Oracle_RealExamQuestions.Com_1Z0-851_v2011-11-09_265q_By-Bomas

Exam A

QUESTION 1
Given:

```
public class Threads5 {
    public static void main (String[] args) {
        new Thread(new Runnable() {
            public void run() {
                System.out.print("bar");
            }}).start();
    }
}
```

What is the result?

A. Compilation fails.
B. An exception is thrown at runtime.
C. The code executes normally and prints "bar".
D. The code executes normally, but nothing prints.

Correct Answer: C

QUESTION 2
Given:

```
public class Threads4 {
    public static void main (String[] args) {
        new Threads4().go();
    }

    public void go() {
        Runnable r = new Runnable() {
            public void run() {
                System.out.print("foo");
            }
        };
        Thread t = new Thread(r);
        t.start();
        t.start();
    }
}
```

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.

Correct Answer: B

QUESTION 3
Given:

```
public class Threads2 implements Runnable {
    public void run() {
        System.out.println("run.");
        throw new RuntimeException("Problem");
    }

    public static void main(String[] args) {
        Thread t = new Thread(new Threads2());
```

```
        t.start();
        System.out.println("End of method.");
    }
}
```

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: Problem
   End of method.

Correct Answer: DE

QUESTION 4
Given:

```
public class TestOne implements Runnable {
    public static void main (String[] args) throws Exception {
        Thread t = new Thread(new TestOne());
        t.start();
        System.out.print("Started");
        t.join();
        System.out.print("Complete");
    }

    public void run() {
        for (int i = 0; i < 4; i++) {
            System.out.print(i);
        }
    }
}
```

What can be a result?

A. Compilation fails.
B. An exception is thrown at runtime.
C. The code executes and prints "StartedComplete".
D. The code executes and prints "StartedComplete0123".
E. The code executes and prints "Started0123Complete".

Correct Answer: E

QUESTION 5
Given:

```
Runnable r = new Runnable() {
    public void run() {
        System.out.print("Cat");
    }
};

Thread t = new Thread(r) {
    public void run() {
        System.out.print("Dog");
    }
```

```
};

t.start();
```

What is the result?

A. Cat
B. Dog
C. Compilation fails.
D. The code runs with no output.
E. An exception is thrown at runtime.

Correct Answer: B

QUESTION 6
Given:

```
1. public class Threads3 implements Runnable {
2.     public void run() {
3.         System.out.print("running");
4.     }
5.     public static void main(String[] args) {
6.         Thread t = new Thread(new Threads3());
7.         t.run();
8.         t.run();
9.         t.start();
10.    }
11. }
```

What is the result?

A. Compilation fails.
B. An exception is thrown at runtime.
C. The code executes and prints "running".
D. The code executes and prints "runningrunning".
E. The code executes and prints "runningrunningrunning".

Correct Answer: E

QUESTION 7
Given:

```
1. public class TestOne {
2.     public static void main (String[] args) throws Exception {
3.         Thread.sleep(3000);
4.         System.out.println("sleep");
5.     }
6. }
```

What is the result?

A. Compilation fails.
B. An exception is thrown at runtime.
C. The code executes normally and prints "sleep".
D. The code executes normally, but nothing is printed.

Correct Answer: C

QUESTION 8
Which two code fragments will execute the method doStuff() in a separate thread? (Choose two.)

A.
```
new Thread() {
    public void run() { doStuff(); }
};
```
B.
```
new Thread() {
    public void start() { doStuff(); }
};
```
C.
```
new Thread() {
    public void start() { doStuff(); }
}.run();
```
D.
```
new Thread() {
    public void run() { doStuff(); }
}.start();
```
E.
```
new Thread(new Runnable() {
    public void run() { doStuff(); }
}).run();
```
F.
```
new Thread(new Runnable() {
    public void run() { doStuff(); }
}).start();
```

Correct Answer: DF

QUESTION 9
Click the Exhibit button.

```
public class Threads1 {
    int x = 0;
    public class Runner implements Runnable {
        public void run(){
            int current = 0;
            for(int i = 0; i<4; i++){
                current = x;
                System.out.println(current + ", ");
                x = current + 2;
            }
        }
    }

    public static void main(String[] args) {
        new Threads1().go();
    }

    public void go(){
        Runnable r1 = new Runner();
        new Thread(r1).start();
        new Thread(r1).start();
    }
}
```

Which two are possible results? (Choose two.)

