# Ourexam

**Higher Quality Better Service!**

We offer free update service for one year

Http://www.ourexam.com

**Exam** : **310-091**

**Title** : Sun Certified Bus.Component Developer Java EE Platform 5

**Version** : DEMO

1. Which is a valid PostConstruct method in a message-driven bean class?

A. @PostConstruct

```
public boolean init() { return true; }
```

B. @PostConstruct

```
private static void init() {}
```

C. @PostConstruct

```
private void init() {}
```

D. @PostConstruct

```
public static void init() {}
```

Answer: C

2. Given:

```
11. @Entity public class X {
12.     @Id int id;
13.     Y y;
14. }
```

A public class Y with NO Java Persistence annotations is defined in the same package.

Which statement is correct about these classes if NO other annotations and mapping descriptors are provided?

A. Class Y must be serializable.

B. Class Y must be marked as an entity.

C. The entity X is not defined correctly.The field y must be marked as @Lob.

D. Class Y must be accessed by a persistence application through a public interface.

Answer: A

3. A developer is working on a user registration application using EJB 3.0. A business method registerUser in stateless session bean RegistrationBean performs the user registration. The registerUser method executes in a transaction context started by the client. If some invalid user data causes the registration to fail, the client invokes registerUser again with corrected data using the same transaction. Which design can meet this requirement?

A. Have registerUser method call EJBContext.setRollbackOnly() method after registration fails.

B. Have registerUser method throw javax.ejb.EJBTransactionRequiredException after registration fails.

C. Have registerUser method throw EJBException without marking the transaction for rollback, after registration fails.

D. Create an application exception with the rollback attribute set to false and have registerUser method throw it after registration fails.

Answer: D

4. Which two class types must be implicitly or explicitly denoted in the persistence.xml descriptor as managed persistence classes to be included within a persistence unit? (Choose two.)

A. Entity classes

B. Interceptor classes

C. Embedded classes

D. Entity listener classes

Answer: AC

5. Which statement about the combination of mapping defaults, annotations, and XML descriptors is correct?

A. All mapping annotations must always be processed by the persistence provider.

B. Some annotations, like the @Entity annotation, must always be processed by the persistence provider.

C. The mapping information for an entity class specified by annotations and in XML descriptors must be distinct.

D. If multiple entity listeners are defined, the order in which they are invoked can be defined or overwritten in the XML descriptor.

Answer: D

6. Which statement about an entity instance lifecycle is correct?

A. A new entity instance is an instance with a fully populated state.

B. A detached entity instance is an instance with no persistent identity.

C. A removed entity instance is NOT associated with a persistence context.

D. A managed entity instance is the instance associated with a persistence context.

Answer: D

7. A developer implements a session bean with a method doStuff which behaves differently depending on the caller's security role. Only users in security roles "ADMIN" and "USER" are allowed to call the method. Assume that there is no security-related metadata in the deployment descriptor. Which two, taken in combination, are appropriate to accomplish this? (Choose two.)

A. Annotate method doStuff with @PermitAll.

B. Annotate method doStuff with @RolesAllowed({"ADMIN","USER"})

C. If EJBContext.getCallerPrincipal returns role "ADMIN", implement the behavior for users in role ADMIN.

D. If EJBContext.isCallerInRole("ADMIN") returns true, implement the behavior defined for users in role "ADMIN".

Answer: BD

8. A developer wants to create a portable EJB 3.0 application that includes the following class definition for the Entity Account:

```
11. @Entity
12. @EntityListeners(com.acme.AlertMonitor.class)
13. public class Account {
14.     // more code here
15.     @PrePersist
16.     protected void validateCreate() {/* more code here */}
17. }
```

Which statement is correct?

A. The validateCreate method may NOT throw runtime exceptions.

B. The validateCreate method can invoke the EntityManager.flush operation.

C. Methods of the class com.acme.AlertMonitor annotated with callback annotations must take an Object or Account instance as the only argument.

D. The above class definition is NOT correct. An entity cannot define a callback method like PrePersist

and use the EntityListeners annotation at the same time.

Answer: C

9. 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

10. The deployment descriptor for a stateless session bean that uses the isCallerInRole method reads as follows:

```
3. <security-role-ref>
4.      <role-name>manager</role-name>
5.      <role-link>humanresources</role-link>
6. <security-role-ref>
16. <security-role>
17.     <description>
18.         Is allowed to view and update all employee records.
19.     </description>
```

```
20.     <role-name>humanresources</role-name>
21. </security-role>
```

Which two roles are responsible for creating this deployment descriptor? (Choose two.)

