

**100%** Money Back  
**Guarantee**

**Vendor:** Oracle

**Exam Code:** 1Z0-898

**Exam Name:** Java Platform, Enterprise Edition 6 Java  
Persistence API Developer Certified Expert Exam

**Version:** Demo

### QUESTION NO: 1

Entity lifecycle callback methods may be defined in which three classes? (Choose three)

- A. Embedded classes
- B. Entity classes
- C. Abstract classes
- D. Entity listener classes
- E. Mapped superclasses
- F. Concrete non-entity superclasses

**Answer: B,D,E**

Reference:<http://stackoverflow.com/questions/3747268/how-to-inject-a-springs-service-bean-into-a-jpa-entity>(See the answer, first paragraph)

### QUESTION NO: 2

A developer wrote an entity class with the following method:

```
Private static Logger logger = Logger.getLogger ("myLogger");
```

```
@PrePersist
```

```
@PreUpdate
```

```
Public void doA ()
```

```
{ Logger.info ("A");
```

```
}
```

```
@PostPersist
```

```
@PostUpdate
```

```
Public void doB ()
```

```
{ logger.info ("B");
```

```
}
```

What will the log message contain when an application does the following?

1. Begins a transaction
2. Creates the entity
3. Persists the entity
4. Commits the transaction
5. Begins the entity data
6. Modifies the entity data
7. Merges the entity
8. Commits the second transaction

**A.** A

A

B

B

**B.** A

B

A

B

**C.** A

B

B

A

B

**D.** The application will throw an exception because multiple lifecycle callback annotations applied to a single method.

**Answer: B**

### **QUESTION NO: 3**

Given the following code:

```
Public void create () {  
try {  
doA () {  
} catch (PersistenceException e) {}
```

```
try (doB) ();  
  
} catch (PersistenceException e) {}  
  
}
```

Calling method doA will cause a NonUniqueResultException to be thrown. Calling method doB will cause an EntityExistsException to be thrown.

What two options describe what will happen when the create method is called within an application ' uses container managed transactions? (Choose two)

- A. Method doB will never be called.
- B. The current transaction will continue after doA executes.
- C. The current transaction will continue after doB executes.
- D. The current transaction will be marked for rollback when doA is called.
- E. The current transaction will be marked for rollback when doB is called.

**Answer: C,E**

#### **QUESTION NO: 4**

An application that uses pessimistic locking calls an updateData method that results in a LockTimeoutException being thrown. What three statements are correct? (Choose three)

- A. The current transaction continues.
- B. The current statement continues.
- C. The current transaction is rolled back.
- D. The current statement is rolled back.
- E. The LockTimeoutException can NOT be caught.
- F. The LockTimeoutException can be caught, and the updateData method retried.

**Answer: A,D,F**

#### **QUESTION NO: 5**

A developer has created a deep entity class hierarchy with many polymorphic relationships between entities. Which inheritance strategy, as defined by the inheritanceType enumerated type,

will be most performed in this scenario?

- A. Single table-per-class-hierarchy (InheritanceType.SINGLE\_TABLE)
- B. Joined-subclass (inheritanceType. JOINED)
- C. Table-per-concrete-class (inheritanceType.TABLE\_PER\_CLASS)
- D. Polymorphic join table (inheritanceType. POLYMORPHIC\_JOIN\_TABLE)

**Answer: C**

### QUESTION NO: 6

A developer is creating an entity which is mapped to a table that has a primary key constraint defined on two character columns and would like to use mapping defaults as much as possible to simplify the code.

Which two mapping options can be chosen? (Choose two.)

- A. Use an @id property that constructs a private field as a concatenation of two columns.
- B. Use a separate class to map those two columns and use an @idclass annotation to denote the primary key field or property in the entity.
- C. Use a separate @Embeddable class to map those two columns and use an @EmbeddedId annotation to denote a single primary key field or property in the entity.
- D. Use a separate @Embeddable class to map those two columns and add two fields or properties to the entity, each marked as @id, that correspond to the fields or properties in the embeddable class.
- E. Use a separate class to map those two columns. Specify that class using @Idclass annotation on the entity class. Add two fields or properties to the entity, each marked as @Id, that correspond to the fields or properties in that separate class.

**Answer: B,C**

### QUESTION NO: 7

A developer wants to model the grades for a student as a Map<course, integer>. Assume that Student and Course are entities, and that grades are modeled by integers.

Which of the following two statements are correct? (Choose two)

- A. The developer can model the grades as an element collection in the Student entity.
- B. The developer can model the grades as a oneToMany relationship in the Student entity.
- C. The mapping for the key of the map can be specified by the @MapKeycolumn annotation.
- D. The mapping for the value of the map can be specified by the @Column annotation.

**Answer: A,C**

#### **QUESTION NO: 8**

Consider a persistence application with the following orm.xml:

```
<entity - mappings ... >  
  <persistence - unit - metadat>  
    <persistence-unit-defaults>  
      <access > FIELD </ access>  
    </ persistence - unit -metadata>  
  </ entity - mappings>
```

What will be the effect of the above orm.xml?

- A. The access type for only those entities that have not explicitly specified @Access will be defaulted to field.
- B. The access type for all entities in the persistence unit will be changed to FIELD.
- C. The access type for all entities specified in this orm.xml will be changed to FIELD.
- D. The access type for only those entities defined in this orm.xml for which access is not specified will be defaulted to FIELD.

