

# DOCUMENTACIÓ

1-

Cambiem el nom del paquet

```
2122ed-u6p2 [2122ed-u6p
  > src/main/java
    > ed.antoni
      > Calculadora.java
```

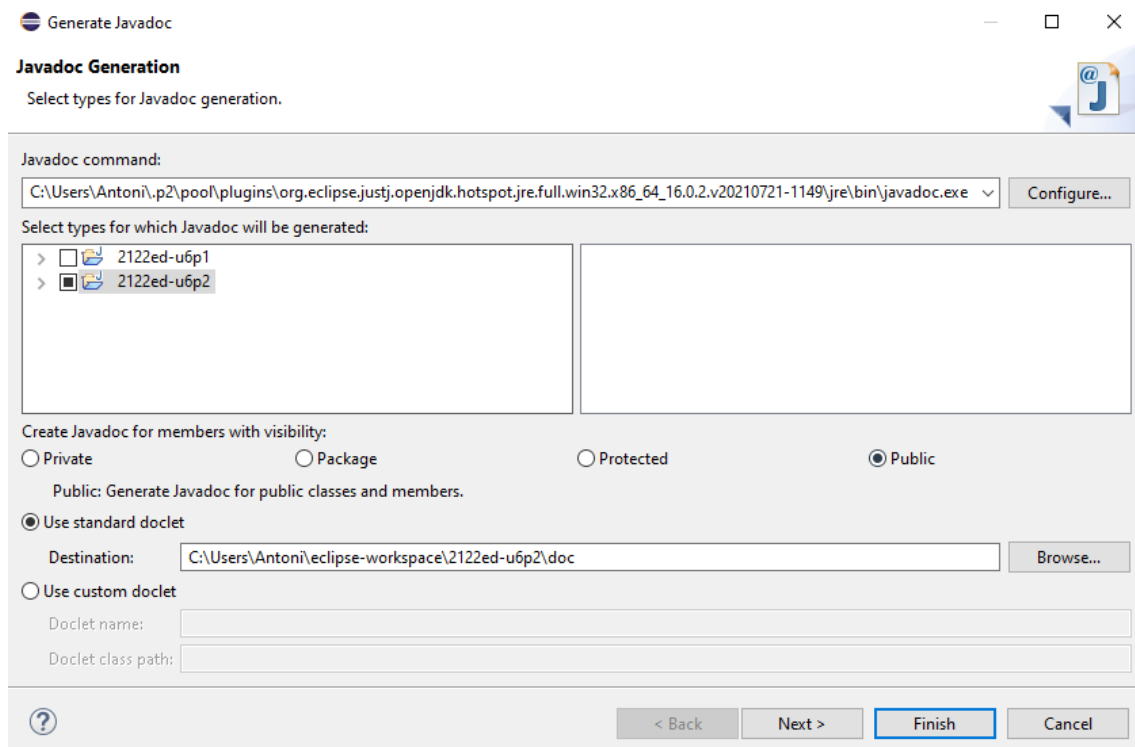
2-

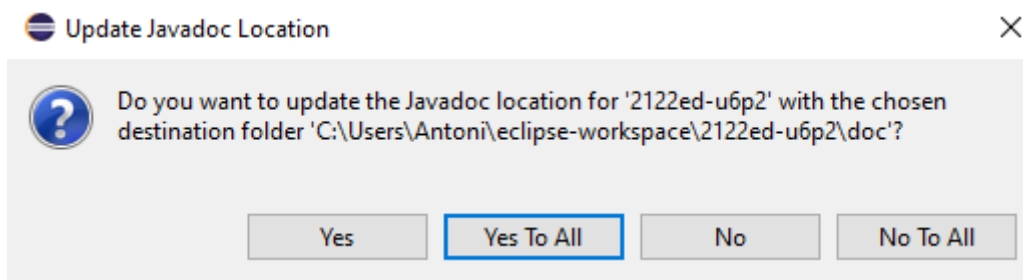
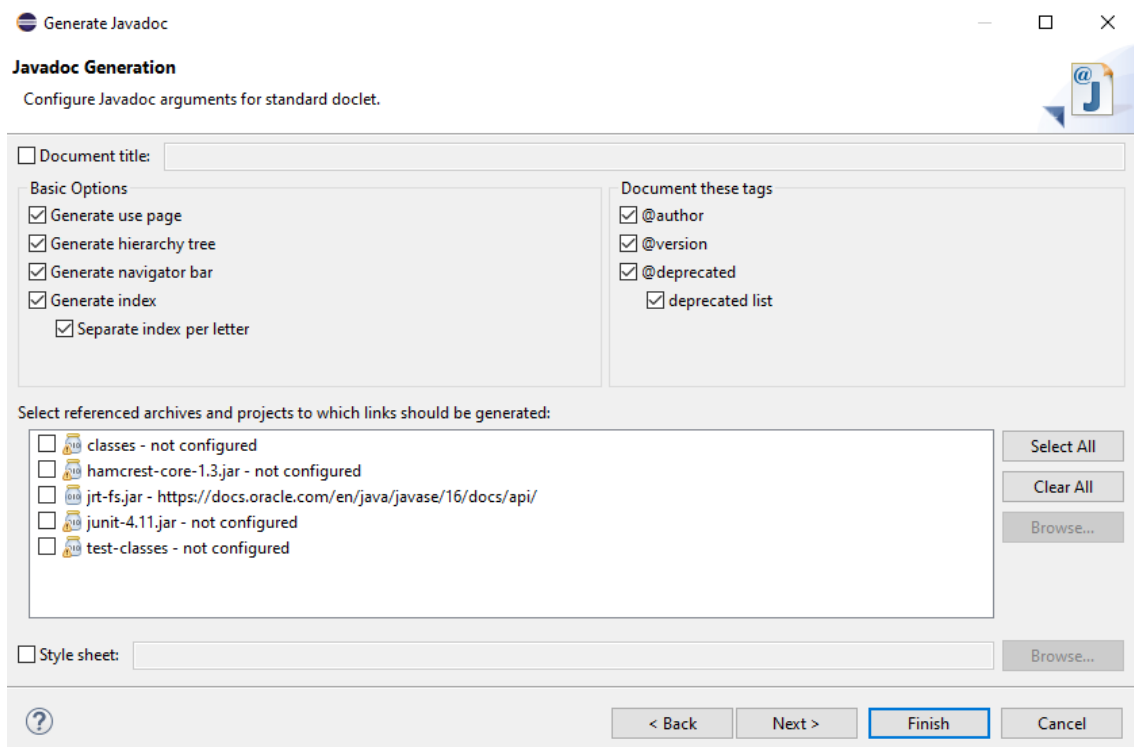
Refactorice i documente el codi

```
*Calculadora.java X
1 package ed.antoni;
2 /**
3  * La clase Calculadora devuelve el resultado al sumar, restar,
4  * multiplicar o dividir 2 numeros enteros
5  * @author Antoni Medina Sanjuan
6  * @version 2.5
7  * Disponible desde la versión 1.9, junio de 2019
8  */
9 public class Calculadora {
10
11     /**
12      *
13      * @param a numero entero 1
14      * @param b numero entero 2
15      * @return el total de la suma de los dos numeros enteros
16      * @exception java.lang.ArithmeticException Error, debe de ser entero
17      */
18     public int suma(int a, int b) {
19         return a + b;
20     }
21     /**
22      *
23      * @param a numero entero 1
24      * @param b numero entero 2
25      * @return el total de la resta de los dos numeros enteros
26      * @exception java.lang.ArithmeticException Error, debe de ser entero
27      */
28     public int resta(int a, int b) {
29         return a - b;
30     }
31     /**
32      *
33      * @param a numero entero 1
34      * @param b numero entero 2
35      * @return el total de la multiplicación de los dos numeros enteros
36      * @exception java.lang.ArithmeticException Error, debe de ser entero
37      */
38     public int multiplica(int a, int b){
39         return a*b;
40     }
41     /**
42      *
43      * @param a numero entero 1
44      * @param b numero entero 2
45      * @return el total de la división de los dos numeros enteros
46      * @exception java.lang.ArithmeticException Error, debe de ser entero y no debe de ser una division entre 0
```

