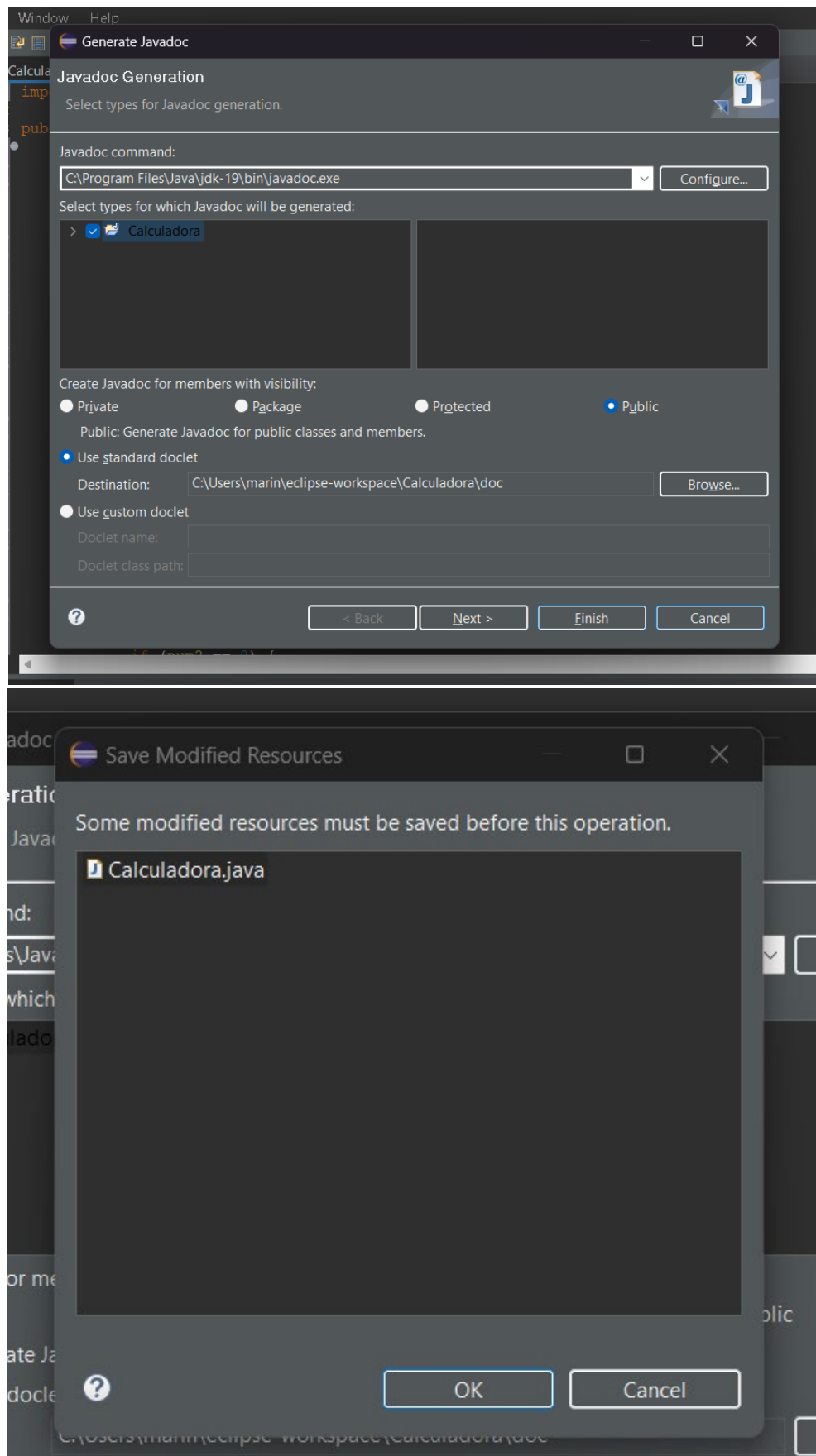
