Java Persistence API (JPA) Mock Exams

http://www.JavaChamp.com

Open Certification Plattform

Authors: N. Ibrahim, Y. Ibrahim

Introducing JavaChamp.com Website

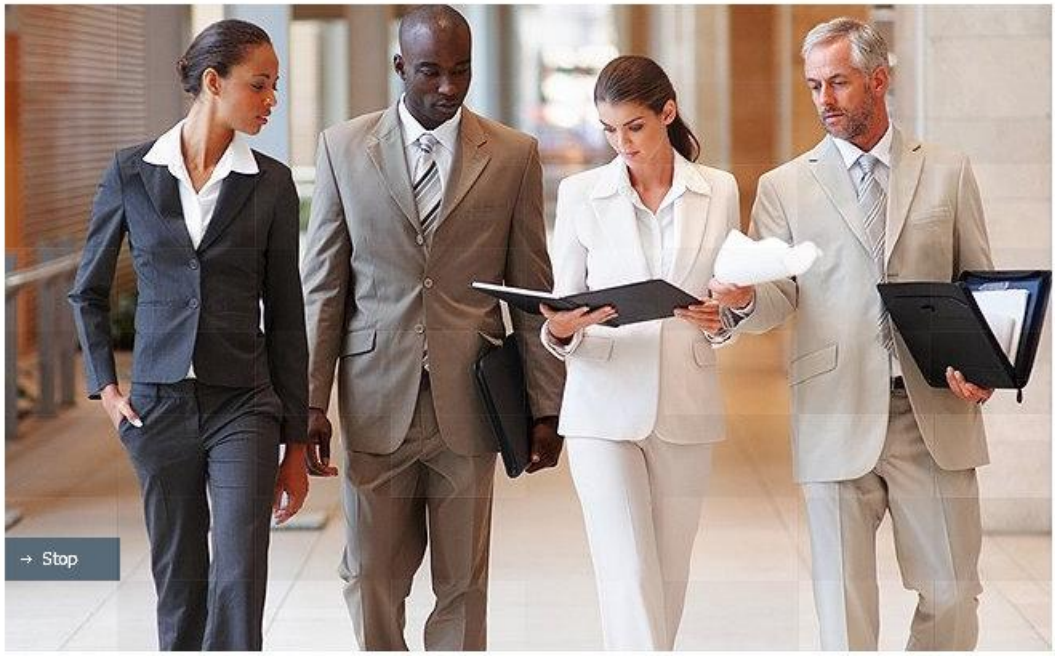
JavaChamp.com is an Open Certification Platform.
What does this mean?
JavaChamp is the best place to learn, share, and certify your professional skills.
We help you develop yourself in the field of computer science and programming
Here are the most significant features offered by JavaChamp:
Online Exams
Start Online Certification Exams in SCJP, SCEA, EJB, JMS, JPA and more...
Top quality mock exams for SCJP, SCEA, EJB, JMS, JPA. Start Express or topic-wise customized exam.
* We offer you unlimited free mock exams
* Exams cover subjects like SCJP, SCEA, EJB, JMS, JPA,..
* You can take as many exams as you want and at any time and for no charges
* Each exam contains 20 multiple choice questions
* You can save the exams taken in your exams history
* Your exams history saves the exams you took, the scores you got, time took you to finish the exam, date of examination and also saves your answers to the questions for later revision
* You can re-take the same exam to monitor your progress
* Your exams history helps the system to offer you variant new questions every time you take a new exam, therefore we encourage you to register and maintain an exams history
Network
Find guidance through the maze, meet Study-Mates, Coaches or Trainees...
Studying together is fun, productive and helps you in building your professional network and collecting leads
Bookshelf
JavaChamp Bookshelf full of PDF eBooks...
Download PDF books with a selected sample of the JavaChamp question bank in SCJP, SCEA, EJB, JMS and more or read it online
JavaChamp Profile
You may publish your profile and connect to your colleagues and friends.
Content Channel
Be an Author and get recognition, leads, and more...
Contributing to the JavaChamp question bank will earn your recognition of your professional skills, expands your network, introduce you to potential leads
Join Us
Join the fast growing JavaChamp Community now.
JavaChamp Community is young and very dynamic, we would be thrilled to welcome you on board :o)

JavaChamp.com

Java Open Certification Plattform

Services | About us | Contac Us

Home | Take Exam | Log in | Register | Book | Search



About us

JavaChamp.com is a Java Open Certification Plattform!