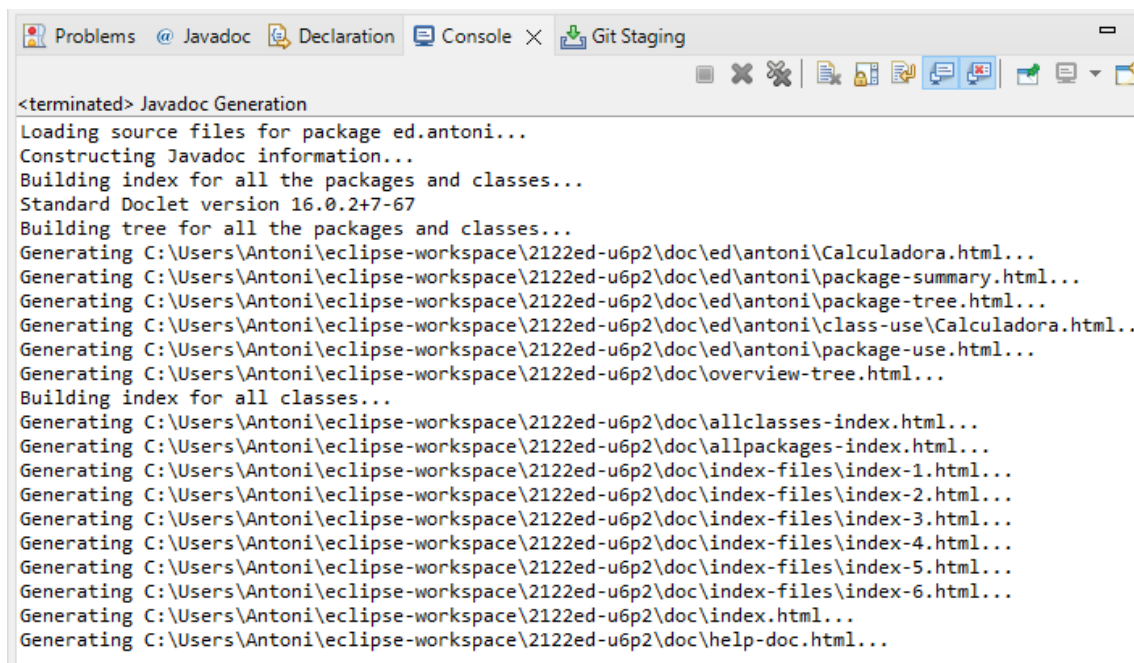
3-

Genere el javadoc i adjunte les captures de lo demanat

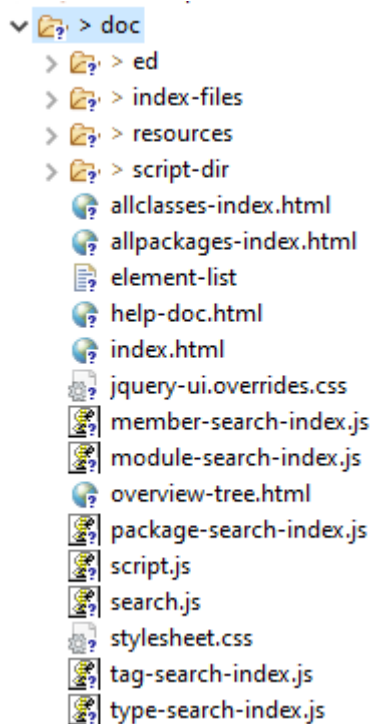




Console:

