

Java Programming :2017

Test ka Aarambh

Which of the following statements is incorrect?

- ☐ public members of class can be accessed by any code in the program.
- ☒ private members of class can only be accessed by other members of the class.
- ☐ private members of class can be inherited by a sub class, and become protected members in sub class.
- ☐ protected members of a class can be inherited by a sub class in different packa

What will happen if we try to overload the main method in a class.

- ☐ Compile Time Error
- ☐ Runtime Error
- ☒ main method gets overloaded
- ☐ None Of these

which interface does the thread class implements ?

- ☒ Runnable
- ☐ Callable
- ☐ Serializable

☐ AutoClosable

☐ Other:

Which conditions should a method satisfy for overloading a method in a class. A.its return type should be same. B.number of parameters it takes should be different. C.type of parameters should be different

☐ Only A

☐ Both A and B

☒ Either B or C

☐ All of the above are true.

what will be the order of the output of the following code ?

```
public class MultiThread {  
    public static class RunnableImplementation implements Runnable {  
        private boolean number;  
        public RunnableImplementation(boolean number) {  
            this.number = number;  
        }  
    }  
}
```

```
    }  
  
    @Override  
    public void run() {  
        for (int i = 0; i <= 9; i++) {  
            if (number) {  
                System.out.println(i);  
            } else {  
                System.out.println((char) (i + 'A'));  
            }  
            try {  
                Thread.sleep(100);  
            } catch (InterruptedException e) {  
                e.printStackTrace();  
            }  
        }  
    }  
}  
  
public static void main(String[] args) throws InterruptedException, ExecutionException {  
    new Thread(new RunnableImplementation(true)).start();  
    new Thread(new RunnableImplementation(false)).start();  
}
```

- ☐ All Numbers will be printed first then all alphabets
- ☐ All Alphabets will be printed first then all numbers
- ☒ Random order
- ☐ First 1 number will be printed and then 1 alphabet and so on, till all numbers and alphabet are printed
- ☐ Other:

Output of following program

```
public static void main(String[] args) {  
    try{  
        System.out.println("hi");  
        System.exit(0);  
    } finally {  
        System.out.println("bye");  
    }  
}
```

}

- ☒ hi
- ☐ bye
- ☐ hi bye
- ☐ no output

Static keyword is used to make a variable Thread-Safe?

- ☐ True
- ☒ False

Compile time polymorphism is achieved through

- ☒ Method Overriding
- ☐ Method Overloading
- ☐ Method Rewriting
- ☐ None of these

Which statement is valid

- ☐ CheckedException are used to indicate programmatic errors
- ☐ CheckedException are check at compile time and hence declare the exception using throws keyword
- ☒ UncheckedException extends Error class

Will this code Execute

```
public final class SuperClass {  
    public void print() {  
        System.out.println("print");  
    }  
}
```



```
public static void main(String[] args) {  
    new SubClass().print();  
}  
  
class SubClass extends SuperClass {  
  
}
```

- ☐ Compile Time Error
- ☐ Run Time Error
- ☒ Successfully execute

What will be the output of the code ?

```
public static void main(String[] args) {  
  
    String a = "hello";  
    String b = "hello";  
    String c = new String("World");  
    String d = new String("World");  
}
```

```
String d = new String("Two");  
if (a == b) {  
    System.out.print("One");  
}  
if (c == d) {  
    System.out.print("Two");  
}  
}
```

- ☐ One
- ☐ Two
- ☐ OneTwo
- ☒ None of the Above

Significance of Volatile keyword in java

- ☒ If a field is declared volatile, in that case the Java memory model ensures that all threads see a consistent value for the variable.
- ☐ Volatile caches methods
- ☐ All of these
- ☐ None

Mark output of the code

```
public class Constructor1 {  
    // no constructor  
  
    public static void main(String[] args) {  
        new Const2();  
    }  
}
```

```
}  
  
class Const2 extends Constructor1 {  
    public Const2() {  
        super(); // will this throw error ?  
        System.out.println(1);  
    }  
}
```

- ☐ compile time error
- ☐ runtime error
- ☒ 1

Output of program

```
public class Indecisive {  
    public static void main(String[ ] args  
    ) {  
        System.out.println(decision( ) );  
    }  
  
    static boolean decision( ) {  
        try {  
            return true;  
        } finally {  
            return false;  
        }  
    }  
}
```