**Answer: D**

#### **QUESTION NO: 9**

A developer has created an application managed entity manager.

Which statement is correct?

- A. A new persistence context begins when the entity manager is created.
- B. A new persistence context begins when a new JTA transaction begins.
- C. A new persistence context begins when the entity manager is invoked in the context of a transaction.
- D. A new persistence context begins when the entity manager is invoked in the context of a resource-local transaction.

**Answer: B**

Reference:<http://docs.oracle.com/javaee/6/tutorial/doc/bnbqw.html#bnbra>

### QUESTION NO: 10

Given:

```
11. @PersistenceContext EntityManager em;
12. public boolean test(Order o) {
13.     boolean b = false;
14.     o = em.merge(o);
15.     em.remove(o);
16.     o = em.merge(o);
17.     b = em.contains(o);
18.     return b;
19. }
```

Which statement is correct?

- A. The method will return TRUE.
- B. The method will return FALSE.
- C. The method will throw an exception.
- D. The order instance will be removed from the database.

**Answer: C**

### QUESTION NO: 11

If an application uses an extended persistence context, which of the following is true?

- A. The persistence context exists until all transactions invoked by the EntityManager complete.
- B. The persistence context exists until all transactions invoked by the EntityManager complete and

the EntityManager.clear () method is invoked.

**C.** The persistence context exists until the EntityManager instance is closed.

**D.** The persistence context exists until the EntityManagerFactory instance is closed.

**Answer: C**

Reference:<http://java.boot.by/scbcd5-guide/ch07s03.html>

## QUESTION NO: 12

An application uses an application-managed entity manager. Which of the following is NOT true?

**A.** The application may specify whether the scope of the persistence context is extended.

**B.** The application must use EntityManagerFactory instances to create entity managers.

**C.** Entity manager instances must be explicitly closed.

**D.** The application may need to call EntityManager.joinTransaction If a JTA aware entity manager is used.

**Answer: A**

## QUESTION NO: 13

An application that uses container-managed transaction demarcation creates a query within an active transaction and receives a QueryTimeoutException. Which of those scenarios describes what happens to the active transaction?

**A.** The statement and the transaction continue.

**B.** The query is recreated within the current transaction.

**C.** The statement and the transaction are marked for rollback.

**D.** The statement is rolled back, but the transaction continues.

**Answer: C**

Reference:[http://www.google.com.pk/url?sa=t&rct=j&q=application+that+uses+container+managed+transaction+demarcation+creates+a+query+within+an+active+transaction+and+receives+a+QueryTimeoutException+active+transaction+&source=web&cd=10&ved=0CFwQFjAJ&url=http%3A%2F%2Fjgk-spring-recipes.googlecode.com%2Ffiles%2Fpersistence-2\\_0-final-spec.pdf&ei=gEc6T4iHHe-N4gS0p-WFCw&usg=AFQjCNHs2d7VPsYMyP7qpC4Z11Piigt2UQ](http://www.google.com.pk/url?sa=t&rct=j&q=application+that+uses+container+managed+transaction+demarcation+creates+a+query+within+an+active+transaction+and+receives+a+QueryTimeoutException+active+transaction+&source=web&cd=10&ved=0CFwQFjAJ&url=http%3A%2F%2Fjgk-spring-recipes.googlecode.com%2Ffiles%2Fpersistence-2_0-final-spec.pdf&ei=gEc6T4iHHe-N4gS0p-WFCw&usg=AFQjCNHs2d7VPsYMyP7qpC4Z11Piigt2UQ)(page 128, summary of exceptions)

### QUESTION NO: 14

The developer has defined the following entity class office:

```
@Entity
public class Office {
    @Id
    private int id;
    private String name;
    @OneToMany
    private List<Room> rooms;
}
```

Which of the following attributes will be in corresponding generated static metamodel class for the rooms' field?

- A. Public static volatile CollectionAttribute<Room> rooms;
- B. Public static volatile ListAttribute<Room> rooms;
- C. Public static volatile ListAttribute<Office, Room> rooms;
- D. Public static volatile SingleAttribute<Room> rooms;

**Answer: B**

### QUESTION NO: 15

Given two entities with many-to-many bidirectional association between them:

```
11. @Entity public class Employee {
12.     Collection projects;
13.     // more code here
14. }

and

11. @Entity public class Project{
12.     Set<Employee> emps;
13.     // more code here
14. }
```

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

# Trying 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 **69,000** Satisfied Customers Worldwide
- ★ Multi-Platform capabilities - **Windows, Mac, Android, iPhone, iPod, iPad, Kindle**

## Need Help

Please provide as much detail as possible so we can best assist you.

To update a previously submitted ticket:



 <b>One Year Free Update</b> <p>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>	 <b>Money Back Guarantee</b> <p>To ensure that you are spending on quality products, we provide 100% money back guarantee for 30 days from the date of purchase.</p>	 <b>Security &amp; Privacy</b> <p>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>
---	---	--

## [Guarantee & Policy](#) | [Privacy & Policy](#) | [Terms & Conditions](#)

Any charges made through this site will appear as Global Simulators Limited.

All trademarks are the property of their respective owners.

Copyright © 2004-2015, All Rights Reserved.