- It provides unique services to help you develop yourself in the field of computer science and programming
- It provides a web interface for the Java community to learn,interact and certify its acquired java experience.
- JavaChamp.com helps the Java developers to achieve the

Why to register in our community?

Registeration is optional and free but brings many adavantages

- By registering in JavaChamp.com you can save and keep track of your exams for later revision and hence to help you monitor your progress.

CHECK OUR eBOOK

JavaChamp.com's eBook encompasses a large number of multiple questions, which cover the subjects java,C and assembly. The book doesn't ...

More →

Exam in Session

Question 3 / 20

What is the expected output?

```
01. public class OuterTest {
02.
03.       public static void main(String args[]) {
04.      Airplane.BlackBox box = new Airplane().new BlackBox(); // line 1
05.           box.printVariables();
06.
07.     }
08. }
09.
10. class Airplane {
11.     String code = "11";
12.
13.     class BlackBox {
14.         String code = "22";
15.
16.         public void printVariables() {
17.                             System.out.print(code);
18.                     System.out.print(Airplane.this.code); // line 20
19.
20.         }
21.     }
22.
23. }
```

- Compile error because of line 1 (incorrect instantiation)
- Compile error because of line 20 (can't access Airplane's variables)
- 2222
- 1111
- 2211

Back | Next | Finish and evaluate | Abort Exam

Copyright

Online version published by JavaChamp.com Germany.

Table of Contents

1. Chapter: Java Persistence API

Chapter Description and Objectives

1. Entity Classes

Exam Category Description and Objectives

1.1.1. jpa entity manager methods

Author: Yasser Ibrahim

which of the following is NOT a jpa entity manager method?

Please choose only one answer:

- persist
- flush
- contains
- detach
- find

Check the answer of this question online on JavaChamp.com: jpa entity manager methods

1.1.2. JPA Entity life cycle example

Author: Yasser Ibrahim

Given the code snippet below, which of the following statements are true?

```
@PersistenceContext
 private EntityManager entityManager;

 public void registerUser() {
  User usr = new User();   // Line 1
  usr.setName(&amp;quot;JavaChamp&amp;quot;); // Line 2
  entityManager.persist(usr); // Line 3
 }
```

Please choose all the answers that apply:

- At line 1 the user instance is a managed Entity
- At line 1 the user instance is a new Entity
- At line 1 a new record will be inserted in the user table
- At line 2 the entityManager will immediately reflect user name change to the database
- At line 3 the entityManager will immediately reflect user name change to the database

Check the answer of this question online on JavaChamp.com: jpa entity life cycle example

1.1.3. jpa entity manager detached state

Author: Yasser Ibrahim

What causes the entity manager to move an Entity instance from the managed to the detached state?

Please choose only one answer:

- Invoke the detach() method of the entity manager on the instance
- Invoke the remove() method of the entity manager on the instance
- Invoke the unsynchronize() method of the entity manager on the instance
- when the associated persistence context ends at the end of the transaction boundary.

Check the answer of this question online on JavaChamp.com: jpa entity manager detached state

1.1.4. JPA Entity class life cycle example

Author: Yasser Ibrahim

Given the code snippet below, which of the following statements are true?

```
@PersistenceContext
 private EntityManager entityManager;

 public void updateUserProfile(Long usrId) {
  User usr = entityManager.find(User.class,usrId); // Line 1
  usr.setName(&amp;quot;JavaChamp&amp;quot;); // Line 2
  entityManager.merge(usr); // Line 3
  entityManager.persist(usr); // Line 4

 }
```

Please choose all the answers that apply:

- At line 1 the user instance is a managed Entity
- At line 1 the user instance is a detached Entity
- At line 1 the find method will return null if the input userID is not found
- line 3 is mandatory to synchronize the changes done at line 2 to the database
- line 4 is mandatory to synchronize the changes done at line 2 to the database

Check the answer of this question online on JavaChamp.com: jpa entity class life cycle example

1.1.5. JPA Entity class @Transient annotation

Author: Yasser Ibrahim

What is the impact of marking an attribute in a JPA Entity class with @Transient annotation?

Please choose only one answer:

- This attribute will should not be serialized by the EJB container
- This attribute will should not be persisted by the EJB container
- This attribute will should not be garbage collected by the EJB container

Check the answer of this question online on JavaChamp.com: jpa entity class @transient annotation

1.1.6. JPA Entity Synchronized life cycle states

Author: Yasser Ibrahim

In which life cycle state the JPA entity class is synchronized with the database?