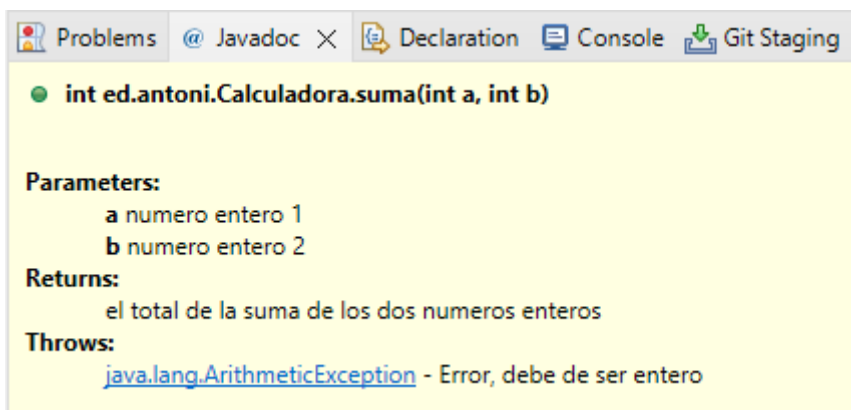
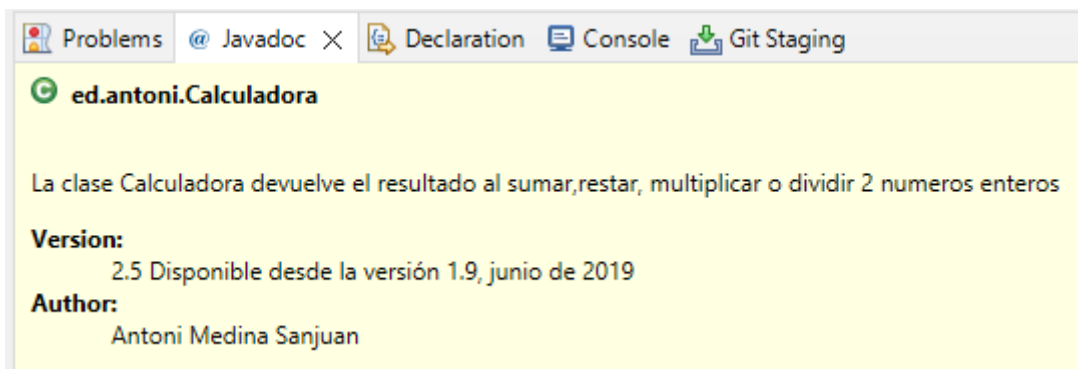


Package explorer de doc:



4-

Adjunte captures del javadoc i codi



Problems @ Javadoc X Declaration Console Git Staging

● **int ed.antoni.Calculadora.resta(int a, int b)**

**Parameters:**  
a numero entero 1  
b numero entero 2

**Returns:**  
el total de la resta de los dos numeros enteros

**Throws:**  
[java.lang.ArithmeticException](#) - Error, debe de ser entero

Problems @ Javadoc X Declaration Console Git Staging

● **int ed.antoni.Calculadora.multiplica(int a, int b)**

**Parameters:**  
a numero entero 1  
b numero entero 2

**Returns:**  
el total de la multiplicación de los dos numeros enteros

**Throws:**  
[java.lang.ArithmeticException](#) - Error, debe de ser entero

Problems @ Javadoc X Declaration Console Git Staging

● **int ed.antoni.Calculadora.diveix(int a, int b)**

**Parameters:**  
a numero entero 1  
b numero entero 2

**Returns:**  
el total de la división de los dos numeros enteros

**Throws:**  
[java.lang.ArithmeticException](#) - Error, debe de ser entero y no debe de ser una division entre 0

```
1 package ed.antoni;
2 /**
3  * La clase Calculadora devuelve el resultado al sumar,restar,
4  * multiplicar o dividir 2 numeros enteros
5  * @author Antoni Medina Sanjuan
6  * @version 2.5
7  * Disponible desde la versión 1.9, junio de 2019
8  */
9 public class Calculadora {
10
11     /**
12      *
13      * @param a numero entero 1
14      * @param b numero entero 2
15      * @return el total de la suma de los dos numeros enteros
16      * @exception java.lang.ArithmeticException Error, debe de ser entero
17      */
18     public int suma(int a, int b) {
19         return a + b;
20     }
21     /**
22      *
23      * @param a numero entero 1
24      * @param b numero entero 2
25      * @return el total de la resta de los dos numeros enteros
26      * @exception java.lang.ArithmeticException Error, debe de ser entero
27      */
28     public int resta(int a, int b) {
29         return a - b;
30     }
31     /**
32      *
33      * @param a numero entero 1
34      * @param b numero entero 2
35      * @return el total de la multiplicación de los dos numeros enteros
36      * @exception java.lang.ArithmeticException Error, debe de ser entero
37      */
38     public int multiplica(int a, int b){
39         return a*b;
40     }
41     /**
42      *
43      * @param a numero entero 1
44      * @param b numero entero 2
45      * @return el total de la división de los dos numeros enteros
46      * @exception java.lang.ArithmeticException Error, debe de ser entero y no debe de ser una division entre 0
47      */
48     public int diveix(int a, int b) {
49         return a/b;
50     }
51 }
52 }
53 }
```

