

# KillTest |

質量更高 服務更好

## 學習資料



<http://www.killtest.net>

一年免費更新服務

**Exam : 310-065**

**Title : Sun Certified Programmer  
for the Java 2 Platform.  
SE6.0**

**Version : DEMO**

1. Given:

```
1. public class Threads2 implements Runnable {  
2.  
3.     public void run() {  
4.         System.out.println("run.");  
5.         throw new RuntimeException("Problem");  
6.     }  
7.     public static void main(String[] args) {  
8.         Thread t = new Thread(new Threads2());  
9.         t.start();  
10.        System.out.println("End of method.");  
11.    }  
12. }
```

Which two can be results? (Choose two.)

- A. java.lang.RuntimeException: Problem
- B. run. java.lang.RuntimeException: Problem
- C. End of method. java.lang.RuntimeException: Problem
- D. End of method. run. java.lang.RuntimeException: Problem
- E. run. java.lang.RuntimeException: ProblemEnd of method.

**Answer:** DE

2. Which two statements are true? (Choose two.)

- A. It is possible for more than two threads to deadlock at once.
- B. The JVM implementation guarantees that multiple threads cannot enter into a deadlocked state.
- C. Deadlocked threads release once their sleep() method's sleep duration has expired.
- D. Deadlocking can occur only when the wait(), notify(), and notifyAll() methods are used incorrectly.
- E. It is possible for a single-threaded application to deadlock if synchronized blocks are used incorrectly.
- F. If a piece of code is capable of deadlocking, you cannot eliminate the possibility of deadlocking by inserting invocations of Thread.yield().

**Answer:** AF

3. Given:

```
7.void waitForSignal() {  
8.    Object obj = new Object();  
9.    synchronized (Thread.currentThread()) {  
10.        obj.wait();  
11.        obj.notify();  
12.    }  
13.}
```

Which statement is true?

- A. This code can throw an InterruptedException.
- B. This code can throw an IllegalMonitorStateException.
- C. This code can throw a TimeoutException after ten minutes.
- D. Reversing the order of obj.wait() and obj.notify() might cause this method to complete normally.
- E. A call to notify() or notifyAll() from another thread might cause this method to complete normally.
- F. This code does NOT compile unless "obj.wait()" is replaced with "((Thread) obj).wait()".

**Answer:** B

4. Click the Exhibit button.What is the output if the main() method is run?

Given:

```
10. public class Starter extends Thread {  
11.     private int x = 2;  
12.     public static void main(String[] args) throws Exception {  
13.         new Starter().makeItSo();  
14.     }  
15.     public Starter() {  
16.         x = 5;  
17.         start();  
18.     }
```

```
19.    public void makeItSo() throws Exception {  
20.        join();  
21.        x = x - 1;  
22.        System.out.println(x);  
23.    }  
24.    public void run() { x *= 2; }  
25. }
```

- A. 4
- B. 5
- C. 8
- D. 9
- E. Compilation fails.
- F. An exception is thrown at runtime.
- G. It is impossible to determine for certain.

**Answer:** D

5. Given:

```
11.class PingPong2 {  
12.synchronized void hit(long n) {  
13.for(int i = 1; i < 3; i++)  
14.System.out.print(n + "-" + i + " ");  
15.}  
16.  
17.public class Tester implements Runnable {  
18.static PingPong2 pp2 = new PingPong2();  
19.public static void main(String[] args) {
```

```
20.new Thread(new Tester()).start();  
21.new Thread(new Tester()).start();  
22.}  
  
23.public void run() { pp2.hit(Thread.currentThread().getId()); }  
  
24.}
```

Which statement is true?

- A. The output could be 5-1 6-1 6-2 5-2
- B. The output could be 6-1 6-2 5-1 5-2
- C. The output could be 6-1 5-2 6-2 5-1
- D. The output could be 6-1 6-2 5-1 7-1

**Answer:** B

6. Given:

```
1. public class Threads4 {  
2. public static void main (String[] args) {  
3. new Threads4().go();  
4. }  
5. public void go() {  
6. Runnable r = new Runnable() {  
7. public void run() {  
8. System.out.print("foo");  
9. }  
10. };  
11. Thread t = new Thread(r);  
12. t.start();  
13. t.start();  
14. }  
15. }
```

What is the result?