- ☐ true
- ☒ false

Mark output of the code

```
public class Constructor1 {  
    // no constructor  
  
    public Constructor1(String x) {  
        System.out.println(x);  
    }  
  
    public static void main(String[] args) {
```

```
        new Const2();  
        new Const2("string");  
    }  
}  
  
class Const2 extends Constructor1 {  
    public Const2(String x) {  
        super(x);  
    }  
  
    public Const2() {  
        System.out.println(1);  
    }  
}
```

- ☐ 1 string
- ☐ string 1
- ☒ error
- ☐ Option 4

How many threads does the below given code opens ?

```
public class MultiThread {  
  
    public static class RunnableImplementation implements Runnable {  
  
        @Override  
        public void run() {  
            System.out.println("It works !!");  
        }  
    }  
}
```



```
}  
  
public static void main(String[] args) throws InterruptedException, ExecutionException {  
    System.out.println("start");  
    ExecutorService executorService = Executors.newFixedThreadPool(10);  
    for (int i = 0; i < 100; i++) {  
        executorService.submit(new RunnableImplementation());  
    }  
    System.out.println("Apparent end");  
}
```

- ☒ 100
- ☐ 10
- ☐ 11
- ☐ Cannot be determined
- ☐ Other:

What will be the output of the following program ?

```
class Test{  
  
    public int test(int a){  
        System.out.println("test1");  
        return a;  
    }  
}
```

```
}  
  
public float test(int a){  
    System.out.println("test2");  
    return b;  
}  
  
public static void main(String args[]){ |  
    Test obj = new Test();  
    obj.test(2);  
}  
}
```

- ☒ Compile Time Error
- ☐ test1
- ☐ test2
- ☐ test1 test2

Runnable defines a method run. What's the signature?

- ☒ public void run()
- ☐ public void run() throws InterruptedException
- ☐ public void run(Runnable r)
- ☐ public void run(Runnable r) throws InterruptedException
- ☐ Other:

Can we overload static methods in java ?

- ☐ Yes
- ☒ No

The client in socket programming must know which information?

- ☐ IPaddress of Server
- ☐ Port number
- ☒ Both A & B
- ☐ None of the above
- ☐ Other:

Will this code Execute

```
public final class SuperClass {  
    static final int var = 10;  
  
    public static void main(String[] args) {  
        var = 1000;  
    }  
}
```

- ☐ Yes
- ☒ Compile Time Error
- ☐ Run Time Error

Can Constructor in Java be STATIC

- ☒ Compile Error
- ☐ yes
- ☐ Runtime Error

☐ Other:

Which of these is correct way of inheriting class A by class?

- ☐ class B + class A {}
- ☐ class B inherits class A {}
- ☒ class B extends A {}
- ☐ class B extends class A {}

What is the name of the method used to start a thread execution?

- ☐ init();
- ☒ start();
- ☐ run();
- ☐ resume();

Thread synchronization is required because

- ☐ all threads of a process share the same address space
- ☒ all threads of a process share the same global variables
- ☐ all of the mentioned

Output following code

```
public static void main(String[] args) {  
    System.out.println(test());  
}  
static int test () {  
    int a = 0;
```

```
try{
    return a;
} finally {
    ++a;
}
```

- ☒ 0
- ☐ 1
- ☐ 2
- ☐ None of above

Mark output of Code

```
class Bike{
    int speedlimit=90;
}

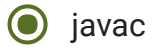
class Honda3 extends Bike{
    int speedlimit=150;

    public static void main(String args[]){
        Bike obj=new Honda3();
        System.out.println(obj.speedlimit);
    }
}
```

- ☐ compile time error
- ☒ 90
- ☐ 150
- ☐ Option 4

Which is not part of jvm

- ☐ JIT
- ☐ GC
- ☐ Classloader



☒ Send me a copy of my responses.

I'm not a robot

reCAPTCHA
[Privacy](#) - [Terms](#)

Page 2 of 2

BACK

SUBMIT

Never submit passwords through Google Forms.

This form was created inside Media.Net. [Report Abuse](#) - [Terms of Service](#) - [Additional Terms](#)

Google Forms