
MyApp
    |
    |-- Directory1
    |       |-- File1.html
    |
    |-- META-INF
    |       |-- File2.html
    |
    |-- WEB-INF
            |-- File3.html
```

You want to know whether File1.html, File2.html, and/or File3.html is protected from direct access by your web client's browsers. What statement is true?

A. All three files are directly accessible.
B. Only File1.html is directly accessible.
C. Only File2.html is directly accessible.
D. Only File3.html is directly accessible.
E. Only File1.html and File2.html are directly accessible.
F. Only File1.html and File3.html are directly accessible.
G. Only File2.html and File3.html are directly accessible.

Answer: B

Explanation:

Note:

* WEB-INF is the folder just under the root of a WAR that holds information that you don't want to be accessible to a client via a URL request. Specifically, it holds the web.xml, classes, and lib directories, but you can put anything you want to hide from the client there.

* META-INF is what discriminates a JAR file from a plain ZIP file. It holds the manifest file and may hold other deployment information as needed.

QUESTION NO: 103

Given a web fragment jar file, where should we put the web fragment.xml inside the jar file?

A. WEB-INF
B. META-INF
C. WEB-INF/lib
D. WEB-INF/classes
E. META-INF/services

Answer: C

Explanation: * If you're dealing with web applications, /WEB-INF/lib is the portable place to put JARs. This is where web servers servlet containers expect to find an application's jar files.

* The /WEB-INF/classes directory is on the ClassLoader's classpath. This is where .class files are loaded from when the web application is executing. Any JAR files placed in the /WEB-INF/lib directory will also be placed on the ClassLoader's classpath.

QUESTION NO: 104

Given the fragment from Java EE deployment descriptor:

341. <error-page>

342. <exception-type>java.lang.Throwable</exception-type>

---

343. <location>/mainError.jsp</location>

344. </error-page>

345. <error-page>

346. <exception-type>java.lang.ClassCastException</exception-type>

347. <location>/castError.jsp</location>

348. </error-page>

If the web application associated with the fragment above throws a ClassCastException.

Which statement is true?

A. The deployment descriptor is invalid.
B. The container invokes mainError.jsp
C. The container invokes castError.jsp
D. Neither mainError.jsp nor castError.jsp is invoked.

Answer: C
Explanation:

QUESTION NO: 105

You have built a web application that you license to small businesses. The webapp uses a context parameter, called licenseExtension, which enables certain advanced features based on your client's license package. When a client pays for a specific service, you provide them with a license extension key that they insert into the <context-param> of the deployment descriptor. Not every client will have this context parameter so you need to create a context listener to set up a default value in the licenseExtension parameter.

Which code snippet will accomplish this goal?

A. You cannot do this because context parameters CANNOT be altered programmatically.
B. String ext = context.getParameter('licenseExtension');if (ext == null) {
context.setParameter('licenseExtension' DEFAULT);}
C. String ext = context.getAttribute('licenseExtension');if (ext == null) {
context.setAttribute('licenseExtension' DEFAULT);}
D. String ext = context.getInitParameter('licenseExtension')if (ext == null) {
context.resetInitParameter('licenseExtension' DEFAULT);}
E. String ext = context.getInitParameter('licenseExtension')if (ext == null) {
context.setInitParameter('licenseExtension' DEFAULT);}

Answer: A
Explanation:

QUESTION NO: 106

Which of the following are attributes of the annotation javax.servlet.annotation.WebFiler?

- Name
- servletNames
- urlPatterns
- dispatcherTypes
- supportAsync

A. (iii) only
B. (iii) and (iv)
C. (ii), (iii) and (iv)
D. (iii), (iv) and (v)
E. (ii), (iii), (iv) and (v)

Answer: B
Explanation: * (iii) urlPatterns
The URL patterns to which the filter applies

* (iv) dispatcherTypes
The dispatcher types to which the filter applies

* (not V): asyncSupported
Declares whether the filter supports asynchronous operation mode.

Note:
* javax.servlet.annotation
Annotation Type WebFilter

Annotation used to declare a servlet filter.

This annotation is processed by the container at deployment time, and the corresponding filter applied to the specified URL patterns, servlets, and dispatcher types.

QUESTION NO: 107

Which of following annotations can be used in a servlet class?

- @javax.annotation.Resource
- @javax.annotation.PreDestroy
- @javax.annotation.security.RunAs
- @javax.annotation.security.RolesAllowed
- @javax.servlet.annotation.WebServlet

A. (v) only
B. (i) and (v)
C. (i), (ii), (iii) and (v)
D. (i), (ii), (iv) and (v)
E. (i), (ii), (iii), (iv) and (v)

Answer: B
Explanation: * (i) The javax.annotation.Resource annotation is used to declare a reference to a resource. @Resource can decorate a class, a field, or a method.

* (v) javax.servlet.annotation
Annotation Type WebServlet

Annotation used to declare a servlet.

This annotation is processed by the container at deployment time, and the corresponding servlet made available at the specified URL patterns.

Incorrect:
* Not (II) : javax.annotation.PreDestroy
Target: Method

QUESTION NO: 108

When using the @WebListener annotation, the class on which the annotation is applied to must also implement at least one o the following interfaces – (Choose two):

A. RequestListener

B. AttributeListener

C. ServletContextListener

D. HttpSessionListener

E. SessionAttributeListener

F. AsyncListener

Answer: C,D

Explanation: @WebListener

The main task of the listener is to listen the particular events and process your own task on that event. For example, if you want to initialize a database connection before your application starts, ServletContextListener will be implemented to do that. Another good example is -when you want to do some task on the creation and destruction of a session. For this purpose you need to implement HttpSessionListener.