A. 0, 2, 4, 4, 6, 8, 10, 6,
B. 0, 2, 4, 6, 8, 10, 2, 4,
C. 0, 2, 4, 6, 8, 10, 12, 14,
D. 0, 0, 2, 2, 4, 4, 6, 6, 8, 8, 10, 10, 12, 12, 14, 14,
E. 0, 2, 4, 6, 8, 10, 12, 14, 0, 2, 4, 6, 8, 10, 12, 14,

Correct Answer: AC

QUESTION 10
Given:

```
1. public class TestFive {
2.     private int x;
3.
4.     public void foo() {
5.         int current = x;
6.         x = current + 1;
7.     }
8.
9.     public void go() {
10.        for(int i = 0; i < 5; i++) {
11.            new Thread() {
12.                public void run() {
13.                    foo();
14.                    System.out.print(x + ", ");
15.                }
16.            }.start();
17.        }
18.    }
19.}
```

Which two changes, taken together, would guarantee the output: 1, 2, 3, 4, 5, ? (Choose two.)

A. move the line 14 print statement into the foo() method
B. change line 9 to public synchronized void go() {
C. change the variable declaration on line 2 to private volatile int x;
D. wrap the code inside the foo() method with a synchronized(this) block
E. wrap the for loop code inside the go() method with a synchronized block synchronized(this) { // for loop code here }

Correct Answer: AD

QUESTION 11
Given:

```
class PingPong2 {
    synchronized void hit(long n) {
        for(int i = 1; i < 3; i++)
            System.out.print(n + "-" + i + " ");
    }
}

public class Tester implements Runnable {
    static PingPong2 pp2 = new PingPong2();

    public static void main(String[] args) {
        new Thread(new Tester()).start();
        new Thread(new Tester()).start();
    }

    public void run() { pp2.hit(Thread.currentThread().getId()); }
}
```

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

Correct Answer: B

QUESTION 12
Given:

```
void waitForSignal() {
    Object obj = new Object();
    synchronized (Thread.currentThread()) {
        obj.wait();
        obj.notify();
    }
}
```

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()".

Correct Answer: B

QUESTION 13
Which three will compile and run without exception? (Choose three.)

A.
```
private synchronized Object o;
```
B.
```
void go() {
  synchronized() { /* code here */ }
}
```
C.
```
public synchronized void go() { /* code here */ }
```
D.
```
private synchronized(this) void go() { /* code here */ }
```
E.
```
void go() {
  synchronized(Object.class) { /* code here */ }
}
```
F.
```
void go() {
  Object o = new Object();
  synchronized(o) { /* code here */ }
}
```

Correct Answer: CEF

QUESTION 14
Given:

```
public class PingPong implements Runnable {
    synchronized void hit(long n) {
        for (int i = 1; i < 3; i++)
            System.out.print(n + "-" + i + " ");
    }

    public static void main(String[] args) {
        new Thread(new PingPong()).start();
        new Thread(new PingPong()).start();
    }

    public void run() {
        hit(Thread.currentThread().getId());
    }
}
```

Which two statements are true? (Choose two.)