Please choose only one answer:

- New
- Managed
- Synchronized
- Detached
- Removed

Check the answer of this question online on JavaChamp.com: jpa entity synchronized life cycle states

1.1.7. How to declare JPA Entity class? scenario

Author: Yasser Ibrahim

You want to write a JPA Entity class to model a databank table named COOPERATE_USER. Each user is uniquely identified in this table by his or her social security number SSN. Along with SSN the system keeps user name, job, address and birth date.

How to declare such JPA Entity class?

Please choose all the answers that apply:

- Write a CooperateUser public class
- Annotate the CooperateUser class with @Table(name=“ COOPERATE_USER”)
- Annotate the CooperateUser class with @Entity
- The CooperateUser class must implement Serializable
- Define private attributes ssn, job, name, addess and birthdate
- Annotate the ' name' attibute with @Id

Check the answer of this question online on JavaChamp.com: how to declare jpa entity class?

1.1.8. jpa entity manager flush

Author: Yasser Ibrahim

Which method of the jpa entity manager would you use to force synchronizing the database with the entities of the persistence context?

Please choose only one answer:

- synchronize()
- flush()
- refresh()
- persist()

Check the answer of this question online on JavaChamp.com: jpa entity manager flush

1.1.9. jpa entity manager merge

Author: Yasser Ibrahim

What causes the entity manager to move an Entity instance from the detached to the managed state?

Please choose only one answer:

- Invoke the persist() method of the entity manager on the instance
- Invoke the merge() method of the entity manager on the instance
- Invoke the manage() method of the entity manager on the instance
- Invoke the remove() method of the entity manager on the instance

Check the answer of this question online on JavaChamp.com: jpa entity manager merge

1.1.10. jpa entity manager refresh

Author: Yasser Ibrahim

Which method of the jpa entity manager would you use to force synchronizing the persistence context from the database?

Please choose only one answer:

- synchronize()
- flush()
- refresh()
- persist()

Check the answer of this question online on JavaChamp.com: jpa entity manager refresh

1.1.11. JPA Entity state transitions

Author: Yasser Ibrahim

What are the possible state transitions for an Entity instance ?

Please choose all the answers that apply:

- from managed to detached
- from detached to removed
- from detached to new
- from new to detached
- from managed to removed

Check the answer of this question online on JavaChamp.com: jpa entity state transitions

1.1.12. How to declare JPA Entity class?

Author: Yasser Ibrahim

How to declare JPA Entity class?

Please choose all the answers that apply:

- Write a private final Entity class named after the mapped table
- Annotate the class with @Entity
- Annotate the primary key attribute or its getter method with @Id
- Define private attributes to map the table columns
- Write the finalize method

Check the answer of this question online on JavaChamp.com: jpa entity

1.1.13. persistence unit definition

Author: Yasser Ibrahim

What is considered optional while you define a persistence unit?

Please choose all the answers that apply:

- persistence.xml configuration file
- Entity classes annotated with @Entity
- The ORM definition (object relational mapping)
- entity manager configuration (e.g. toplink, hibernate)

Check the answer of this question online on JavaChamp.com: persistence unit definition

2. Entity association

Exam Category Description and Objectives

1.2.1. Fetch modes in Entity beans

Author: Java Champ

Which of the following statements about the fetch modes in Entity beans is FALSE?

Please choose only one answer:

- There are two fetch modes : EAGER and LAZY
- The default fetching mode for a field in an Entity bean annotated by @Basic is LAZY
- The default fetching mode for a field in an Entity bean annotated by @OneToMany is LAZY
- @Lob annotation does not have a default fetch mode

Check the answer of this question online on JavaChamp.com: fetch modes in entity beans

1.2.2. Cascade Mode Attributes in Entity Beans

Author: Java Champ

What is TRUE about the cascading and cascade mode attributes in Entity Beans?

