



1. Create a class named MyException that extends Exception and contains:
 - a) a constructor that has a String parameter whose value will be returned by the getMessage() method belonging to this exception; getMessage() is not implemented again in the extended class (do not use overriding).
 - b) a method that returns for how many times MyException or any of its derived classes have been instantiated.

Create into a main method three instances of MyException that will be referred by the same reference. Call for each object at least two services.



2. What is going to be printed after the execution of the program given below?

```
class L1 extends Exception {
    public String toString() { return "L1"; }
}

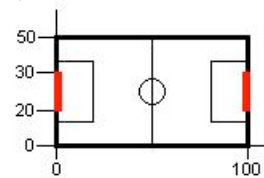
class L2 extends Exception {
    public String toString() { return "L2"; }
}

class Test {
    public static void main(String[] args) {
        try {
            for (int i=0; i<4; i++) {
                if(i==0) throw new L1();
                else throw new L2();
            } catch (Exception e) { System.out.println(e); }
        }
    }
}
```



3. Create a program for modeling a simplified soccer game. The soccer field has the dimensions described in the figure.

The program has a Ball class that has two attributes for recording the current position of the ball (x and y). These coordinates are set via the constructor of the class. The class has two methods for returning these positions. A method named play can be called for a Ball object and it generates new values for the position of the ball (x and y) and sometimes it will end its execution by throwing different checked exceptions, as it follows.



if the current position of the ball (X, Y) satisfies $Y=0$ OR $Y=50$, an Out exception will be raised.
if the current position of the ball (X, Y) satisfies $(X=0$ OR $X=100)$ AND $(Y \geq 20$ AND $Y \leq 30)$, a Goal exception will be raised.
if the current position of the ball (X, Y) satisfies $(X=0$ OR $X=100)$ AND there is no out or goal $(0 < Y < 20$ OR $30 < Y < 50)$, a Corner exception will be raised.



The program contains also a Game class that contains the names of two teams (two String attributes) that are set when an object of this type is instantiated. The class has attributes for recording the score of the game, and the total number of the outs and corners that were thrown in the current game. The class has a method for printing the values of all the stored attributes within. The game is simulated as it follows:

a ball is created

the play method from the Ball class is called for 2000 times

for each position a message that contains the value of the two coordinates is printed

for each of the special situation that may appear, the game attributes are modified accordingly, a suitable message revealing the special situation is printed on the screen and a new ball is created

for a goal, a new ball having $(X=50, Y=25)$ will be created

for an out, a new ball having the previous coordinates will be created

for a corner, the new ball will be placed in the appropriate corner of the field.

The coordinates will be randomly generated using the method

public int nextInt(int n) from java.util.Random.