- A. Compilation fails.
- B. An exception is thrown at runtime.
- C. The code executes normally and prints "foo".

D. The code executes normally, but nothing is printed.

**Answer:** B

7. Given:

```
11. public abstract class Shape {  
12.     private int x;  
13.     private int y;  
14.     public abstract void draw();  
15.     public void setAnchor(int x, int y) {  
16.         this.x = x;  
17.         this.y = y;  
18.     }  
19. }
```

Which two classes use the Shape class correctly? (Choose two.)

A. public class Circle implements Shape {private int radius; }

B. public abstract class Circle extends Shape { private int radius; }

C. public class Circle extends Shape { private int radius; public void draw(); }

D. public abstract class Circle implements Shape { private int radius; public void draw(); }

E. public class Circle extends Shape { private int radius; public void draw() /\* code here \*/ }

F. public abstract class Circle implements Shape { private int radius; public void draw() { /\* code here \*/ }}

**Answer:** BE

8. Given:

```
11. public class Barn {  
12.     public static void main(String[] args) {  
13.         new Barn().go("hi", 1);  
14.         new Barn().go("hi", "world", 2);  
15.     }  
16.     public void go(String... y, int x) {
```

```
17.     System.out.print(y[y.length - 1] + " ");
18. }
19. }
```

What is the result?

- A. hi hi
- B. hi world
- C. world world
- D. Compilation fails.
- E. An exception is thrown at runtime.

**Answer:** D

9.Given:

```
10 class Nav{
11. public enum Direction { NORTH, SOUTH, EAST, WEST }
12. }
13. public class Sprite{
14.     // insert code here
15. }
```

Which code, inserted at line 14, allows the Sprite class to compile?

- A. Direction d = NORTH;
- B. Nav.Direction d = NORTH;
- C. Direction d = Direction.NORTH;
- D. Nav.Direction d = Nav.Direction.NORTH;

**Answer:** D

10. Click the Exhibit button.

```
1. public interface A {
2.     public void doSomething(String thing);
```

```
3. }

1. public class AImpl implements A {
2.     public void doSomething(String msg) { }
3. }

1. public class B {
2.     public A doit() {
3.         // more code here
4.     }
5.
6.     public String execute() {
7.         // more code here
8.     }
9. }

1. public class C extends B {
2.     public AImpl doit() {
3.         // more code here
4.     }
5.
6.     public Object execute() {
7.         // more code here
8.     }
9. }
```

Which statement is true about the classes and interfaces in the exhibit?

- A. Compilation will succeed for all classes and interfaces.
- B. Compilation of class C will fail because of an error in line 2.
- C. Compilation of class C will fail because of an error in line 6.
- D. Compilation of class AImpl will fail because of an error in line 2.

**Answer:** C

- 11. Click the Exhibit button.

```
11. class Person {  
12.     String name = "No name";  
13.     public Person(String nm) { name = nm; }  
14. }  
15.  
16. class Employee extends Person {  
17.     String empID = "0000";  
18.     public Employee(String id) { empID = id; }  
19. }  
20.  
21. public class EmployeeTest {  
22.     public static void main(String[] args) {  
23.         Employee e = new Employee("4321");  
24.         System.out.println(e.empID);  
25.     }  
26. }
```

What is the result?

- A. 4321
- B. 0000
- C. An exception is thrown at runtime.
- D. Compilation fails because of an error in line 18.

**Answer:** D

12. Given:

```
11. public class Rainbow {  
12.     public enum MyColor {  
13.         RED(0xff0000), GREEN(0x00ff00), BLUE(0x0000ff);  
14.     private final int rgb;  
15.     MyColor(int rgb) { this.rgb = rgb; }
```

```
16. public int getRGB() { return rgb; }

17.};

18. public static void main(String[] args) {

19. // insert code here

20.}

21.}
```

Which code fragment, inserted at line 19, allows the Rainbow class to compile?

