

Java Message Service (JMS) Mock Exams

<http://www.JavaChamp.com>

Open Certification Plattform

Authors: N. Ibrahim, Y. Ibrahim

Copyright (c) 2009-2010

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)



★ 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?

Registration is optional and free but brings many advantages

- 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](#)

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](#) →

Copyright

Copyright 2009 JavaChamp.com

Online version published by JavaChamp.com Germany.

DISCLAIMER

All services and content of JavaChamp.com are provided under JavaChamp.com terms of use on an "as is" basis, without warranty of any kind, either expressed or implied, including, without limitation, warranties that the provided services and content are free of defects, merchantable, fit for a particular purpose or non-infringing. The entire risk as to the quality and performance of the provided services and content is with you. In no event shall JavaChamp.com be liable for any damages whatsoever arising out of or in connection with the use or performance of the services. Should any provided services and content prove defective in any respect, you (not the initial developer, author or any other contributor) assume the cost of any necessary servicing, repair or correction. This disclaimer of warranty constitutes an essential part of these "terms of use". No use of any services and content of JavaChamp.com is authorized hereunder except under this disclaimer.

The detailed "terms of use" of JavaChamp.com can be found under:

<http://www.javachamp.com/public/termsOfUse.xhtml>

This work is licensed under a Creative Commons Attribution-Noncommercial-No Derivative Works 3.0 license.

The full license legal code can be found under:

<http://creativecommons.org/licenses/by-nc-nd/3.0/legalcode>

And a human-readable summary of the this license can be found under:

<http://creativecommons.org/licenses/by-nc-nd/3.0/>

According to the Creative Commons Attribution-Noncommercial-No Derivative Works 3.0 license You agree to the following:

You are free to share, copy, distribute and transmit the work under the following conditions:

- You must attribute the work to JavaChamp.com with a link to <http://www.javachamp.com>.
- You may not use this work for commercial purposes.
- You may not alter, transform, or build upon this work.

Table of Contents

1. Java Messaging Service

- Synchronous and Asynchronous Messaging
- Basics
- Point to Point and Publish/Subscribe Messagings

1. Chapter: Java Messaging Service

Chapter Description and Objectives

1. Synchronous and Asynchronous Messaging

Exam Category Description and Objectives

1.1.1. Synchronous and Asynchronous messaging in JMS

Author: Java Champ

Which is true about synchronous and asynchronous messaging in JMS?

Please choose all the answers that apply:

- a QueueReceiver or a TopicSubscriber in JMS, use the receive() method to consume a message synchronously
- a QueueReceiver or a TopicSubscriber in JMS, must use a message listener to consume a message synchronously
- a QueueReceiver or a TopicSubscriber in JMS, can consume a message asynchronously by using a message listener
- in asynchronous messaging, the onMessage() method of the consumer is invoked when a message arrives

Check the answer of this question online on JavaChamp.com: [synchronous and asynchronous messaging in jms java message service](#)

1.1.2. Synchronous and Asynchronous Communications

Author: [Java Champ](#)

What is TRUE about synchronous and asynchronous communications?

Please choose all the answers that apply:

- In an asynchronous communication, the requester of a service must wait for a response from the service
- In synchronous communication, the requester of a service must wait for a response from the service
- In synchronous communication, both parties must be active
- Synchronous communication overcomes asynchronous communication in terms of performance

Check the answer of this question online on [JavaChamp.com](#): [asynchronous and synchronous communications](#)

1.1.3. Synchronous and Asynchronous Consumers and Destinations

Author: Java Champ

Which of the following is FALSE?

Please choose only one answer:

- the asynchronous consumer is notified when new messages arrive
- The synchronous consumer client checks and collects messages from destination and doesn't wait for notifications from destination
- An asynchronous consumer client checks the destination in time-intervals for any new messages

Check the answer of this question online on JavaChamp.com: [asynchronous consumer versus synchronous consumer](#)

1.1.4. Asynchronous Communication Applications

Author: [Java Champ](#)

Which of the following applications are appropriate to communicate asynchronously?

Please choose all the answers that apply:

- Courses registration in a university, where courses availability must be checked by another application before registering the course
- Email services (sending and receiving emails)
- A client sends a document (such as an XML document) to a travel agency service requesting arrangements for a particular trip

Check the answer of this question online on [JavaChamp.com](#): [asynchronous communication use cases](#)

1.1.5. Asynchronous versus Synchronous Protocols

Author: [Java Champ](#)

Which is considered an asynchronous protocol?

Please choose only one answer:

- Remote Procedure Call (RPC)
- Remote Method Invocation (RMI)
- Java Messaging Service (JMS)
- HTTP

Check the answer of this question online on [JavaChamp.com](#): [asynchronous protocols](#)

2. Basics

Exam Category Description and Objectives

1.2.1. What is the role of a Session component in an JMS application?

Author: Java Champ

Which JMS messaging application component is responsible for producing and consuming messages, managing transactions, and handling acknowledgments?

Please choose only one answer:

- Connection
- ConnectionFactory
- Session
- Message Sender and Message Receiver

Check the answer of this question online on JavaChamp.com: [the role of a session in java message service](#)

1.2.2. Message Producers in JMS

Author: Java Champ

What is TRUE about Message Producers in JMS?

Please choose all the answers that apply:

- The Point To Point form of a message producer implements the TopicPublisher
- A message producer is an object created by a session and is used for sending messages to a destination
- to send a message using a QueueSender object, the send() method is called
- to send a message using a TopicPublisher object, the send() method is called
- to send a message using a TopicPublisher object, the publish() method is called

Check the answer of this question online on JavaChamp.com: [jms java message service queuesender](#)

1.2.3. Message Acknowledgement Ways

Author: Java Champ

What is not considered a way in messages acknowledgement?

