Question :
If I have 2 try blocks and there corresponding catch blocks, and there is an exception catched
in the first try block, will the second try block be executed?

```
try{
            //risky code
}catch(Exception e){
            //handling code
}

try{
      //risky code
}catch(Exception e){
      //handling code
}
```

working of throws is not clear
   throws => to work with checked exception
   throw => to work with unchecked exception

to handle the checked exceptions we use throws keyword????
    throws => this keyword is used in realtime coding

Sir how is the exception handeled when we use ducking?
  => excpetion is ducked then the exception object reaches to DefaultExceptionhandler
     DefaultException handler uses printStackTrace() to handle it.

if we are use ducking for compile time error, does it cause any problem for run time
exception handles? I mean does that key word Duck the User Exception during run time?
    exception => occurs in runtime and it will be ducked.

int c = 100/0;   is it a runtime or compile time error as value for constants are assigned at compile time ???
  it is runtime error as the execution happens at the runtime.

can we use different try blocks and one catch block

```
    try{

     }//compile time error
    try{

    }
    catch(Exception e){

    }
```

sir can u pls repeat compile time and runtime exception briefly
   CompileTime => no exception(just compiler will see what code would create the problem at
                                                                     runtime)
   RuntimeExeption => problem occured at the runtime due to some unwanted disturbance.

which is recommended try catch or throw throws

```
    try{}catch(XXXX e) => it is used for uncheckedExcpetion
    throws -> checked Exception
    throw  -> handle the exception and to throw the exception manually to the
caller.


          Student s[] = new Student[3];
             Scanner sc = new Scanner(System.in);
           for(int i=0; i<s.length; i++) {
                  int age= sc.nextInt();
                  float salary =  sc.nextDouble(),
                  String name = sc.nextLine()
                  s[i] = new Student(age,salary,name);
           }

           for(int i=0; i<s.length; i++) {
             System.out.println(s[i].getName() + " " + s[i].getCpi() + " " +
s[i].getId());
           }

      code works fine


ducking => user is not intereseted in handling the exception,so jvm automatically
sends
           the exception object to the caller.

but here we have just ducking it by throws keyword but still exception not occured
    Thread.sleep(5000);//here exception wont occur still why compiler warned ?
        Compiler will support our code to make sure at the run time problem should
not come for jvm exeution.

Sir Could you please explain throws keyword.. how we are handling exception there?

//Vijet wrote the code
class Demo{
            public void m1()throws Exception {
                Thread.sleep(5000);
       }
 }
 public class Sample{
            public static void main(String[] args) throws Exception{

                  try{
                        new Demo().m1();
                  }catch(Exception e){
                        //handling logic
                  }
                        or

                  new Demo().m1();
            }
   }

JVM -> deafaultExceptionhandler ->handle it by using printStackTrace().


interface Calculator
{
      public float add(int a, int b);
}

 class launch4Calculator {
      public static void main(String[] args) {
            Scanner scan = new Scanner(System.in);
            int x = scan.nextInt();
            int y = scan.nextInt();

            Calculator cal = (a,b) -> a+b;
            System.out.println(cal.add(x, y));
            cal.add(x,y);
      }

}
```

why a and b creates error?

I didn't got exact diff between aggregation and compostion can you please explain?

```
   Composition => HAS-A realtionship
                              (container and contained object)

   Aggregation => HAS-A relationsihp
                              (container and contained object)
```

difference between error and exception

```
   Error => Problem occured in a program at runtime,but not able to handle the
problem through
               language specification.
                              eg: int[] a= new int[99999999];
                                    output: OutOfMemoryError.
                  here the problem is with the RAMSpace,we can't write a handling
code in catch block
                  to increase the RAM Space so it is called "Error".

   Exception => problem occured in program at runtime and programmatically we can
handle the problem through
                        language specification(Exception handling).
                                    syntax: try{} catch(XXXX e){}
```

```
//JDK code(sunmicorsystem team code)
public final class String{
                  public String subString(int x,int y)throws
StringIndexOutBoundsExeption{
                              //logic
                  }
}

public class Demo{
  main() throws SIOBE{
      new String("dhoni").substring(0,3);//dho
      new String("sachin").substring(-1,-100);//SIOBE
            ;;;;
            ;;;;
            ;;;;
            ;;;;
            ;;;;
  }
}

sir how to read exception message like exception(e){
                                                             system.out.print(e) or
System.out.println(e.getString());
                                                             e.printStackTrace();


      System.out.println(e.getMessage());
                                                       }

try{
   int res=10/0;//risky code
 }catch(ArithmeticExeption ae){
      //handling exception
      int res=10/2;
 }
```