- A. MyColor skyColor = BLUE;
- B. MyColor treeColor = MyColor.GREEN;
- C. if(RED.getRGB() < BLUE.getRGB()) {}
- D. Compilation fails due to other error(s) in the code.
- E. MyColor purple = new MyColor(0xff00ff);
- F. MyColor purple = MyColor.BLUE + MyColor.RED;

**Answer:** B

13. Given:

```
11. class Mud {

12.   // insert code here

13.   System.out.println("hi");

14 }

15.}
```

And the following five fragments:

```
public static void main(String...a) {

public static void main(String.* a) {

public static void main(String... a) {

public static void main(String[]... a) {
```

```
public static void main(String...[] a) {
```

How many of the code fragments, inserted independently at line 12, compile?

A. 0

B. 1

C. 2

D. 3

E. 4

F. 5

**Answer:** D

14. Given:

```
5. class Atom {
```

```
6.     Atom() { System.out.print("atom "); }
```

```
7. }
```

```
8. class Rock extends Atom {
```

```
9.     Rock(String type) { System.out.print(type); }
```

```
10. }
```

```
11. public class Mountain extends Rock {
```

```
12.     Mountain() {
```

```
13.         super("granite ");
```

```
14.         new Rock("granite ");
```

```
15.     }
```

```
16.     public static void main(String[] a) { new Mountain(); }
```

```
17. }
```

What is the result?

A. Compilation fails.

B. atom granite

- C. granite granite
- D. atom granite granite
- E. An exception is thrown at runtime.
- F. atom granite atom granite

**Answer:** F

15. Given:

```
1. interface TestA { String toString(); }
2. public class Test {
3.     public static void main(String[] args) {
4.         System.out.println(new TestA() {
5.             public String toString() { return "test"; }
6.         });
7.     }
8. }
```

What is the result?

- A. test
- B. null
- C. An exception is thrown at runtime.
- D. Compilation fails because of an error in line 1.
- E. Compilation fails because of an error in line 4.
- F. Compilation fails because of an error in line 5.

**Answer:**A

16. Given:

```
11. public static void parse(String str) {
12.     try {
13.         float f = Float.parseFloat(str);
14.     } catch (NumberFormatException nfe) {
15.         f = 0;
16.     } finally {
```

```
17    System.out.println(f);
18. }
19. }
20. public static void main(String[] args) {
21.     parse("invalid");
22. }
```

What is the result?

- A. 0.0
- B. Compilation fails.
- C. A ParseException is thrown by the parse method at runtime.
- D. A NumberFormatException is thrown by the parse method at runtime.

**Answer:** B

17. Given:

```
1. public class Blip {
2.     protected int blipvert(int x) { return 0; }
3. }
4. class Vert extends Blip {
5. // insert code here
6. }
```

Which five methods, inserted independently at line 5, will compile? (Choose five.)

- A. public int blipvert(int x) { return 0; }
- B. private int blipvert(int x) { return 0; }
- C. private int blipvert(long x) { return 0; }
- D. protected long blipvert(int x) { return 0; }
- E. protected int blipvert(long x) { return 0; }
- F. protected long blipvert(long x) { return 0; }
- G. protected long blipvert(int x, int y) { return 0; }

**Answer:** ACEFG

18. Given:

```
1. class Super {  
2.     private int a;  
3.     protected Super(int a) { this.a = a; }  
4. }
```

```
11. class Sub extends Super {  
12.     public Sub(int a) { super(a); }  
13.     public Sub() { this.a = 5; }  
14. }
```

Which two, independently, will allow Sub to compile? (Choose two.)

- A. Change line 2 to:public int a;
- B. Change line 2 to: protected int a;
- C. Change line13 to:public Sub() { this(5); }
- D. Change line 13 to:public Sub() { super(5); }
- E. Change line13 to:public Sub() { super(a); }

**Answer:** CD

19. Which Man class properly represents the relationship "Man has a best friend who is a Dog"?

- A. class Man extends Dog { }
- B. class Man implements Dog { }
- C. class Man { private BestFriend dog; }
- D. class Man { private Dog bestFriend; }
- E. class Man { private Dog<bestFriend>; }
- F. class Man { private BestFriend<dog>; }

**Answer:** D

20. Given:

- 1. package test;
- 2.

```
3. class Target {  
4.     public String name = "hello";  
5. }
```

What can directly access and change the value of the variable name?

- A. any class
- B. only the Target class
- C. any class in the test package
- D. any class that extends Target

**Answer:** C