A. The output could be 8-1 7-2 8-2 7-1
B. The output could be 7-1 7-2 8-1 6-1
C. The output could be 8-1 7-1 7-2 8-2
D. The output could be 8-1 8-2 7-1 7-2

Correct Answer: CD

QUESTION 15
Click the Exhibit button.

```
class Computation extends Thread {
    private int num;
    private boolean isComplete;
    private int result;

    public Computation(int num){ this.num = num; }

    public synchronized void run() {
        result = num * 2;
        isComplete = true;
        notify();
    }

    public synchronized int getResult() {
        while ( ! isComplete ){
            try {
                wait();
            } catch (InterruptedException e) {
            }
        }
        return result;
    }

    public static void main(String[] args) {
        Computation[] computations = new Computation[4];
        for (int i = 0; i < computations.length; i++) {
            computations[i] = new Computation(i);
            computations[i].start();
        }
        for (Computation c : computations) {
            System.out.println(c.getResult() + " ");
        }
    }
}
```

What is the result?

A. The code will deadlock.
B. The code may run with no output.
C. An exception is thrown at runtime.
D. The code may run with output "0 6".
E. The code may run with output "2 0 6 4".
F. The code may run with output "0 2 4 6".

Correct Answer: F

QUESTION 16
Given:

```
public class TestSeven extends Thread {
    private static int x;
    public synchronized void doThings() {
        int current = x;
```

```
        current++;
        x = current;
    }
    public void run() {
        doThings();
    }
}
```

Which statement is true?

A. Compilation fails.
B. An exception is thrown at runtime.
C. Synchronizing the run() method would make the class thread-safe.
D. The data in variable "x" are protected from concurrent access problems.
E. Declaring the doThings() method as static would make the class thread-safe.
F. Wrapping the statements within doThings() in a synchronized(new Object()) { } block would make the class thread-safe.

Correct Answer: E

QUESTION 17
Given that Triangle implements Runnable, and:

```
    void go() throws Exception {
        Thread t = new Thread(new Triangle());
        t.start();
        for(int x = 1; x < 100000; x++) {
            //insert code here line 35
            if(x%100 == 0) System.out.print("g");
        }
   }

    public void run() {
        try {
            for(int x = 1; x < 100000; x++) {
                // insert the same code here line 41
                if(x%100 == 0) System.out.print("t");
            }
        } catch (Exception e) {

        }
    }
```

Which two statements, inserted independently at both lines 35 and 41, tend to allow both threads to temporarily pause and allow the other thread to execute? (Choose two.)

A. Thread.wait();
B. Thread.join();
C. Thread.yield();
D. Thread.sleep(1);
E. Thread.notify();

Correct Answer: CD

QUESTION 18
What is the output if the main() method is run?

```
public class Starter extends Thread {
    private int x = 2;

    public static void main(String[] args) throws Exception {
        new Starter().makeItSo();
```

```
    }

    public Starter(){
        x = 5;
        start();
    }

    public void makeItSo() throws Exception {
        join();
        x = x - 1;
        System.out.println(x);
    }

    public void run() { x *= 2; }
}
```

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.

Correct Answer: D

QUESTION 19
Given:

```
public class NamedCounter {
    private final String name;
    private int count;

    public NamedCounter(String name) {
        this.name = name;
    }

    public String getName() {
        return name;
    }

    public void increment() {
        count++;
    }

    public int getCount() {
        return count;
    }

    public void reset() {
        count = 0;
    }
}
```

Which three changes should be made to adapt this class to be used safely by multiple threads? (Choose three.)

A. declare reset() using the synchronized keyword
B. declare getName() using the synchronized keyword
C. declare getCount() using the synchronized keyword
D. declare the constructor using the synchronized keyword
E. declare increment() using the synchronized keyword

Correct Answer: ACE

QUESTION 20
Given:
foo and bar are public references available to many other threads. foo refers to a Thread and bar is an Object. The thread foo is currently executing bar.wait(). From another thread, what provides the most reliable way to ensure that foo will stop executing wait()?

A. `foo.notify();`
B. `bar.notify();`
C. `foo.notifyAll();`
D. `Thread.notify();`
E. `bar.notifyAll();`
F. `Object.notify();`

Correct Answer: E

QUESTION 21
Given that t1 is a reference to a live thread, which is true?