Please choose all the answers that apply:

- Cascade mode attributes can be specified for the association annotaions (like @OneToMany) in an entity bean
- The cascading direction is from the target entity to the source entity
- PERSIST, DELETE and REFRESH are cascading mode attributes
- Refresh cascade causes to refresh the target entities of a relationship when refresh is invoked on the source entity of the relationship

Check the answer of this question online on JavaChamp.com: cascade mode attributes in entity beans

1.2.3. Entity association cardinality OneToOne bidirectional

Author: Yasser Ibrahim

Which @Entity association cardinality and direction is implemented in the following code snippet?

```
@Entity
public class Yard {
 @Id
 private int yardNo;
 @OneToOne(mappedBy=&amp;quot;backYard&amp;quot;)
 private House aHouse;
}

@Entity
public class House {
 @Id
 private int houseNo;
 @OneToOne
 private Yard backYard;
}
```

Please choose only one answer:

- @OneToOne unidirectional
- @OneToOne bidirectional
- @OneToOne no directional attribute specified

Check the answer of this question online on JavaChamp.com: entity association cardinality onetoone bidirectional

1.2.4. Entity association cardinality OneToOne unidirectional

Author: Yasser Ibrahim

Which @Entity association cardinality and direction is implemented in the following code snippet?

```
@Entity
public class Yard {
 @Id
 private int yardNo;
}

@Entity
public class House {
 @Id
 private int houseNo;
 @OneToOne
 private Yard backYard;
}
```

Please choose only one answer:

- @OneToOne unidirectional
- @OneToOne bidirectional
- @OneToOne no directional attribute specified

Check the answer of this question online on JavaChamp.com: entity association cardinality onetoone unidirectional

1.2.5. Entity association OneToMany bidirectional

Author: Yasser Ibrahim

Which @Entity association attributes are implemented in the following code snippet?

```
@Entity
public class Window {
 @Id
 private int winNo;
 @ManyToOne
 private House aHouse;
}

@Entity
public class House {
 @Id
 private int houseNo;
 @OneToMany(mappedBy=&amp;quot;aHouse&amp;quot;)
 private List&amp;lt;Window&amp;gt; windows;
}
```

Please choose only one answer:

- @OneToMany unidirectional
- @OneToMany bidirectional
- @OneToMany no directional attribute specified

Check the answer of this question online on JavaChamp.com: entity association onetomany bidirectional

1.2.6. what are the JPA @Entity association attributes?

Author: Yasser Ibrahim

what are the JPA @Entity association attributes?

Please choose all the answers that apply:

- Association validation
- Association multiplicity
- Association cascade behavior
- Association direction

Check the answer of this question online on JavaChamp.com: what are the jpa @entity association attributes?

1.2.7. Entity association OneToMany bidirectional example

Author: Yasser Ibrahim

What's true about the following @Entity association between House and Window?

```
@Entity
public class Window {
 @Id
 private int winNo;
 @ManyToOne
 private House aHouse;
}

@Entity
public class House {
 @Id
 private int houseNo;
 @OneToMany(mappedBy=&amp;quot;aHouse&amp;quot;)
 private List&amp;lt;Window&amp;gt; windows;
}
```

Please choose all the answers that apply:

- It's OneToMany unidirectional association
- It's OneToMany bidirectional association
- The association owner is the House class
- The association owner is the Window class

Check the answer of this question online on JavaChamp.com: entity association onetomany bidirectional example

1.2.8. Entity association ManyToMany bidirectional

Author: Yasser Ibrahim

Which @Entity association attributes are implemented in the following code snippet?

```
@Entity
public class Course {
 @Id
 private int courseNo;
 @ManyToMany
 private List&lt;Student&gt; studentList;
}

@Entity
public class Student {
 @Id
 private int StudentNo;
 @ManyToMany(mappedBy=&quot;studentList&quot;)
 private List&lt;Course&gt; CourseList;
}
```

Please choose only one answer:

- @ManyToMany unidirectional lazy fetch mode
- @ManyToMany bidirectional lazy fetch mode
- @ManyToMany bidirectional eager fetch mode
- @ManyToMany unidirectional eager fetch mode

Check the answer of this question online on JavaChamp.com: entity association manytomany bidirectional

1.2.9. Entity association OneToOne bidirectional

Author: Yasser Ibrahim

What's true about the following @Entity association between House and Yard?

```
@Entity
public class Yard {
 @Id
 private int yardNo;
 @OneToOne(mappedBy=&amp;quot;backYard&amp;quot;)
 private House aHouse;
}

@Entity
public class House {
 @Id
 private int houseNo;
 @OneToOne
 private Yard backYard;
}
```

Please choose all the answers that apply:

- It's OneToOne unidirectional association
- It's OneToOne bidirectional association
- The association owner is the House class
- The association owner is the Yard class

Check the answer of this question online on JavaChamp.com: entity association onetoone bidirectional