

**Oracle 1z0-859**

**Java Enterprise Edition 5 Web Component Developer  
Certified Professional Upgrade Exam**

**Practice Test**

Version: 14.22

**QUESTION NO: 1**

Given the relationship:



The tag handler MyTag extends SimpleTagSupport. At runtime, the doTag method throws a SkipPageException. Which three events occur after the SkipPageException is thrown?

(Choose three.)

- A. Evaluation of page2.jsp stops.
- B. Evaluation of page1.jsp stops.
- C. The MyTag instance is NOT reused.
- D. Evaluation of page2.jsp continues.
- E. Evaluation of page1.jsp continues.

**Answer: A,C,E**

**Explanation:**

**QUESTION NO: 2**

A developer is designing a multi-tier web application and discovers a need to log each incoming client request. Which two patterns, taken independently, provide a solution for this problem?

(Choose two.)

- A. Transfer Object
- B. Service Locator
- C. Front Controller
- D. Intercepting Filter
- E. Business Delegate
- F. Model-View-Controller

**Answer: C,D**

**Explanation:**

**QUESTION NO: 3**

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 implementing the `javax.servlet.Filter` interface.
- E. An `HttpServletRequestWrapper` CANNOT be used on the request passed to the `RequestDispatcher.include` method.
- 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: 4**

Which two are valid values for the `<transport-guarantee>` element inside a `<securityconstraint>` element of a web application deployment descriptor? (Choose two.)

- A. NULL
- B. SECURE
- C. INTEGRAL
- D. ENCRYPTED
- E. CONFIDENTIAL

**Answer: C,D**

**Explanation:**

**QUESTION NO: 5**

Given a web application in which the request parameter `productID` contains a product identifier. Which two EL expressions evaluate the value of the `productID`? (Choose two.)

- A. `${productID}`
- B. `${param.productID}`
- C. `${params.productID}`
- D. `${params.productID[1]}`

- E. `${paramValues.productId}`
- F. `${paramValues.productId[0]}`
- G. `${pageContext.request.productId}`

**Answer: B,F**

**Explanation:**

#### QUESTION NO: 6

Given the function invocation expression `${my:reverse("42")}`, and that the function reverse is mapped into a Java method called reverse, which two are valid signatures for the Java method reverse? (Choose two.)

- A. `public int reverse(String val)`
- B. `public String reverse(String val)`
- C. `public static int reverse(String val)` D. `public static String reverse(int val)`
- D. `private static double reverse(double val)`
- E. `public int reverse(String value, String name)`
- F. `public static int reverse(int value, String name)`

**Answer: C,D**

**Explanation:**

#### QUESTION NO: 7

For an `HttpServletResponse` response, which two create a custom header? (Choose two.)

- A. `response.setHeader("X-MyHeader", "34");`
- B. `response.addHeader("X-MyHeader", "34");`
- C. `response.setHeader(new HttpHeaders("X-MyHeader", "34"));`
- D. `response.addHeader(new HttpHeaders("X-MyHeader", "34"));`
- E. `response.addHeader(new ServletHeader("X-MyHeader", "34"));`
- F. `response.setHeader(new ServletHeader("X-MyHeader", "34"));`

**Answer: A,B**

**Explanation:**

#### QUESTION NO: 8

For a given `ServletResponse` response, which two retrieve an object for writing text data?

(Choose two.)

- A. `response.getWriter()`
- B. `response.getOutputStream()`
- C. `response.getWriter().getOutputStream()`
- D. `response.getWriter().getOutputStream()`
- E. `response.getWriter(Writer.OUTPUT_TEXT)`

**Answer: A,B**

**Explanation:**

### QUESTION NO: 9

Given an `HttpServletRequest` request and `HttpServletResponse` response, 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

Your company has a corporate policy that prohibits storing a customer's credit card number in any corporate database. However, users have complained that they do NOT want to re-enter their credit card number for each transaction. Your management has decided to use client-side cookies to record the user's credit card number for 120 days.

Furthermore, they also want to protect this information during transit from the web browser to the web container; so the cookie must only be transmitted over HTTPS. Which code snippet creates the "creditCard" cookie and adds it to the out going response to be stored on the user's web browser?

- A.** 10. `Cookie c = new Cookie("creditCard", usersCard);`  
11. `c.setSecure(true);`  
12. `c.setAge(10368000);`  
13. `response.addCookie(c);`
- B.** 10. `Cookie c = new Cookie("creditCard", usersCard);`  
11. `c.setHttps(true);`  
12. `c.setMaxAge(10368000);`  
13. `response.setCookie(c);`
- C.** 10. `Cookie c = new Cookie("creditCard", usersCard);`  
11. `c.setSecure(true);`  
12. `c.setMaxAge(10368000);`  
13. `response.addCookie(c);`
- D.** 10. `Cookie c = new Cookie("creditCard", usersCard);`  
11. `c.setHttps(true);`  
12. `c.setAge(10368000);`  
13. `response.addCookie(c);`
- E.** 10. `Cookie c = new Cookie("creditCard", usersCard);`  
11. `c.setSecure(true);`  
12. `c.setAge(10368000);`  
13. `response.setCookie(c);`

**Answer: C**

**Explanation:**

## QUESTION NO: 11 DRAG DROP

Click the Task button.

