1Z0-858 Q&As

Java Enterprise Edition 5 Web Component Developer Certified Professional Exam

QUESTION 1

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: objectID. 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

list from the data.

Which JSTL code snippet will accomplish this goal?

A.

- $ {item}

B.

- $ {item}

C.

- $ {item}

D.

- $ {item}

Correct Answer: D

QUESTION 2

Which is true about the web container request processing model?

A. The init method on a filter is called the first time a servlet mapped to that filter is invoked.

B. A filter defined for a servlet must always forward control to the next resource in the filter chain.

C. Filters associated with a named servlet are applied in the order they appear in the web application deployment descriptor file.

D. If the init method on a filter throws an UnavailableException, then the container will make no further attempt to execute it.

Correct Answer: C

QUESTION 3

Which two security mechanisms can be directed through a sub-element of the element in a web application deployment descriptor? (Choose two.)

A. authorization

B. data integrity

C. confidentiality

D. authentication

Correct Answer: BC

QUESTION 4

Squeaky Beans Inc. hired an outside consultant to develop their web application. To finish the job quickly, the consultant created several dozen JSP pages that directly communicate with the database. The Squeaky business team has since purchased a set of business objects to model their system, and the Squeaky developer charged with maintaining the web application must now refactor all the JSPs to work with the new system. Which pattern can the developer use to solve this problem?

A. Transfer Object

B. Service Locator

C. Intercepting Filter

D. Business Delegate

Correct Answer: D

QUESTION 5

Your web site has many user-customizable features, for example font and color preferences on web pages. Your IT department has already built a subsystem for user preferences using the Java SE platform\\'s lang.util.prefs package APIs, and you have been ordered to reuse this subsystem in your web application. You need to create an event listener that constructs the preferences factory and stores it in the application scope for later use. Furthermore, this factory requires that the URL to a database must be declared in the deployment descriptor like this:

42.

43.

prefsDbURL

44.

45.

jdbc:pointbase:server://dbhost:4747/prefsDB

46.

47.

Which partial listener class will accomplish this goal?

A. public class PrefsFactoryInitializer implements ContextListener {public void contextInitialized (ServletContextEvent e) {ServletContext ctx = e.getContext();String prefsURL = ctx.getParameter ("prefsDbURL");PreferencesFactory myFactory = makeFactory(prefsURL);ctx.putAttribute ("myPrefsFactory", myFactory);}// more code here}

B. public class PrefsFactoryInitializer implements ServletContextListener {public void contextCreated (ServletContext ctx) {String prefsURL = ctx.getInitParameter("prefsDbURL");PreferencesFactory myFactory = makeFactory (prefsURL);ctx.setAttribute("myPrefsFactory", myFactory);}// more code here}

C. public class PrefsFactoryInitializer implements ServletContextListener {public void contextInitialized (ServletContextEvent e) {ServletContext ctx = e.getServletContext();String prefsURL = ctx.getInitParameter ("prefsDbURL");PreferencesFactory myFactory = makeFactory (prefsURL);ctx.setAttribute("myPrefsFactory", myFactory);}// more code here}

D. public class PrefsFactoryInitializer implements ContextListener {public void contextCreated (ServletContext ctx) {String prefsURL = ctx.getParameter("prefsDbURL");PreferencesFactory myFactory = makeFactory (prefsURL);ctx.putAttribute("myPrefsFactory", myFactory);}// more code here}

Correct Answer: C