

# Planificación y realización de pruebas

## Test-Driven Development (TDD)

Esta práctica ha sido desarrollada en su totalidad por Román Ginés Martínez Ferrández (rgmf@riseup.net) salvo referencias al pie de página.

Todas las imágenes y todos los recursos utilizados son de Dominio Público a menos que se diga lo contrario.



Creative Commons Reconocimiento – NoComercial - CompartirIgual  
CC by-nc-sa

## Introducción

Test-Driven Development o TDD es una práctica de programación en la que se comienza escribiendo las pruebas en primer lugar. Cuando se ejecutan estas pruebas la primera vez se obtendrán errores porque estas no pasarán. Entonces se desarrolla aquel componente que provoca el error, es decir, que provoca que estas pruebas no pasen.

Con esta metodología acabas escribiendo el programa en el proceso de conseguir que dichas pruebas pasen.

Te recomiendo el libro *Obey the Testing Goat! TDD for the Web, with Python, Selenium, Django, JavaScript and pals...* que puedes encontrar en [esta web](#) Con licencia Creative Commons.

## Práctica

Usando la metodología TDD tienes que desarrollar una calculadora en Java que realice las operaciones básicas:

- suma de números enteros,
- resta de números enteros,
- multiplicación de números enteros, y
- división de números enteros.

Así, habrá una clase llamada Calculator con cuatro métodos que realicen las operaciones indicadas arriba. Cada uno de estos métodos devolverá un número entero con el resultado de la operación y recibe dos argumentos: el operando 1 y el operando 2.

En el caso de la división el operando 2 no podrá ser 0. En el caso en que se indique un 0 en el operando 2 entonces se tendrá que lanzar la excepción **ArithmeticException** con el mensaje **"No se puede dividir entre 0"**. Además, la división devolverá un **double** en vez de un **int**.

Esta es la signatura de todos los métodos:

```
public int add(int a, int b)
public int subtract(int a, int b)
public int multiply(int a, int b)
public double divide(int a, int b)
```

## Casos de prueba para el método divide

Entradas		Salida
A	B	
10	5	2d
4	3	1.33
-4	3	-1.33
4	-3	-1.33
0	4	0d
4	0	ArithmeticException con mensaje: "No se puede dividir entre 0"