Order of Steps	Event
1st	web container instantiates the servlet
2nd	web container calls the servlet's <code>destroy()</code> method
3rd	web container loads the servlet class
4th	web container calls the servlet's <code>init()</code> method
5th	web container calls the servlet's <code>service()</code> method

Done

Place the events in the order they occur.

**Answer:**

Drag and Drop

Place the events in the order they occur.

Order of Steps	Event
web container loads the servlet class	web container instantiates the servlet
web container instantiates the servlet	web container calls the servlet's destroy() method
web container calls the servlet's init() method	web container loads the servlet class
web container calls the servlet's service() method	web container calls the servlet's init() method
web container calls the servlet's destroy() method	web container calls the servlet's service() method

Done

### QUESTION NO: 12

You are creating a servlet that generates stock market graphs. You want to provide the web browser with precise information about the amount of data being sent in the response stream. Which two HttpServletResponse methods will you use to provide this information?

(Choose two.)

- A. response.setLength(numberOfBytes);
- B. response.setContentLength(numberOfBytes);
- C. response.setHeader("Length", numberOfBytes);
- D. response.setIntHeader("Length", numberOfBytes);
- E. response.setHeader("Content-Length", numberOfBytes);
- F. response.setIntHeader("Content-Length", numberOfBytes);

**Answer: B,F**

**Explanation:**

### QUESTION NO: 13

Which two prevent a servlet from handling requests? (Choose two.)

- A. The servlet's init method returns a non-zero status.



- B. The servlet's init method throws a ServletException.
- C. The servlet's init method sets the ServletResponse's content length to 0.
- D. The servlet's init method sets the ServletResponse's content type to null.
- E. The servlet's init method does NOT return within a time period defined by the servlet container.

**Answer: B,E**

**Explanation:**

#### QUESTION NO: 14

Given an HttpSession session, a ServletRequest request, and a ServletContext context, which retrieves a URL to /WEB-INF/myconfig.xml within a web application?

- A. session.getResource("/WEB-INF/myconfig.xml")
- B. request.getResource("/WEB-INF/myconfig.xml")
- C. context.getResource("/WEB-INF/myconfig.xml")
- D. getClass().getResource("/WEB-INF/myconfig.xml")

**Answer: C**

**Explanation:**

#### QUESTION NO: 15

Given:

```
11. public void service(ServletRequest request,  
12.     ServletResponse response)  
13.     throws IOException {  
14.     ServletContext ctx = getServletConfig().getServletContext();  
15.     InputStream in =  
16.     // insert code here  
17. }
```

Which statement, at line 16, retrieves an InputStream for the file /WEB-INF/myresrc.bin?



- A. `new InputStream("/WEB-INF/myresrc.bin");`
- B. `ctx.getInputStream("/WEB-INF/myresrc.bin");`
- C. `ctx.getResourceAsStream("/WEB-INF/myresrc.bin");`
- D. `new InputStream(new URL("/WEB-INF/myresrc.bin"));`
- E. `getClass().getResourceAsStream("/WEB-INF/myresrc.bin");`

**Answer: C**

**Explanation:**

**QUESTION NO: 16**

Click the Exhibit button.

```
1. package com.example;
2.
3. import javax.servlet.http.*;
4.
5. public class MyWebDAV extends HttpServlet {
6.     private String resourceDirectory;
7.
8.     public MyWebDAV(String resDir) {
9.         this.resourceDirectory = resDir;
10.    }
11.    public void doPut(HttpServletRequest req,
12.        HttpServletResponse resp) {
13.        // store file to resourceDirectory (code not shown)
14.
15.    }
16.
17.    public void doDelete(HttpServletRequest req,
18.        HttpServletResponse resp) {
19.        // remove file from resourceDirectory (code not shown)
20.
21.    }
22.
23.    }
24. }
```

As a maintenance feature, you have created this servlet to allow you to upload and remove files on your web server. Unfortunately, while testing this servlet, you try to upload a file using an HTTP request and on this servlet, the web container returns a 404 status. What is wrong with this servlet?

- A. HTTP does NOT support file upload operations.
- B. The servlet constructor must NOT have any parameters.
- C. The servlet needs a service method to dispatch the requests to the helper methods.
- D. The doPut and doDelete methods do NOT map to the proper HTTP methods.

**Answer: B**