



1Z0-898^{Q&As}

Java EE 6 Java Persistence API Developer Certified Expert

Pass Oracle 1Z0-898 Exam with 100% Guarantee

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

<https://www.geekcert.com/1Z0-898.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

The developer wants to write a criteria query that will return the number of orders made by customer of each county.

Assume that customer is an entity with a unidirectional one-to-many relationship to the Order entity and that Address is an embeddable class, with an attribute country of type String.

Which one of the queries below correctly achieves this?

A. `CriteriaBuilder cb> = ... CriteriaQuery cq = cb.createQuery(); Root c = cq.from(Customer.class); Join o = c.join(Customer_.orders); cq.multiselect(cb.count(o), c.get(customer_.address).get(address_.country)) cq.groupBy(c.get(customer_.address).get(address_.country))`

B. `CriteriaBuilder cb> = ... CriteriaQuery cq = cb.createQuery(); Root c = cq.from(Customer.class); cq.select(cb.count(c.join(customer_.Orders)), c.get(customers(o), c.get(customer_.address).get(Address_.'country'))); (c.get(Customer_.address).get(address_.country));`

C. `CriteriaBuilder cb> = ... CriteriaQuery cq = cb.createQuery(); Root c = cq.from(Customer.class); Join o = c.join(Customer_.orders); cq.select(cb.count(o)); cq.groupBy(c.get(Customer.address) - get(Address_.country)) ;`

D. `CriteriaBuilder cb = ... CriteriaQuery cq = cb.createQueryO; Root c = cq.from(Customer.class); Root c = cq . from (Customer . class) ,- Join o = c.join(Customer_.orders); Join country= c.join(Customer,.address) .join(Address cq.multiselect(cq.count(o), country)); cq.groupBy(c.get(Customer.address)- get (Address_ . country)) ;`

Correct Answer: A

Reference:<http://www.jarvana.com/jarvana/view/org/apache/openjpa/openjpa-persistence-jdbc/2.0.0/openjpa-persistence-jdbc-2.0.0-testsources.jar!/org/apache/openjpa/persistence/criteria/TestTypesafeCriteria.java?format=ok>

QUESTION 2

A developer wants to create a Java Persistence query that will include a subquery. Which three are true? (Choose three.)

- A. Subqueries can be used in a FROM clause.
- B. Subqueries can be used in a WHERE clause.
- C. The ANY expression can be used only with a subquery.
- D. The EXISTS expression can be used only with a subquery.
- E. The MEMBER expression can be used only with a subquery.

Correct Answer: BCD

QUESTION 3

Which one of the following queries selects the customer whose order has the highest total price?



A. CriteriaBuilder cb = ... Criteria Query cq = cb.createQuery(Customer.class); Root c = cq.from(Customer.class); Join o = c.join(Customer .orders); cq.select(c).distinct(true); Subquery sq = cq.subquery(Double.class); Root subo = cq.correlate(o); sq.select(cb.max(subo.get(Order_.totalPrice))); cq.where(cb.equal(o.get(Order_.totalPrice), cb.all(sq)));

B. CriteriaBuilder cb = ... CriteriaQuery cq = cb.createQuery(customer.class) Root c = cq.from(Customer.class); Join o = c.join(Customer .orders); cq.select(c).distinct(true); Subquery sq = cq.subquery(Double.class); Root subo = cq.correlate(o); sq.select(cb.max(subo.get(Order_.totalPrice))); cq.where(cb.equal(o.get(Order_.totalPrice), cb.all(sq)));

C. CriteriaBuilder cb = ... CriteriaQuery cq = cb.cteateQuery(Customer.class); Root c = cq.from(Customer.class); Join o = c.join(Customer .orders); cq.select(c).distinct(true); Subquery sq = cq.subquery(Double.class); Root subo = cq.correlate(o); sq.select(cb.max(subo.get(Order_.totalPrice))); cq.where(cb.equal(o.get(Order_.totalPrice), cb.all(sq)));

D. CriteriaBuilder cb = ... CriteriaQuery cq = cb.createQuery(Customer.class); Root c = cq.from(Customer.class); Join o = c.join(Customer_.orders); cq.select(c).distinct(true); Subquery sq = cq.subquery(Double.class); Root subo = sq.from(Order.class); sq. select (ci: . max (subo . get (Order_ . Total Price))) ; cq.where(sq.all(o.gei(Order_.totalPrice)));

Correct Answer: B

QUESTION 4

A developer who is designing entity classes to map a legacy database encounters a table called STUDENT_RECORD.

This table has two columns, STUDENT_ID and STUDENT_INFO_ID. The primary key of this table consists of both columns, and there is a unique constraint on each info column.

The STUDENT_ID column is foreign key to the STUDENT table and STUDENT_INFO_ID column is a foreign key to the STUDENT_DAT table.

What entity classes and relationships can the developer use to model these tables and relationship?(Choose two)

A. Model the student table as a student entity and STUDENT_DATA table StudentData entity. Model the STUDENT_RECORDS table as bidirectional many-to-many relationship between the student entity on the student data entity.

B. Model the STUDENT table as a Student entity and the STUDENT-DATA table as a StudentData entity. Model the STUDENT_RECORD table as a bidirectional one-to-one relationship between the student entity and the StudentData entity.

C. Model the STUDENT table as a Student entity and the STUDENT-DATA table as a StudentData entity. Model the Student-Records table as a student record entity. Create a many-to- one one relationship from the StudentRecord entity to the student entity and many-to-one relationship from the StudentRecord entity entity to the Student entity and many-to-one relationship from the student entity to the StudentData entity and one-to-many relationship from the StudentData entity to the StudentRecord entity.

D. Model the STUDENT table as a Student entity and the STUDENT-DATA table as a StudentData entity. Model the STUDENT-RECORD table as a StudentRecord entity. Create a bidirectional one-to-one relationship between the StudentRecord entity and bidirectional one-to- one relationship between the Student Record entity and the Student Data entity.

Correct Answer: AC



QUESTION 5

A developer wants to ensure that an entity's data is up-to-date with regard to the database. Which of the following statements is guaranteed to accomplish this?

- A. Call EntityManager.refresh on the entity.
- B. Add @cacheable (false) annotation on the entity class.
- C. Call EntityManager.find on the entity.
- D. Use a named query to retrieve the entity.

Correct Answer: C

[Latest 1Z0-898 Dumps](#)

[1Z0-898 VCE Dumps](#)

[1Z0-898 Practice Test](#)



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>One Year Free Update 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>Money Back Guarantee 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>Security & Privacy We respect customer privacy. We use McAfee's security service to provide you with utmost security for your personal information & 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.