5- Adjunte captures del index

## Package ed.antoni

package ed.antoni

Class Summary	
Class	Description
Calculadora	La clase Calculadora devuelve el resultado al sumar,restar, multiplicar o dividir 2 numeros enteros

Package `ed.antoni`

## Class `Calculadora`

`java.lang.Object`

`ed.antoni.Calculadora`

```
public class Calculadora
```

```
extends Object
```

La clase `Calculadora` devuelve el resultado al sumar,restar, multiplicar o dividir 2 numeros enteros

**Version:**

2.5 Disponible desde la versiÃ³n 1.9, junio de 2019

**Author:**

Antoni Medina Sanjuan

### Constructor Summary

#### Constructors

Constructor	Description
-------------	-------------

<code>Calculadora()</code>	
----------------------------	--

### Method Summary

#### All Methods

#### Instance Methods

#### Concrete Methods

Modifier and Type	Method
-------------------	--------

Description
-------------

<code>int</code>	<code>divideix(int a, int b)</code>
------------------	-------------------------------------

<code>int</code>	<code>multiplica(int a, int b)</code>
------------------	---------------------------------------

<code>int</code>	<code>resta(int a, int b)</code>
------------------	----------------------------------

int            **multiplica**(int a, int b)

int            **resta**(int a, int b)

int            **suma**(int a, int b)

#### Methods inherited from class java.lang.Object<sup>Ⓜ</sup>

[equals<sup>Ⓜ</sup>](#), [getClass<sup>Ⓜ</sup>](#), [hashCode<sup>Ⓜ</sup>](#), [notify<sup>Ⓜ</sup>](#), [notifyAll<sup>Ⓜ</sup>](#), [toString<sup>Ⓜ</sup>](#), [wait<sup>Ⓜ</sup>](#), [wait<sup>Ⓜ</sup>](#), [wait<sup>Ⓜ</sup>](#)

### Constructor Details

#### Calculadora

```
public Calculadora()
```

### Method Details

#### suma

```
public int suma(int a,  
                int b)
```

##### Parameters:

a - numero entero 1



**Parameters:**

a - numero entero 1

b - numero entero 2

**Returns:**

el total de la suma de los dos numeros enteros

**Throws:**

`ArithmeticException`<sup>19</sup> - Error, debe de ser entero

## resta

```
public int resta(int a,  
                int b)
```

**Parameters:**

a - numero entero 1

b - numero entero 2

**Returns:**

el total de la resta de los dos numeros enteros

**Throws:**

`ArithmeticException`<sup>19</sup> - Error, debe de ser entero

## multiplica

```
public int multiplica(int a,  
                     int b)
```

**Parameters:**

a - numero entero 1

b - numero entero 2

**Returns:**

el total de la multiplicaci3n de los dos numeros enteros

**Throws:**

**Returns:**

el total de la resta de los dos numeros enteros

**Throws:**

ArithmeticException<sup>®</sup> - Error, debe de ser entero

## multiplica

```
public int multiplica(int a,  
                     int b)
```

**Parameters:**

a - numero entero 1

b - numero entero 2

**Returns:**

el total de la multiplicaci3n de los dos numeros enteros

**Throws:**

ArithmeticException<sup>®</sup> - Error, debe de ser entero

## diveix

```
public int diveix(int a,  
                 int b)
```

**Parameters:**

a - numero entero 1

b - numero entero 2

**Returns:**

el total de la divisi3n de los dos numeros enteros

**Throws:**

ArithmeticException<sup>®</sup> - Error, debe de ser entero y no debe de ser una division entre 0