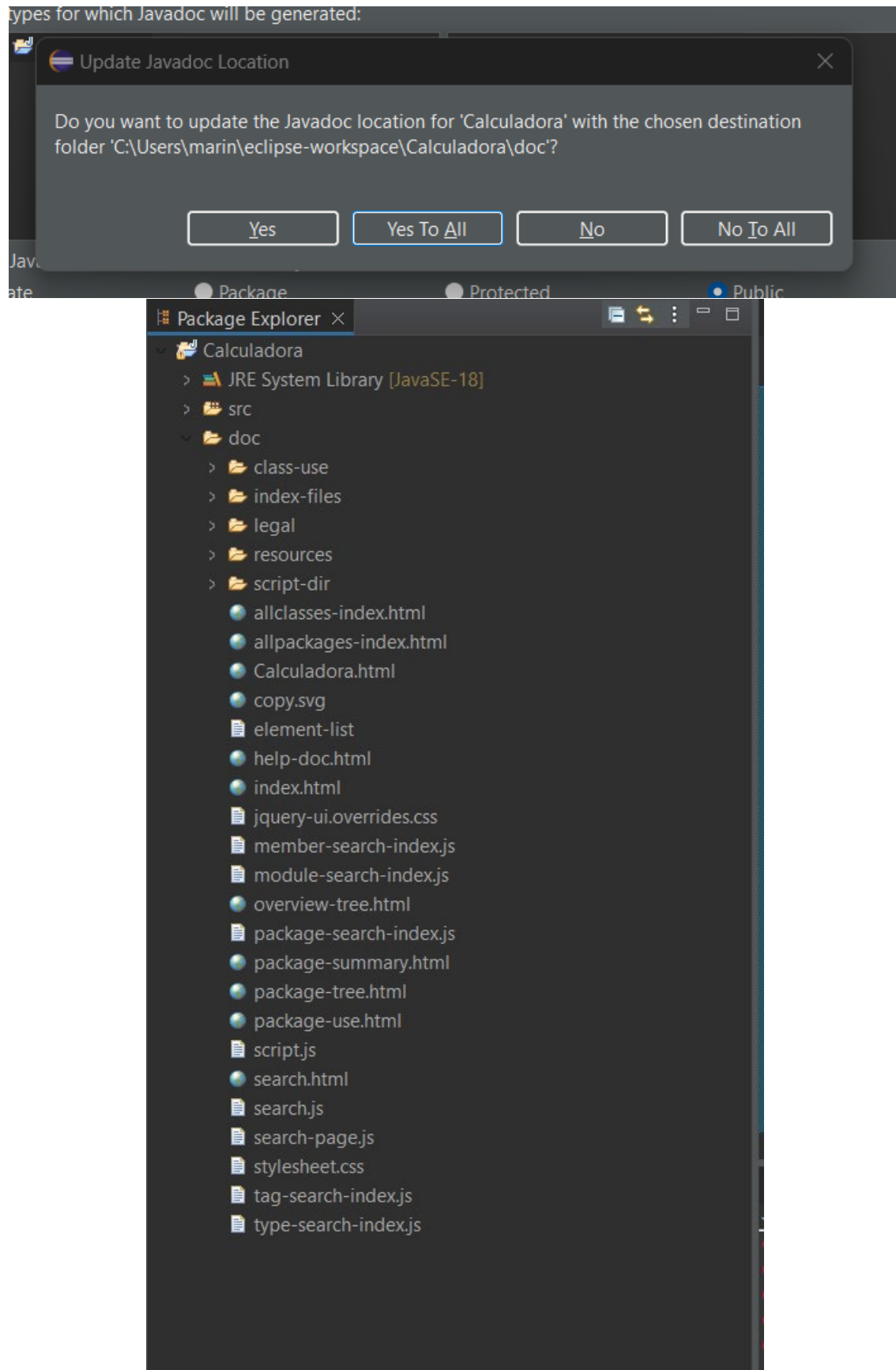
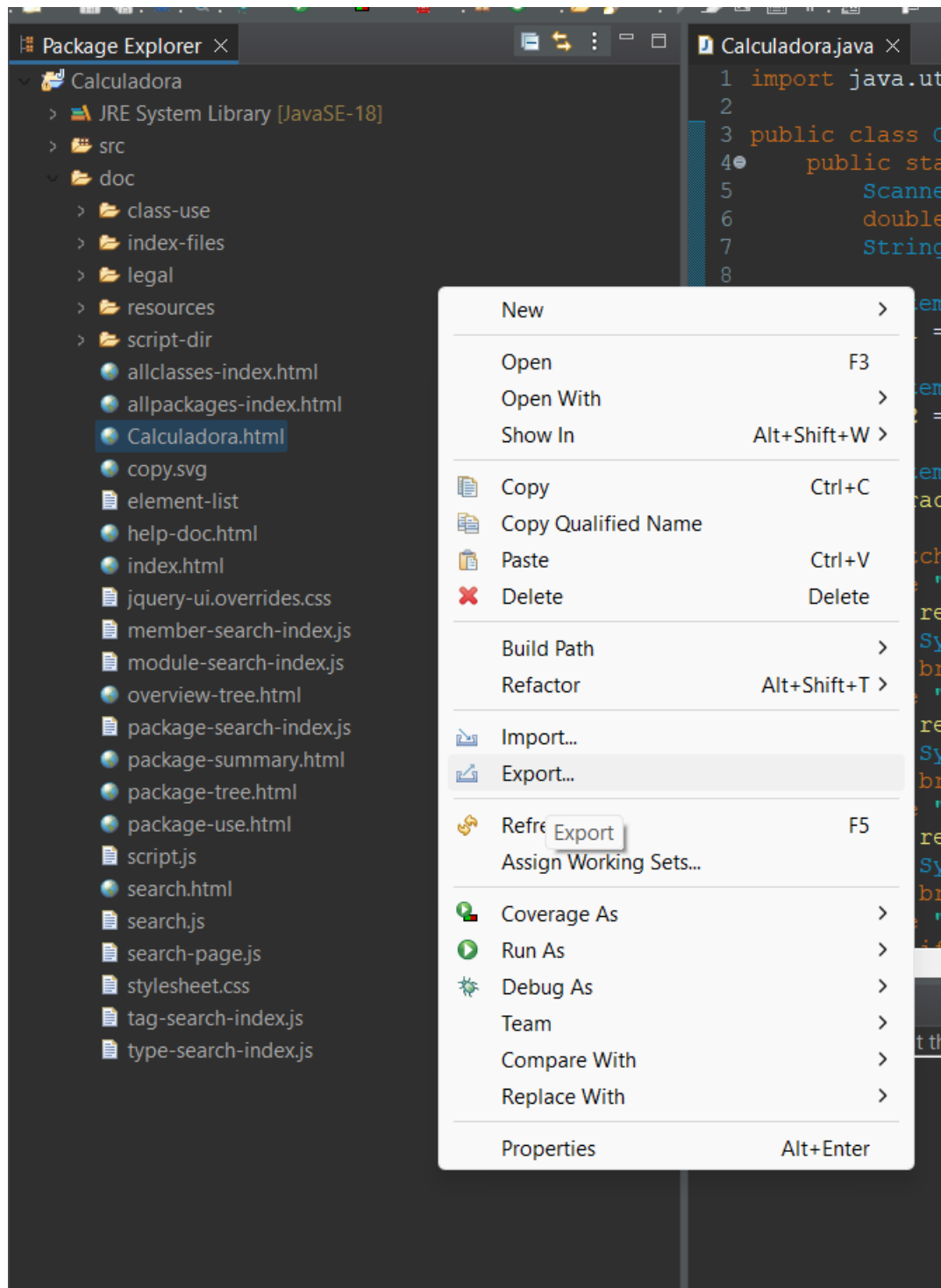
