

Pruebas con JUnit – Ejercicio 2

1.-Métodos Test

En la clase Calculadora.

Modifica el método resta() y añade los métodos resta2() y divide2()

Crea después los test para probar los tres métodos. Utiliza los métodos assertTrue(), assertFalse(), assertNull(), assertNotNull() o assertEquals() según convenga.

Los métodos son:

```
public int resta() {
    int result;
    if (resta2()) {
        result = num1 - num2;
    } else {
        result = num2 - num1;
    }
    return result;
}

public boolean resta2() {
    if (num1 > num2) {
        return true;
    } else {
        return false;
    }
}

public String divide2() {
    if (num2 == 0) {
        return null;
    }
    int result = num1 / num2;
    return String.valueOf(result);
}
```

2.- Fecha

Desarrolla una batería de pruebas para probar el método DevuelveFecha() de la clase Fecha que se expone a continuación. El método recibe un número entero y devuelve un String con un formato de fecha que dependerá del valor de dicho número. Si el número recibido es distinto de 1,2 o 3 el método devuelve ERROR. La clase es la siguiente:

```
import java.util.Date;
import java.text.SimpleDateFormat;

public class Fecha {
    SimpleDateFormat formato;
    Date hoy;

    public Fecha(){
        hoy=new Date();
    }

    public String DevuelveFecha(int tipo){
        String cad="";
        switch (tipo) {
            case 1: {
                formato=new SimpleDateFormat("yyyy/MM");
                cad=formato.format(hoy);
                break;
            }
            case 2: {
                formato=new SimpleDateFormat("MM/yyyy");
                cad=formato.format(hoy);
                break;
            }
            case 3: {
                formato=new SimpleDateFormat("MM/yy");
                cad=formato.format(hoy);
                break;
            }
            default: {
                cad="ERROR";
            }
        }

        return cad;
    }
}
```

3.- Factorial

Dado el siguiente método de la clase Factorial. Genera la batería de pruebas para validarlo.

```
public class Factorial {  
    public static int calculo(int n) {  
        //si n menor que 0 excepción  
        if (n<0) {  
            throw new IllegalArgumentException ("Número " + n + "no puede ser <0");  
        }  
  
        //inicializo factorial a 1  
        int fact=1;  
  
        //calculo factorial  
        for (int i=2;i<=n;i++)  
            fact*=i;  
        |  
  
        return fact;  
    }  
}
```

O bien:

```
package Factorial;  
  
public class Factorial {  
    int num;  
  
    public Factorial(int num) {  
        this.num = num;  
    }  
  
    public long calculo() {  
        if (num < 0) {  
            throw new IllegalArgumentException("Número " + num + " no puede ser <0");  
        }  
        long fact = 1;  
        for (int i = 2; i <= num; i++) {  
            fact *= i;  
        }  
        return fact;  
    }  
}
```

Subir:

- Captura del código Junit.
- Captura de la salida de los resultados.