A. Deployer

B. Bean Provider

C. System Administrator

D. Application Assembler

Answer: BD

11. Given two entities with a 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. }
```

What set of annotations correctly defines the association?

A. @ManyToMany on the projects field,

@ManyToMany(mappedBy="projects") on the emps field

B. @ManyToMany(mappedBy="emps") on the projects field,

@ManyToMany on the emps field

C. @ManyToMany(targetEntity=Project.class) on the projects field,

@ManyToMany(mappedBy="projects") on the emps field

D. @ManyToMany(targetEntity=Project.class) on the projects field,

@ManyToMany on the emps field

Answer: C

12. Given code snippets from two files:

```
public class Dog {
    public void onMessage(Message m) { System.out.print("1 "); }
}
```

and

```
@MessageDriven
class MessageDog extends Dog implements MessageDrivenBean {
    MessageDog(Message m) { System.out.print("2 "); }
}
```

Which four code changes, when used together, create a valid JMS message-driven bean? (Choose four.)

A. Make class MessageDog public.

B. Make the MessageDog constructor no-arg.

C. Make the MessageDog constructor public.

D. Move the onMessage method to class MessageDog.

E. Change MessageDog so that it is NOT a subclass of Dog.

F. Make class MessageDog implement MessageListener instead of MessageDrivenBean.

Answer: ABCF

13. Which statement about entity manager is true?

A. A container-managed entity manager must be a JTA entity manager.

B. An entity manager injected into session beans can use either JTA or resource-local transaction control.

C. An entity manager created by calling the EntityManagerFactory.createEntityManager method always uses JTA transaction control.

D. An entity manager obtained through resource injection in a stateful session bean can use a resource-local EntityTransaction for transaction control.

Answer: A

14. Which two options can be used to predefine Java Persistence queries for easy use? (Choose two.)

A. @NamedQuery annotation

B. @NamedNativeQuery annotation

C. using the named-query element in the XML descriptor

D. using the named-native-query element in the XML descriptor

Answer: AC

15. Given a set of CMT bean methods with the following transaction attributes:

Method M1=SUPPORTS

Method M2=REQUIRED

Method M3=NOT_SUPPORTED

Method M4=REQUIRES_NEW

And the following method invocation sequence:

Method M1 invokes Method M2

Method M2 invokes Method M3

Method M1 invokes Method M4

If Method M1 is invoked by a method that does NOT have a transaction context, which describes a possible scenario?

A. Method M1: no transaction

Method M2: new transaction

Method M3: no transaction

Method M4: new transaction

B. Method M1: no transaction

Method M2: Container throws EJBTransactionRequiredException

C. Method M1: new transaction

Method M2: runs in same transaction as M1

Method M3: Container throws TransactionNotSupportedException

D. Method M1: no transaction

Method M2: new transaction

Method M3: Container throws TransactionNotSupportedException

Answer: A

16. A developer wants to create a Java Persistence query that returns valid U.S. phone numbers (formatted as "123-456-7890" or "800-RUN-EJB3") from a collection of differently formatted international phone numbers. The developer needs only those numbers that begin with 303. Which WHERE clause is correct?

A. WHERE addr.phone LIKE '303_'

B. WHERE addr.phone LIKE '303%'

C. WHERE addr.phone LIKE '303-_-_'

D. WHERE addr.phone LIKE '303-%-%'

E. WHERE addr.phone LIKE '303-___-____'

F. WHERE addr.phone LIKE '303-%%%-%%%%'

Answer: E

17. A developer is working on a project that includes both EJB 2.1 and EJB 3.0 session beans. A lot of business logic has been implemented and tested in these EJB 2.1 session beans. Some EJB 3.0 session beans need to access this business logic.

Which design approach can achieve this requirement?

A. Add adapted home interfaces to EJB 3.0 session beans to make EJB 3.0 and EJB 2.1 session beans interoperable.

B. Add EJB 3.0 business interfaces to existing EJB 2.1 session beans and inject references to these business interfaces into EJB 3.0 session beans.

C. No need to modify existing EJB 2.1 session beans. Use the @EJB annotation to inject a reference to the EJB 2.1 home interface into the EJB 3.0 bean class.

D. No need to modify existing EJB 2.1 session beans. Use the @EJB annotation to inject a reference to the EJB 2.1 component interface into the EJB 3.0 bean class.

Answer: C

18. A developer is modifying an existing Java EE application that uses the JDBC API to access a database. This code must be used but cannot be changed, and new code needs to be added that uses the Java Persistence API.

Both parts of the code will execute in the same JTA transaction.

Which method of the EntityManager interface should the new code use to ensure that the data read by JDBC statements includes the state of managed entities?

A. Call refresh on each changed entity.

B. Call flush at the end of each business method.

C. Call find before accessing any managed entity.

D. Call lock at the beginning of each business method.

Answer: B

19. A developer wants to achieve the following two behaviors for an EJB 3.0 session bean:

(1) If the client calls a business method with a transaction context, the container will invoke the enterprise bean's method in the client's transaction context.

(2) If the client calls a business method without a transaction context, the container will throw the javax.ejb.EJBTransactionRequiredException.

Which transaction attribute should be used?

A. REQUIRED

B. SUPPORTS

C. MANDATORY

D. REQUIRES_NEW

E. NOT_SUPPORTED

Answer: C

20. Given a stateless session bean with container-managed transaction demarcation, from which two methods can a developer access another enterprise bean? (Choose two.)

A. bean constructor

B. Timeout callback method

C. PreDestroy lifecycle callback method

D. PostConstruct lifecycle callback method

E. business method from the business interface

Answer: BE