

PROGRAMACION JAVA. Ejercicios Tema 10. Excepciones.

1. ¿Qué se muestra por pantalla al ejecutar el siguiente programa?

```
public class MiExcepcion extends Exception{
    public MiExcepcion() {
    }
    public MiExcepcion(String mensaje) {
        super(mensaje);
    }
}

public class Main {
    public static void main(String[] args) {
        int cont = -1;
        while (true) {
            try {
                cont++;
                if (cont % 2 == 0) {
                    throw new MiExcepcion();
                }
                System.out.println(cont);
            } catch (MiExcepcion e) {
                System.out.println("Excepcion capturada");
            } finally {
                System.out.println("finally " + cont);
                if (cont == 3) {
                    break;
                }
            }
        }
    }
}
```

2. ¿Qué se muestra por pantalla al ejecutar el siguiente programa?

```
public class MiExcepcion extends Exception{
    public MiExcepcion() {
    }
    public MiExcepcion(String mensaje) {
        super(mensaje);
    }
}

public static void main(String[] args) {
    int cont = -1;
    while (true) {
        try {
            cont++;
            if(cont == 0){
                throw new MiExcepcion("Contador 0");
            }
            if (cont % 2 == 0) {
                throw new MiExcepcion();
            }
            System.out.println(cont);
        } catch (MiExcepcion e) {
            System.out.println("Excepcion capturada " + e.getMessage());
        } finally {
            System.out.println("finally " + cont);
            if (cont == 3) {
                break;
            }
        }
    }
}
```

3. Si en el ejercicio anterior cambiamos dentro del `catch` `e.getMessage()` por `e.toString()` ¿Qué muestra ahora?
4. Escribe un programa que cree un array bidimensional irregular de enteros. El array tendrá 100 filas. Para cada fila se pedirá su tamaño y se asignará la memoria correspondiente. Si no es posible reservar memoria para todas las filas el programa mostrará un mensaje indicando el número de filas que ha podido asignar.

Las excepciones a capturar serán las siguientes:

- Si el tamaño del array (en este caso el tamaño de la fila de la matriz) es negativo, el sistema lanza la excepción `NegativeArraySizeException`.
 - Cuando no hay más memoria disponible el sistema lanza la excepción `OutOfMemoryError`.
 - Capturar la `InputMismatchException` que se lanza cuando se intenta leer el tamaño de la fila de la matriz y se introduce un valor que no es un entero.
5. Crea una excepción propia llamada *NumeroNegativoException*. Contendrá solamente un constructor sin parámetros que invocará al constructor de *Exception* enviándole el mensaje "Debe introducir un número positivo". Utiliza esta excepción de la siguiente forma: escribe un método que recibe una variable de tipo `double` y devuelve su raíz cuadrada. Este método lanzará la excepción si el número que recibe es negativo. La excepción se captura en el `main`.
 6. Modifica el programa anterior para que la excepción sea capturada dentro del propio método.
 7. Modifica el programa anterior para que la excepción después de ser capturada por el método se relance al método `main`.
 8. Explica qué se muestra por pantalla al ejecutar este código en cada uno de los casos:
 - a) `metodo1()` no lanza ninguna excepción.
 - b) `metodo1()` lanza una excepción de tipo `ArithmeticException`
 - c) `metodo1()` lanza una excepción de tipo `InputMismatchException`
 - d) `metodo1()` lanza una excepción de tipo `NumberFormatException`

```
public static void main(String[] args) {
    try{
        System.out.println("Método main antes de llamar a metodo()");
        metodo();
        System.out.println("Método main después de llamar a metodo()");
    }catch(InputMismatchException e){
        System.out.println("InputMismatchException capturada en el main");
    }
}

public static void metodo(){
    try{
        System.out.println("Antes de llamar a metodo1");
        metodo1();
        System.out.println("Después de llamar a metodo1");
    }catch(ArithmeticException e){
        System.out.println("Método: cazada ArithmeticException");
    }
}

public static void metodo1(){
    //a)    metodo1() no lanza ninguna excepción.
    //b)    metodo1() lanza una excepción de tipo ArithmeticException
    //c)    metodo1() lanza una excepción de tipo InputMismatchException
    //d)    metodo1() lanza una excepción de tipo NumberFormatException
}
```