A. The Thread.sleep() method can take t1 as an argument.
B. The Object.notify() method can take t1 as an argument.
C. The Thread.yield() method can take t1 as an argument.
D. The Thread.setPriority() method can take t1 as an argument.
E. The Object.notify() method arbitrarily chooses which thread to notify.

Correct Answer: E

QUESTION 22
Which two statements are true? (Choose two.)

A. It is possible to synchronize static methods.
B. When a thread has yielded as a result of yield(), it releases its locks.
C. When a thread is sleeping as a result of sleep(), it releases its locks.
D. The Object.wait() method can be invoked only from a synchronized context.
E. The Thread.sleep() method can be invoked only from a synchronized context.
F. When the thread scheduler receives a notify() request, and notifies a thread, that thread immediately releases its lock.

Correct Answer: AD

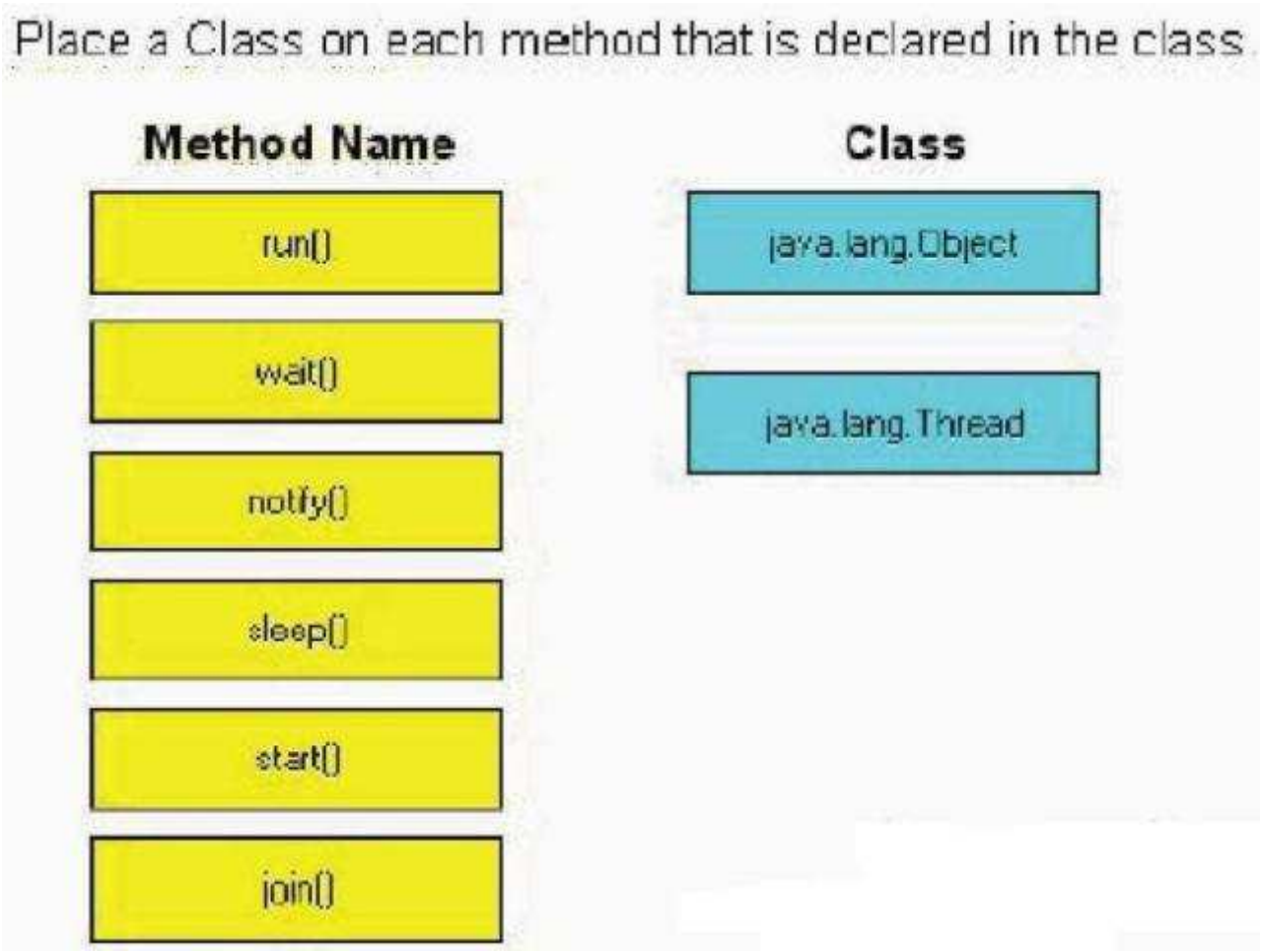
QUESTION 23
Which two statements are true? (Choose two.)

