



# 1Z0-895<sup>Q&As</sup>

Java EE 6 Enterprise JavaBeans Developer Certified Expert

**Pass Oracle 1Z0-895 Exam with 100% Guarantee**

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.geekcert.com/1Z0-895.html>

100% Passing Guarantee  
100% Money Back Assurance

Following Questions and Answers are all new published by Oracle  
Official Exam Center

-  **Instant Download** After Purchase
-  **100% Money Back** Guarantee
-  **365 Days** Free Update
-  **800,000+** Satisfied Customers





## QUESTION 1

Given singleton bean FooEJB:

```
@Singleton
public class FooEJB {

    public long convertDaysToHours(long numDays) {
        if( numDays < 0 ) {
            throw new EJBException("Invalid num days = " + numDays);
        }

        return numDays * 24;
    }
}

A caller acquires an EJB reference to FooEJB in variable fooRef and executes:

100.
101. try {
102.     fooRef.convertDaysToHours(-1);
103. } catch(Throwable t) {}
104.
105. fooRef.convertDaysToHours(10);
106.
```

How many distinct FooEJB bean instances will be used to process the code on the lines 101-105?

- A. 0
- B. 1
- C. 2

Correct Answer: B

Java has several design patterns Singleton Pattern being the most commonly used. Java Singleton pattern belongs to the family of design patterns, that govern the instantiation process. This design pattern proposes that at any time there can only be one instance of a singleton (object) created by the JVM.

## QUESTION 2

Which two statements are correct about stateless session beans? (Choose two.)

- A. The bean class may declare instance variables.
- B. The lifetime of the bean instance is controlled by the client.
- C. The container may use the same bean instance to handle multiple business method invocations at the same time.
- D. The container may use the same bean instance to handle business method invocations requested by different clients, but not concurrently.



Correct Answer: AC

\* A: Stateless session beans are EJB's version of the traditional transaction processing applications, which are executed using a procedure call. The procedure executes from beginning to end and then returns the result. Once the procedure is done, nothing about the data that was manipulated or the details of the request are remembered. There is no state.

These restrictions don't mean that a stateless session bean can't have instance variables and therefore some kind of internal state. There's nothing that prevents you from keeping a variable that tracks the number of times a bean has been called or that tracks data for debugging. An instance variable can even hold a reference to a live resource like a URL connection for writing debugging data, verifying credit cards, or anything else that might be useful.

A stateless session bean is relatively easy to develop and also very efficient.

C:

Stateless session beans require few server resources because they are neither persistent nor dedicated to one client. Because they aren't dedicated to one client, many EJB objects can use just a few instances of a stateless bean. A

stateless session bean does not maintain conversational state relative to the EJB object it is servicing, so it can be swapped freely between EJB objects. As soon as a stateless instance services a method invocation, it can be swapped to

another EJB object immediately. Because there is no conversational state, a stateless session bean doesn't require passivation or activation, further reducing the overhead of swapping. In short, they are lightweight and fast!

\* The Lifecycle of a Stateless Session Bean Because a stateless session bean is never passivated, its lifecycle has only two stages: nonexistent and ready for the invocation of business methods. The EJB container typically creates and maintains a pool of stateless session beans, beginning the stateless session bean's lifecycle. The container performs any dependency injection and then invokes the method annotated @PostConstruct, if it exists. The bean is now ready to have its business methods invoked by a client.


At the end of the lifecycle, the EJB container calls the method annotated @PreDestroy, if it exists (not B). The bean's instance is then ready for garbage collection.

### QUESTION 3

Assume an EJB application is comprised of the following EJB fragment:

```
@Stateless
@LocalBean
public class InventoryReportBean {

    public Report generateInventoryReport() {
        //perform db intensive operations
    }
}
```



You have been asked to convert the type of InventoryReportBean into a singleton session bean. How would you achieve this task?

Exhibit C:



```
<?xml version="1.0" encoding="UTF-8"?>

<ejb-jar xmlns = "http://java.sun.com/xml/ns/javaee"
version = "3.1"
xmlns:xsi = "http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation = "http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/ejb-jar_3_1.xsd">
  <enterprise-beans>
    <session>
      <ejb-name>InventoryReportBean</ejb-name>
      <ejb-class>com.acme.InventoryReportBean</ejb-class>
      <session-type>Singleton</session-type>
    </session>
  </enterprise-beans>
</ejb-jar>
```



Exhibit D:

```
<?xml version="1.0" encoding="UTF-8"?>

<ejb-jar xmlns = "http://java.sun.com/xml/ns/javaee"
version = "3.1"
xmlns:xsi = "http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation = "http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/ejb-jar_3_1.xsd">
  <enterprise-beans>
    <session>
      <ejb-name>InventoryReportBean</ejb-name>
      <ejb-class>com.acme.InventoryReportBean</ejb-class>
      <session-type>Singleton</session-type>
      <override-type>True</override-type>
    </session>
  </enterprise-beans>
</ejb-jar>
```



- A. Keep InventoryReportBean as it is, modifying the internal structure to function as a singleton
- B. Change the @Stateless annotation of InventoryReportBean to @Singleton
- C. Create an ejb-jar.xml file, and override the annotation configuration information as in exhibit C above.
- D. Create an ejb-jar.xml file, and override the annotation configuration information as in exhibit D above.