Please choose only one answer:

- AUTO_ACKNOWLEDGE
- RECEIVER_ACKNOWLEDGE
- CLIENT_ACKNOWLEDGE
- DUPS_OK_ACKNOWLEDGE

Check the answer of this question online on JavaChamp.com: [messages acknowledgement ways in Java Message Service JMS](#)

1.2.4. What is the role of a JMS Destination component?

Author: Java Champ

What is the role of a JMS Destination component?

Please choose all the answers that apply:

- a client uses the destination component to specify the target of messages that it produces
- a client uses the destination component to specify the source of messages that it consumes.
- a destination component creates connections with the source clients
- a destination component creates connections with the destinations clients

Check the answer of this question online on JavaChamp.com: [what is the role of a java message service jms destination component?](#)

1.2.5. Messaging versus RMI

Author: Java Champ

What is TRUE about Messaging versus RMI (Remote Method Invocation)?

Please choose only one answer:

- Remote Method Invokation (RMI) enables communication in a loose coupling way, where the sender does not need to know anything about the receiver and vice versa
- Remote Method Invocation (RMI), which requires an invoker of a remote method to know about the other application
- Asynchronous messaging is considered a tightly coupled technology

Check the answer of this question online on JavaChamp.com: [messaging versus rmi \(remote method invocation\)](#)

1.2.6. Which JMS messaging application components are obtained via JNDI lookups?

Author: Java Champ

Which JMS messaging application components are obtained via JNDI lookups?

Please choose all the answers that apply:

- Connection
- ConnectionFactory
- Session
- MessageProducer
- Destination

Check the answer of this question online on JavaChamp.com: [which jms java message service components are obtained via jndi lookups?](#)

1.2.7. Components of a JMS messaging application

Author: [Java Champ](#)

What are the components of a JMS messaging application?

Please choose all the answers that apply:

- a Queue
- a Message Producer
- a WebService end point
- a Message Convertor
- a Connection

Check the answer of this question online on [JavaChamp.com](#): [what are the components of a jms messaging application?](#)

1.2.8. JMS Message Body Formats

Author: Java Champ

Which of the following is NOT a JMS message valid body format?

Please choose only one answer:

- ByteMessage
- ObjectMessage
- CharMessage
- TextMessage

Check the answer of this question online on JavaChamp.com: [jms java message service message valid body format](#)

1.2.9. What is the role of MessageSelector in a JMS application?

Author: Java Champ

What would you use to filter out messages that don't meet specified criteria?

Please choose only one answer:

- a MessageListener
- a MessageScanner
- a MessageSelector
- a MessagePattern

Check the answer of this question online on JavaChamp.com: [what is the role of a messageselector in java message service jms?](#)

1.2.10. What is a StreamMessage?

Author: [Java Champ](#)

What type of data does a StreamMessage carry?

Please choose only one answer:

- StreamMessage carries a stream of bytes
- StreamMessage carries a Serializable java object
- StreamMessage carries a stream of Java primitive types
- StreamMessage carries name-value pairs, where values can be Java primitives or their wrappers

Check the answer of this question online on [JavaChamp.com](#): [what type of data does a streammessage carry?](#)

1.2.11. Which method is called to receive messages in Java Message Service?

Author: [Java Champ](#)

Which method is called to receive messages in Java Message Service?

Please choose all the answers that apply:

- `receiveMessage()`
- `receive()`
- `receiveNoWait()`
- `acknowledge()`
- `onMessage()`

Check the answer of this question online on [JavaChamp.com](#): [which method is called to receive messages in java message service?](#)

1.2.12. JMS Administered objects

Author: Java Champ

Preconfigured JMS objects created by an administrator for the use of clients are:

Please choose all the answers that apply:

- messages
- destinations
- connections
- connection factories

Check the answer of this question online on JavaChamp.com: [what are the administered objects in jms java message service messaging application?](#)

3. Point to Point and Publish/Subscribe Messagings

Exam Category Description and Objectives

1.3.1. What design pattern best describes the Publish/Subscribe messaging?

Author: Java Champ

What design pattern best describes the Publish/Subscribe messaging?

Please choose only one answer:

- Proxy
- Bridge
- Observer
- Builder

Check the answer of this question online on JavaChamp.com: [what design pattern best describe the publish/subscribe messaging?](#)

1.3.2. Messaging with Timing Dependencies

Author: [Java Champ](#)

Which of the following messaging approaches requires no timing dependencies between the sender(s) and receiver(s)?

Please choose all the answers that apply:

- Point-to-Point Messaging
- durable Publish/Subscribe Messaging
- non durable Publish/Subscribe Messaging
- Remote Procedure Call

Check the answer of this question online on [JavaChamp.com](#): [timing dependencies in Publish/Subscribe Messaging](#)

1.3.3. Destination components in Point to Point messaging vs Publish/Subscribe messaging

Author: Java Champ

What is true about Destination components in Point to Point messaging and Publish/Subscribe messaging?

Please choose all the answers that apply:

- a destination in Point to Point messaging is called Topic
- a destination in Point to Point messaging has one consumer for each message
- messages in a destination of Point to Point messaging are consumed in FIFO order
- messages sent to a destination of Publish/Subscribe messaging must be generated by only one publisher but can be consumed by many subscribers

Check the answer of this question online on JavaChamp.com: [destination components in point to point messaging and publish/subscribe messaging](#)