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().

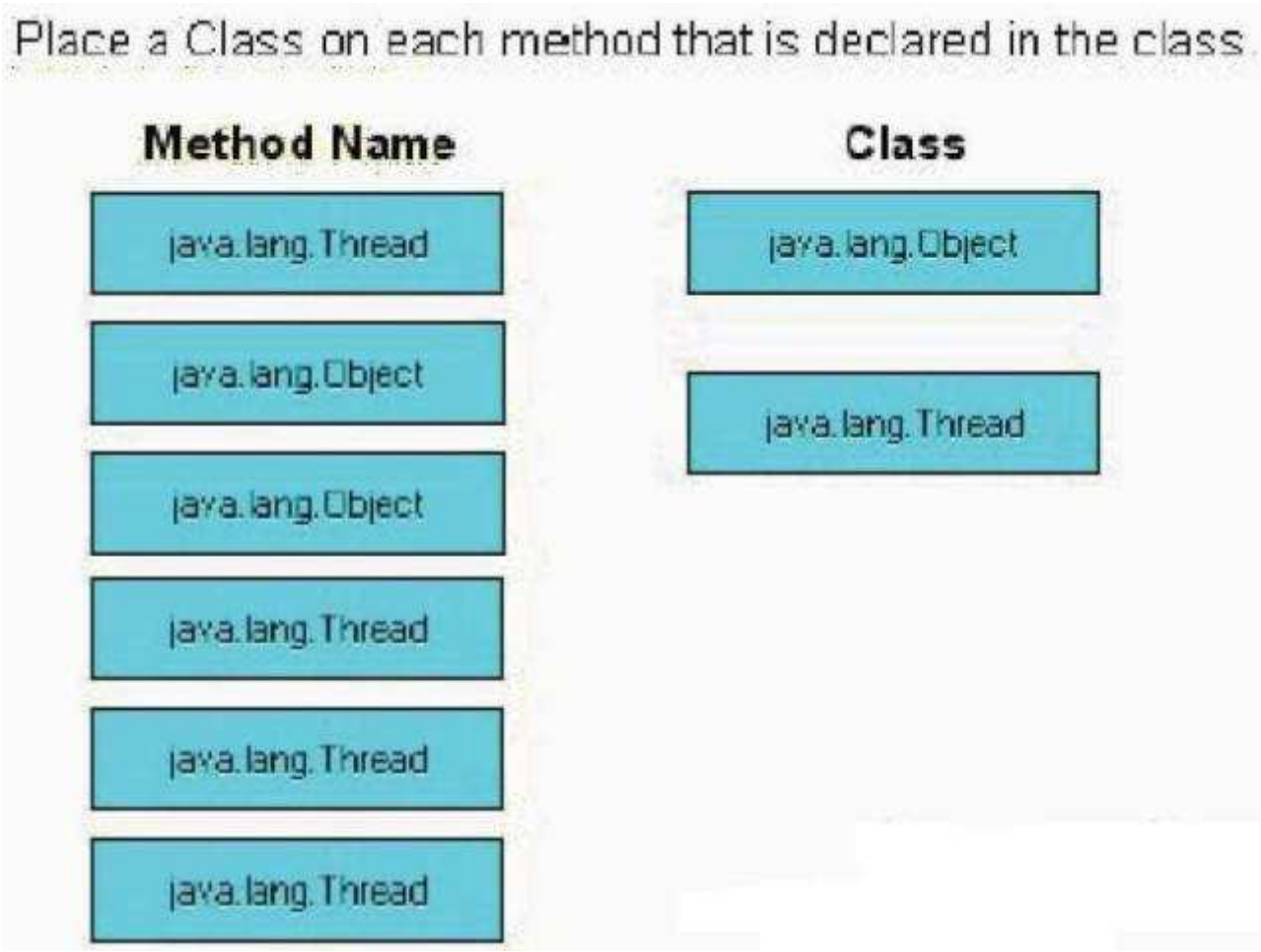
Correct Answer: AF

QUESTION 24

Select and Place:



Correct Answer:

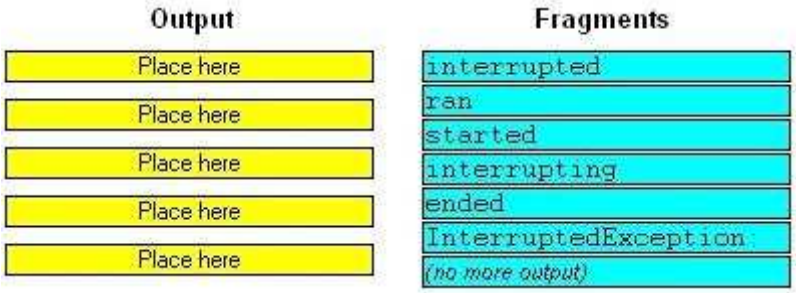


QUESTION 25
Given:

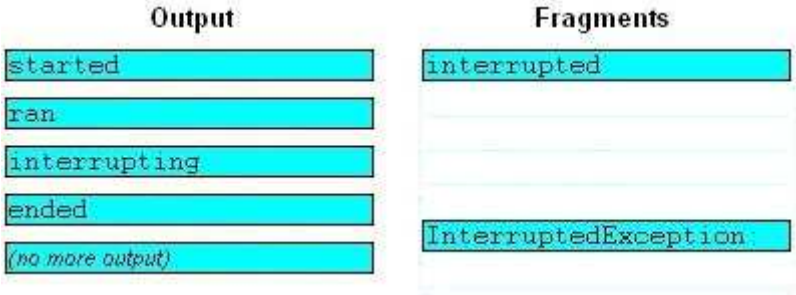
```
Runnable r = new Runnable() {
    public void run() {
        try {
            Thread.sleep(1000);
        } catch (interruptedException e) {
            System.out.println("interrupted");
        }
        System.out.println("ran");
    }
};
Thread t = new Thread(r);
t.start();
System.out.println("started");
t.sleep(2000);
System.out.println("interrupting");
t.interrupt();
System.out.println("ended");
```

Assue that sleep(n) executes in exactly n milliseconds, and all other code executes in an insignificant amount of time. Place the fragments in the output area to show the result of running this code.

Select and Place:



Correct Answer:



QUESTION 26

Select and Place:

Place the code elements into the class so that the code compiles and prints "Run. Run. dolt." in exactly that order. Note that there may be more than one correct solution.

```
public class TesTwo extends Thread {
 public static void main (String[] a) throws Exception {
    TesTwo t = new TesTwo();
    t.start();
    Place here
    Place here
  } Place here
  public void run() {
    System.out.print("Run. ");
  }
  public void doIt() {
    System.out.print("doIt. ");
  }
}
```

Code Elements

t.start();	t.join();	t.pause(10);	run();
t.run();	t.doIt();	doIt();	

Correct Answer:

