



1Z0-804^{Q&As}

Java SE 7 Programmer II

Pass Oracle 1Z0-804 Exam with 100% Guarantee

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

<https://www.pass4lead.com/1Z0-804.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**

Given:

```
public class SleepOtherThread {  
  
    public static void main(String[] args) throws InterruptedException {  
  
        Runnable r = new Runnable() {  
  
            public void run() {  
  
                System.out.print(Thread.currentThread().getName());  
  
            }  
  
        };  
  
        Thread t1 = new Thread(r, "One ");  
  
        t1.start();  
  
        t1.sleep(2000);  
  
        Thread t2 = new Thread(r, "Two ");  
  
        t2.start();  
  
        t2.sleep(1000);  
  
        System.out.print("Main ");  
  
    }  
  
}
```

What is the most likely result?

- A. Main One Two
- B. Main Two One
- C. One Two Main
- D. One Main Two
- E. Two Main One

Correct Answer: C

QUESTION 2



Which two forms of abstraction can a programmer use in Java?

- A. enums
- B. interfaces
- C. primitives
- D. abstract classes
- E. concrete classes
- F. primitive wrappers

Correct Answer: BD

*

When To Use Interfaces

An interface allows somebody to start from scratch to implement your interface or implement your interface in some other code whose original or primary purpose was quite different from your interface. To them, your interface is only incidental, something that have to add on to the their code to be able to use your package. The disadvantage is every method in the interface must be public. You might not want to expose everything.

*

When To Use Abstract classes An abstract class, in contrast, provides more structure. It usually defines some default implementations and provides some tools useful for a full implementation. The catch is, code using it must use your class as the base. That may be highly inconvenient if the other programmers wanting to use your package have already developed their own class hierarchy independently. In Java, a class can inherit from only one base class.

*

When to Use Both You can offer the best of both worlds, an interface and an abstract class. Implementors can ignore your abstract class if they choose. The only drawback of doing that is calling methods via their interface name is slightly slower than calling them via their abstract class name.

Reference: <http://mindprod.com/jgloss/interfacevsabstract.html>

QUESTION 3

Which statement creates a low overhead, low-contention random number generator that is isolated to thread to generate a random number between 1 and 100?

- A. `int i = ThreadLocalRandom.current().nextInt(1, 101);`
- B. `int i = ThreadSafeRandom.current().nextInt(1, 101);`
- C. `int i = (int) Math.random()*100+1;`
- D. `int i = (int) Math.random(1, 101);`
- E. `int i = new random().nextInt(100)+1;`



Correct Answer: A

```
public class ThreadLocalRandom  
  
extends Random\{'
```

A random number generator isolated to the current thread. Like the global Random generator used by the Math class, a ThreadLocalRandom is initialized with an internally generated seed that may not otherwise be modified. When

applicable, use of ThreadLocalRandom rather than shared Random objects in concurrent programs will typically encounter much less overhead and contention. Use of ThreadLocalRandom is particularly appropriate when multiple tasks (for

example, each a ForkJoinTask) use random numbers in parallel in thread pools.

Usages of this class should typically be of the form:

ThreadLocalRandom.current().nextX(...) (where X is Int, Long, etc). When all usages are of this form, it is never possible to accidentally share a ThreadLocalRandom across multiple threads.

This class also provides additional commonly used bounded random generation methods.

Reference: Class ThreadLocalRandom

QUESTION 4

Given the two Java classes:

```
public class Word {  
  
private Word(int length) {}  
  
protected Word(String w) {}  
  
}  
  
public class Buzzword extends Word {  
  
public Buzzword() {  
  
// Line ***, add code here  
  
}  
  
public Buzzword(String s) {  
  
super(s);  
  
}  
  
}
```

Which two code snippets, added independently at line ***, can make the Buzzword class compile?

A. this ();



- B. this (100);
- C. this ("Buzzword");
- D. super ();
- E. super (100);
- F. super ("Buzzword");

Correct Answer: AF

QUESTION 5

Given the code fragment:

```
public class Base {  
    BufferedReader br;  
    String record;  
    public void process() throws FileNotFoundException {  
        br = new BufferedReader(new FileReader("manual.txt"));  
    }  
}  
  
public class Derived extends Base {  
    // insert code here. Line ***  
    public static void main(String[] args) {  
        try {  
            new Derived().process();  
        } catch (Exception e) { }  
    }  
}
```

Which code fragment inserted at line ***, enables the code to compile?

- A. `public void process () throws FileNotFoundException, IOException { super.process (); while ((record = br.readLine()) != null) { System.out.println(record); } }`
- B. `public void process () throws IOException { super.process (); while ((record = br.readLine()) != null) { System.out.println(record); } }`
- C. `public void process () throws Exception { super.process (); while ((record = br.readLine()) != null) {`



```
System.out.println(record); } }
```

```
D. public void process () { try { super.process (); while ((record = br.readLine()) != null) { System.out.println(record); } } catch (IOException | FileNotFoundException e) { } }
```

```
E. public void process () { try { super.process (); while ((record = br.readLine()) != null) { System.out.println(record); } } catch (IOException e) { } }
```

Correct Answer: E

Incorrect answer:

D: exception java.io.FileNotFoundException has already been caught

Alternatives in a multi-catch statement cannot be related to subclassing Alternative java.io. FileNotFoundException is a subclass of alternative java.io.IOException

[1Z0-804 PDF Dumps](#)

[1Z0-804 VCE Dumps](#)

[1Z0-804 Exam Questions](#)



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](#)

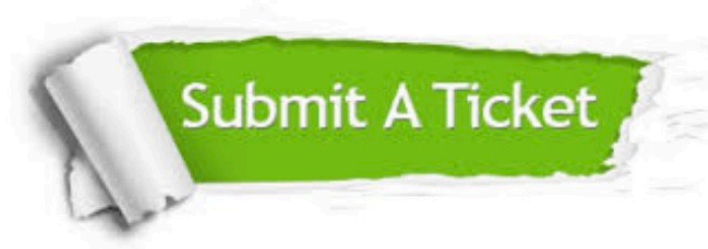
We provide exam PDF and VCE of Cisco, Microsoft, IBM, CompTIA, Oracle and other IT Certifications. You can view Vendor list of All Certification Exams offered:

<https://www.pass4lead.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 © pass4lead, All Rights Reserved.