1Z0-819 Q&As

Java SE 11 Developer

Pass Oracle 1Z0-819 Exam with 100% Guarantee

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

https://www.pass4itsure.com/1z0-819.html

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Oracle Official Exam Center

QUESTION 1

Given:

```
public class Tester {
   public static void main(String[] args) {
      char letter = 'b';
      int i = 0;
      switch(letter) {
         case 'a':
            i++;
            break;
         case 'b':
            i++;
         case 'c' | 'd': // line 1
            i++;
         case 'e':
            i++;
            break;
         case 'f':
            i++;
            break;
         default:
            System.out.print(letter);
      }
      System.out.println(i);
   }
}
```

What is the result?

A. b1

B. 2

C. b2

D. 1

E. b3

F. 3

G. The compilation fails due to an error in line 1.

Correct Answer: F

QUESTION 2

Your organization makes mlib.jar available to your cloud customers. While working on a new feature for mlib.jar, you see that the customer visible method public void enableService(String hostName, String portNumber) executes this code fragment

```
try {
   AccessController.doPrivileged((PrivilegedExceptionAction<Void>) () -> {
    transportSocket = new Socket(hostname, portNumber);
    return null;
   });
}
```

and you see this grant is in the security policy file:

```
grant codebase "file:${mlib.home}/j2se/home/mlib.jar" {
  permission java.io.SocketPermission "*", "connect";
};
```

What security vulnerability does this expose to your cloud customer\\'s code?

A. privilege escalation attack against the OS running the customer code

B. SQL injection attack against the specified host and port

C. XML injection attack against any mlib server

D. none because the customer code base must also be granted SocketPermission

E. denial of service attack against any reachable machine

Correct Answer: E

The correct answer is E. denial of service attack against any reachable machine. The code fragment shows that the enableService method uses the AccessController.doPrivileged method to create a new Socket with the specified hostname and portNumber. The security policy file grants the codebase permission to connect to any host using SocketPermission. This means that an attacker could potentially use this method to repeatedly create connections to any reachable machine, overwhelming its resources and causing a denial of service attack.

QUESTION 3

Given the code fragment:

var pool = Executors.newFixedThreadPool(5);

Future outcome = pool.submit(() > 1);

Which type of lambda expression is passed into submit()?

A. java.lang.Runnable

B. java.util.function.Predicate

C. java.util.function.Function

D. java.util.concurrent.Callable

Correct Answer: D

Reference: https://www.codota.com/code/java/methods/java.util.concurrent.Executors/newFixedThread Pool

QUESTION 4

Given:

```
public class Test {
    private int sum;
    public int compute() {
        int x = 0;
        while(x < 3) {
            sum += x++;
        }
        return sum;
    }
    public static void main(String[] args) {
        Test t = new Test();
        int sum = t.compute();
        sum = t.compute();
        t.compute();
        System.out.println(sum);
    }
}
```

What is the result?

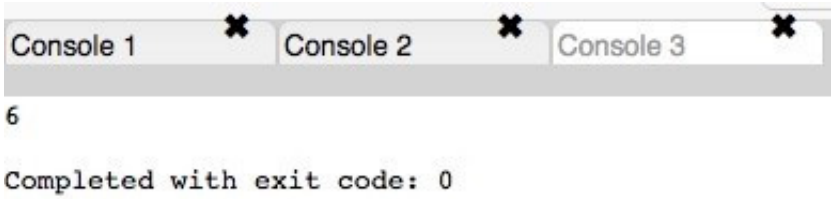
A. 9

B. An exception is thrown at runtime.

C. 3

D. 6

Correct Answer: D



QUESTION 5

Given:

```
public class Sports {
   ....
   public double getRatings() {
   ....
   }
  ....
}
```

and

```
public class Football extends Sports {
   ....
   public double getRatings() {
   ....
   }
  ....
}
```

Which is the correct implementation of the getRatings method in the Football subclass?

A. The subclass getRatings method uses public.getRatings() to call the base class method but uses its own named fields in the implementation.

B. The subclass getRatings method implementation directly accesses the fields in the Sports superclass.

C. The subclass getRatings method uses new.getRatings() to call the base class method but uses its own named fields in the implementation.

D. The subclass getRatings method uses super.getRatings() to call the base class method but uses its own named fields in the implementation.

Correct Answer: D