

**APUTNS T06 - T07**

# Constructor

```
public Asignatura(int id) {
    this.id = id;
}
```

# Getter y setter

```
private double calificacion;
```

```
public double getCalificacion() { #OJO:Devolver variable static = método static
    return calificacion;
}
```

```
public void setCalificacion(double calificacion) {
    this.calificacion = calificacion;
}
```

# Override toString()

@Override

```
public String toString() {
    return String.format("Alumno → %d", idAlumno + 1);
}
```

# For each

```
private Alumno[] grupo;
```

```
for (Alumno alumno : this.grupo) {
    if (alumno.getVotos() ≥ masVotos) {
        masVotos = alumno.getVotos();
    }
}
```

## NOTA: Código después de throw, si se lanza NO SE EJECUTA, es como return

# Excepción TRATADA, NO PROPAGA

```
static void dividirEntreArray(int entero, int[] array) {
    for (int i : array) {
        try {
            if (i == 0 || entero == 0) throw new Exception("ERROR:Entre 0");
        }
        catch (Exception e) {
            System.out.println(e.getMessage());
        }
    }
}
```

# Excepción NO TRATADA, PROPAGA al main() o otros métodos

```
public static int getEntero(int num) throws Exception {
    try { if (num % 2 == 0) throw new Exception("Lanzando excepcion rana");}
    catch (IOException io) {
        System.out.println(io.getMessage());
        return 2;
    }
}
```

### # Excepción personalizada

```

public class AgotadoException extends Exception {

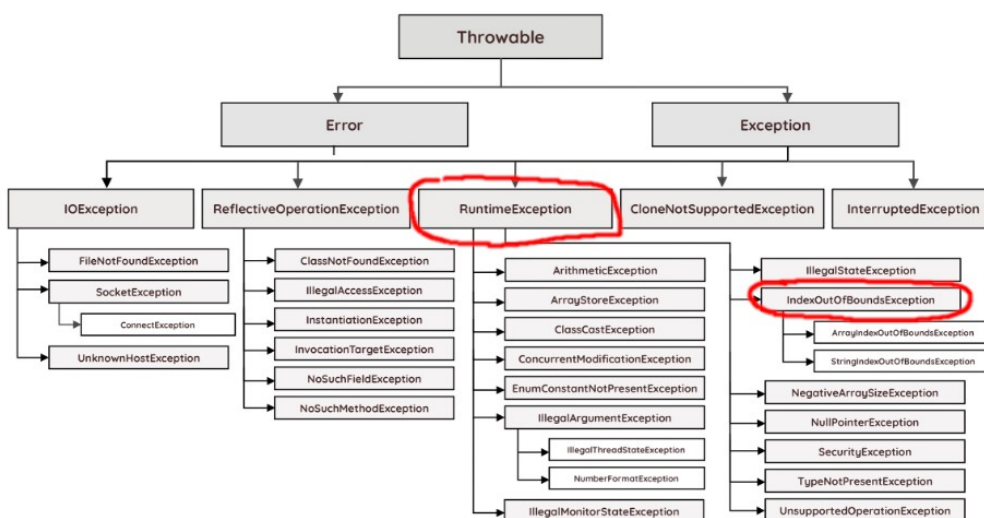
    public AgotadoException(String message) {
        super(message);
    }
}

public class Corredor {

    public void correr() throws AgotadoException {
        if (energia < 10) throw new AgotadoException("Sin energia");
    }
}

```

### # TIPOS DE EXPEPCIONES



#### ArithmeticException

It is thrown when an exceptional condition has occurred in an arithmetic operation.

#### ArrayIndexOutOfBoundsException

It is thrown to indicate that an array has been accessed with an illegal index. The index is either negative or greater than or equal to the size of the array.

#### FileNotFoundException

This Exception is raised when a file is not accessible or does not open.

#### IOException

It is thrown when an input-output operation failed or interrupted

#### NullPointerException

This exception is raised when referring to the members of a null object. Null represents nothing

#### NumberFormatException

This exception is raised when a method could not convert a string into a numeric format.

#### RuntimeException

This represents any exception which occurs during runtime.

#### StringIndexOutOfBoundsException

It is thrown by String class methods to indicate that an index is either negative or greater than the size of the string