

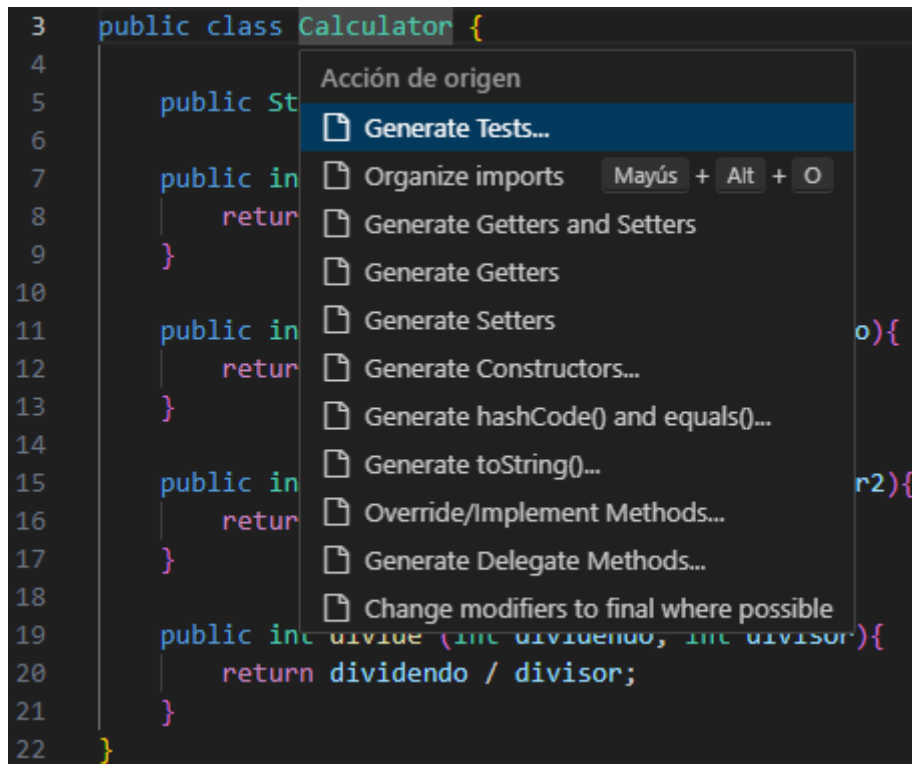
# Ejercicio 1: Calculadora Simple

Crea una clase Calculator que tenga métodos para sumar, restar, multiplicar y dividir dos números. Luego, escribe pruebas unitarias para cada uno de estos métodos para verificar su funcionamiento.

En primer lugar, creo la clase “Calculator”, con todas sus funciones:

```
public class Calculator {  
  
    public String nombre = "";  
  
    public int suma(int sumando1, int sumando2){  
        return sumando1 + sumando2;  
    }  
  
    public int resta (int minuendo, int sustraendo){  
        return minuendo - sustraendo;  
    }  
  
    public int multiplica (int factor1, int factor2){  
        return factor1 * factor2;  
    }  
  
    public int divide (int dividendo, int divisor){  
        return dividendo / divisor;  
    }  
}
```

Ahora, hacemos click derecho sobre la clase, pulsamos “Acción de código fuente” > “Generar test”.



Una vez hayamos generado el archivo para test, haremos las funciones que realizarán los test.

```
public void testSuma_2_y_3() {  
    //Arrange  
    int sumando1 = 2;  
    int sumando2 = 3;  
    int sumaTotal = 5;  
    Calculator calculator = new Calculator();  
  
    //Act  
    int resultadoReal = calculator.suma(sumando1, sumando2);  
  
    //Assert  
    Assert.assertEquals(sumaTotal, resultadoReal);  
}
```

Ahora veremos que significan este código por partes.

En la parte de “Arrange” organizaremos lo que vamos a hacer. Declaramos variables y creamos instancias.

```
//Arrange
    int sumando1 = 2;
    int sumando2 = 3;
    int sumaTotal = 5;
    Calculator calculator = new Calculator();
```

En la parte de “Act” tomaremos las acciones necesarias para tomar hacer el test adecuadamente.

```
//Act
    int resultadoReal = calculator.suma(sumando1, sumando2);
```

Por último, en la parte de “Assert”, verificaremos los resultados.

```
//Assert
    Assert.assertEquals(sumaTotal, resultadoReal);
```

Para acabar, podemos realizar al menos un par de pruebas por cada función con distintos valores para tener mayor certeza de un buen funcionamiento.

```
import org.junit.Assert;

import org.junit.Test;

public class CalculatorTest {
    @Test
    public void testSuma_2_y_3() {
        //Arrange
        int sumando1 = 2;
        int sumando2 = 3;
        int sumaTotal = 5;
        Calculator calculator = new Calculator();

        //Act
        int resultadoReal = calculator.suma(sumando1, sumando2);

        //Assert
        Assert.assertEquals(sumaTotal, resultadoReal);
    }
}
```

```
@Test
public void testSuma_12_y_menos7() {
    //Arrange
    int sumando1 = 12;
    int sumando2 = -7;
    int sumaTotal = 5;
    Calculator calculator = new Calculator();

    //Act
    int resultadoReal = calculator.suma(sumando1, sumando2);

    //Assert
    Assert.assertEquals(sumaTotal, resultadoReal);
}
```

```
@Test
public void testResta_12_y_5() {
    //Arrange
    int minuendo = 12;
    int sustraendo = 5;
    int diferencia = 7;

    //Act
    Calculator calculator = new Calculator();
    int resultadoReal = calculator.resta(minuendo, sustraendo);

    //Assert
    Assert.assertEquals(diferencia, resultadoReal);
}
```

```
@Test
public void testResta_7_y_menos7() {
    //Arrange
    int minuendo = 7;
    int sustraendo = -7;
    int diferencia = 14;

    //Act
    Calculator calculator = new Calculator();
    int resultadoReal = calculator.resta(minuendo, sustraendo);
}
```

```
        //Assert
        Assert.assertEquals(diferencia, resultadoReal);
    }

    @Test
    public void testMultiplicacion_7_y_9() {
        //Arrange
        int factor1 = 7;
        int factor2 = 9;
        int producto = 63;

        //Act
        Calculator calculator = new Calculator();
        int resultadoReal = calculator.multiplica(factor1, factor2);

        //Assert
        Assert.assertEquals(producto, resultadoReal);
    }

    @Test
    public void testMultiplicacion_5_y_2() {
        //Arrange
        int factor1 = 5;
        int factor2 = 2;
        int producto = 10;

        //Act
        Calculator calculator = new Calculator();
        int resultadoReal = calculator.multiplica(factor1, factor2);

        //Assert
        Assert.assertEquals(producto, resultadoReal);
    }

    @Test
    public void testDivision_27_y_3() {
        //Arrange
        int dividendo = 27;
        int divisor = 3;
        int cociente = 9;

        //Act
```

```
Calculator calculator = new Calculator();
int resultadoReal = calculator.divide(dividendo, divisor);

//Assert
Assert.assertEquals(cociente, resultadoReal);
}

@Test
public void testDivision_84_y_7() {
    //Arrange
    int dividendo = 84;
    int divisor = 7;
    int cociente = 12;

    //Act
    Calculator calculator = new Calculator();
    int resultadoReal = calculator.divide(dividendo, divisor);

    //Assert
    Assert.assertEquals(cociente, resultadoReal);
}
}
```