Correct Answer: D

Note the line with

#### QUESTION 4

Which statement is true about both stateful session beans and stateless session beans? (Select Two)

- A. Bean instance are NOT required to survive container crashes.
- B. Any bean instance must be able to handle concurrent invocations from different threads.
- C. A bean with bean-managed transactions must commit or roll back any transaction before returning from a business method.
- D. The container passivates and activates them using methods annotated with @PrePassivate and @PostActivate



annotations.

Correct Answer: AC

Note:

\* Session beans can either be stateful or stateless. With stateful beans, the EJB container saves internal bean data during and in between method calls on the client's behalf. With stateless beans, the clients may call any available instance of an instantiated bean for as long as the EJB container has the ability to pool stateless beans. This enables the number of instantiations of a bean to be reduced, thereby reducing required resources.

Incorrect:

B: Stateful session beans maintain state both within and between transactions. Each stateful session bean is therefore associated with a specific client.

D: `@PrePassivate(javax.ejb.PrePassivate)` :

If a stateful session bean instance is idle for too long, the container might passivate it and store its state to a cache.

The method tagged by this annotation is called before the container passivates the bean instance. This annotation is only applicable to stateful session beans.

---

## QUESTION 5

Which two are programming restrictions in the EJB specification? (Choose two.)

- A. An enterprise bean must NOT attempt to load a native library.
- B. An enterprise bean must NOT declare static fields as final.
- C. An enterprise bean must NOT attempt to create a new security manager.
- D. An enterprise bean must NOT propagate a RuntimeException to the container.
- E. An enterprise bean must NOT attempt to obtain a `javax.naming.InitialContext`.

Correct Answer: AC

The following is a list of Java features that you should avoid, hence restricting their use in your EJB components' implementation code:

(A)

Loading native libraries.

(C)

Attempting to create or obtain a class loader, set or create a new security manager (C), stop the JVM, change the input, output, and error streams. That restriction enforces security and maintains the EJB container's ability to manage the runtime environment.

(not B) Using static, nonfinal fields. Declaring all static fields in the EJB component as final is recommended. That ensures consistent runtime semantics so that EJB containers have the flexibility to distribute instances across multiple JVMs.



Reference: Programming restrictions on EJB

[1Z0-895 PDF Dumps](#)

[1Z0-895 Practice Test](#)

[1Z0-895 Braindumps](#)



To Read the [Whole Q&As](#), please purchase the [Complete Version](#) from [Our website](#).

## Try our product !

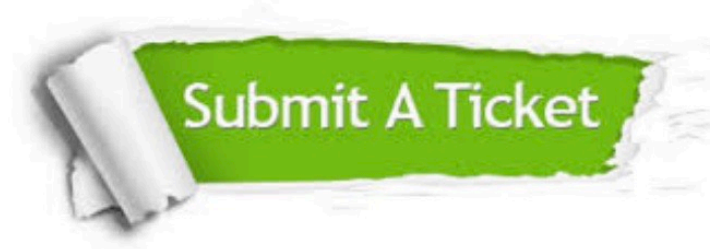
- 100% Guaranteed Success
- 100% Money Back Guarantee
- 365 Days Free Update
- Instant Download After Purchase
- 24x7 Customer Support
- Average 99.9% Success Rate
- More than 800,000 Satisfied Customers Worldwide
- Multi-Platform capabilities - [Windows](#), [Mac](#), [Android](#), [iPhone](#), [iPod](#), [iPad](#), [Kindle](#)

We provide exam PDF and VCE of Cisco, Microsoft, IBM, CompTIA, Oracle and other IT Certifications. You can view Vendor list of All Certification Exams offered:

<https://www.geekcert.com/allproducts>

## Need Help

Please provide as much detail as possible so we can best assist you.  
To update a previously submitted ticket:



 <p><b>One Year Free Update</b> Free update is available within One Year after your purchase. After One Year, you will get 50% discounts for updating. And we are proud to boast a 24/7 efficient Customer Support system via Email.</p>	 <p><b>Money Back Guarantee</b> To ensure that you are spending on quality products, we provide 100% money back guarantee for 30 days from the date of purchase.</p>	 <p><b>Security &amp; Privacy</b> We respect customer privacy. We use McAfee's security service to provide you with utmost security for your personal information &amp; peace of mind.</p>
---	---	--

Any charges made through this site will appear as Global Simulators Limited.  
All trademarks are the property of their respective owners.  
Copyright © geekcert, All Rights Reserved.