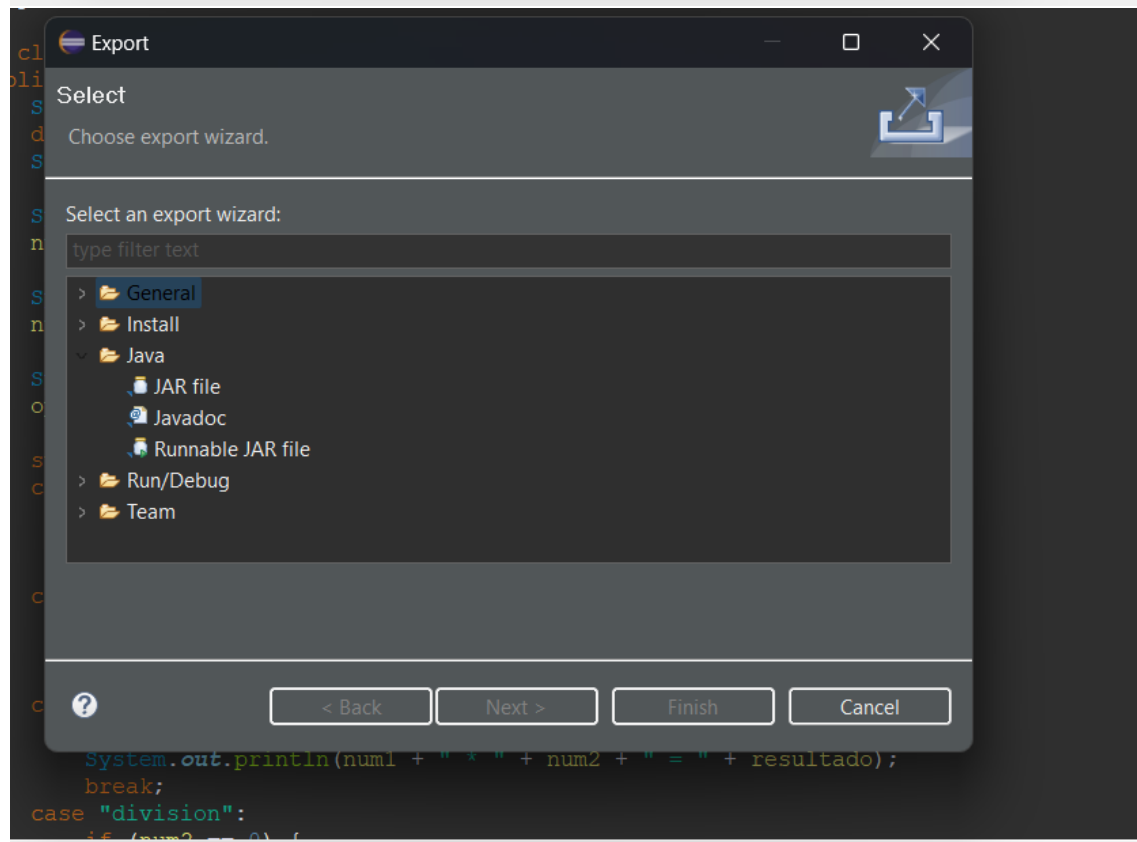