Place the code elements into the class so that the code compiles and prints "Run. Run. dolt." in exactly that order. Note that there may be more than one correct solution.

```
public class TesTwo extends Thread {
 public static void main (String[] a) throws Exception {
    TesTwo t = new TesTwo();
    t.start();
    [ t.run(); ]
    [ t.join(); ]
  } [ t.doIt(); ]
  public void run() {
    System.out.print("Run. ");
  }
  public void doIt() {
    System.out.print("doIt. ");
  }
}
```

Code Elements

[t.start();] [t.pause(10);] [run();] [doIt();]

QUESTION 27
Place the code elements in positions so that the Flags2 class will compile and make appropriate use of the wait/notify mechanism. Note: You may reuse code elements.

Select and Place:

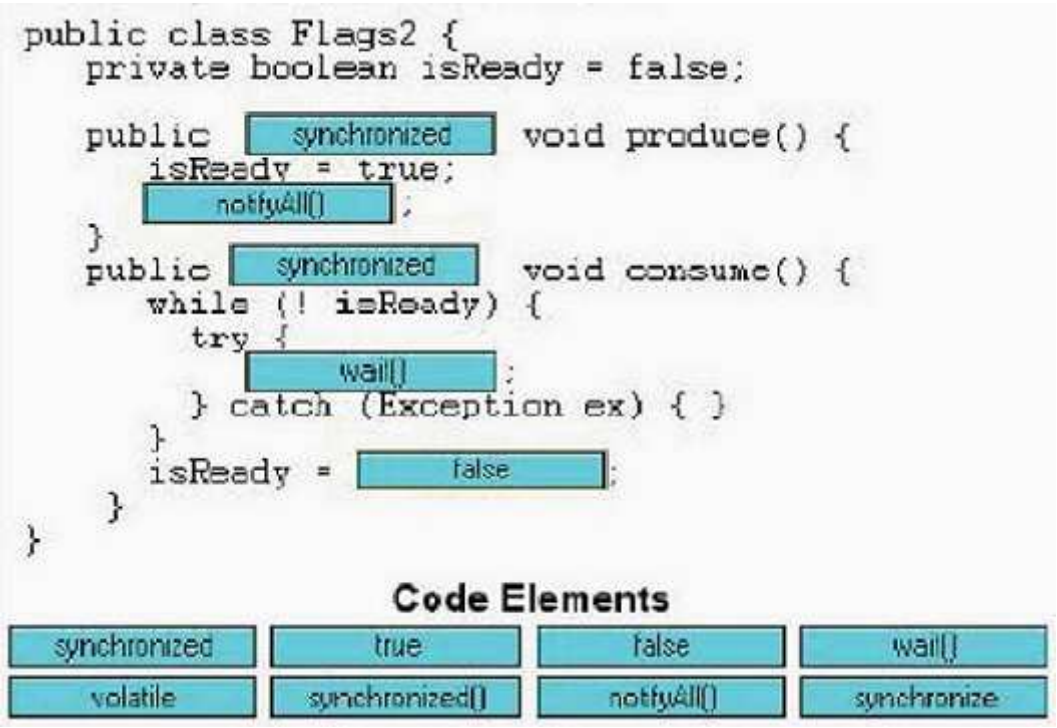
```
public class Flags2 {
   private boolean isReady = false;

   public [ Place here ] void produce() {
      isReady = true;
      [ Place here ];
   }
   public [ Place here ] void consume() {
      while (! isReady) {
        try {
           [ Place here ];
        } catch (Exception ex) { }
      }
      isReady = [ Place here ];
    }
}
```

Code Elements

synchronized	true	false	wait()
volatile	synchronized()	notifyAll()	synchronize

Correct Answer:


public class Flags2 {
private boolean isReady = false;
public synchronized void produce() {
isReady = true;
notifyAll() ;
}
public synchronized void consume() {
while (! isReady) {
try {
wait() ;
} catch (Exception ex) { }
}
isReady = false;
}
}
Code Elements
synchronized
true
false
wait()
volatile
synchronized()
notifyAll()
synchronize