## 6- Adició dels mètodes, refactorització, documentació del codi y capturació de lo necessari

```
53  *
54  * @param base de la potencia, se multiplica por sí mismo tantas veces como indique el otro numero llamado exponente
55  * @param exponente es el numero que indica las veces que la base se multiplicará por sí mismo
56  * @return el resultado de la potencia
57  */
58  public int calculaPotencia ( int base , int exponente ){
59
60      int totalPotencia = 1;
61
62      for ( int i = 0; i < exponente ; i++){
63          totalPotencia = totalPotencia * base;
64      }
65
66      return totalPotencia;
67
68  }
69
70  /**
71   *
72   * @param numero que comprobaremos si es par o impar
73   * @return true si es par, false si es impar
74   */
75  public boolean esPar ( int numero ){
76
77      boolean esPar;
78
79      if ( numero % 2 == 0) {
80          esPar = true;
81      } else {
82          esPar = false;
83      }
84
85      return esPar;
86
87  }
88  /**
89   *
90   * @param entero al que convertiremos en numero binario
91   * @return el entero pasado a binario
92   */
93  public String conversorABinari(int entero){
94
95      String resultatBinari = "";
96      int numeroEntero = entero;
97      int resto;
98
99      while ( numeroEntero > 0) {
100
101          resto = numeroEntero % 2;
102
103          resultatBinari = resto + resultatBinari;
104
105          numeroEntero = numeroEntero / 2;
106
107      }
108      return resultatBinari;
```

Problems @ Javadoc Declaration Console Git Staging

● **int ed.antoni.Calculadora.calculaPotencia(int base, int exponente)**

**Parameters:**  
base de la potencia, se multiplica por sí mismo tantas veces como indique el otro numero llamado exponente  
exponente es el numero que indica las veces que la base se multiplicará por sí mismo

**Returns:**  
el resultado de la potencia

Problems @ Javadoc Declaration Console Git Staging

● **boolean ed.antoni.Calculadora.esPar(int numero)**

**Parameters:**  
numero que comprobaremos si es par o impar

**Returns:**  
true si es par, false si es impar

## String ed.antoni.Calculadora.conversorABinari(int entero)

### Parameters:

**entero** al que convertiremos en numero binario

### Returns:

el entero pasado a binario

## Method Summary

All Methods	Instance Methods	Concrete Methods
Modifier and Type	Method	Description
int	<code>calculaPotencia(int base, int exponente)</code>	
String <sup>Ⓜ</sup>	<code>conversorABinari(int entero)</code>	
int	<code>diveix(int a, int b)</code>	
boolean	<code>esPar(int numero)</code>	
int	<code>multiplica(int a, int b)</code>	
int	<code>resta(int a, int b)</code>	
int	<code>suma(int a, int b)</code>	
Methods inherited from class java.lang.Object <sup>Ⓜ</sup>		
<code>equals<sup>Ⓜ</sup>, getClass<sup>Ⓜ</sup>, hashCode<sup>Ⓜ</sup>, notify<sup>Ⓜ</sup>, notifyAll<sup>Ⓜ</sup>, toString<sup>Ⓜ</sup>, wait<sup>Ⓜ</sup>, wait<sup>Ⓜ</sup>, wait<sup>Ⓜ</sup></code>		

### calculaPotencia

```
public int calculaPotencia(int base,  
                           int exponente)
```

**Parameters:**

base - de la potencia, se multiplica por s  mismo tantas veces como indique el otro numero llamado exponente

exponente - es el numero que indica las veces que la base se multiplicar ; por s  mismo

**Returns:**

el resultado de la potencia

### esPar

```
public boolean esPar(int numero)
```

**Parameters:**

numero - que comprobaremos si es par o impar

**Returns:**

true si es par, false si es impar

### conversorABinari

```
public String  conversorABinari(int entero)
```

**Parameters:**

entero - al que convertiremos en numero binario

**Returns:**

el entero pasado a binario