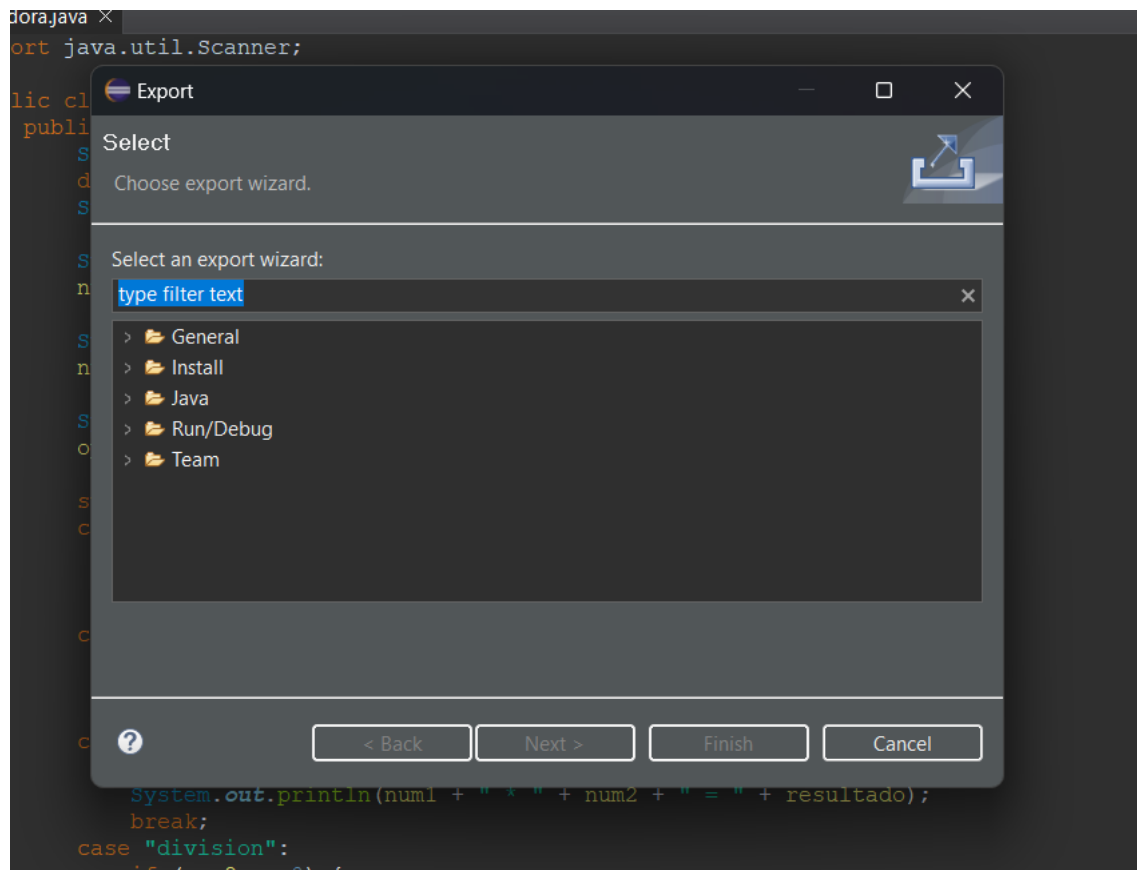


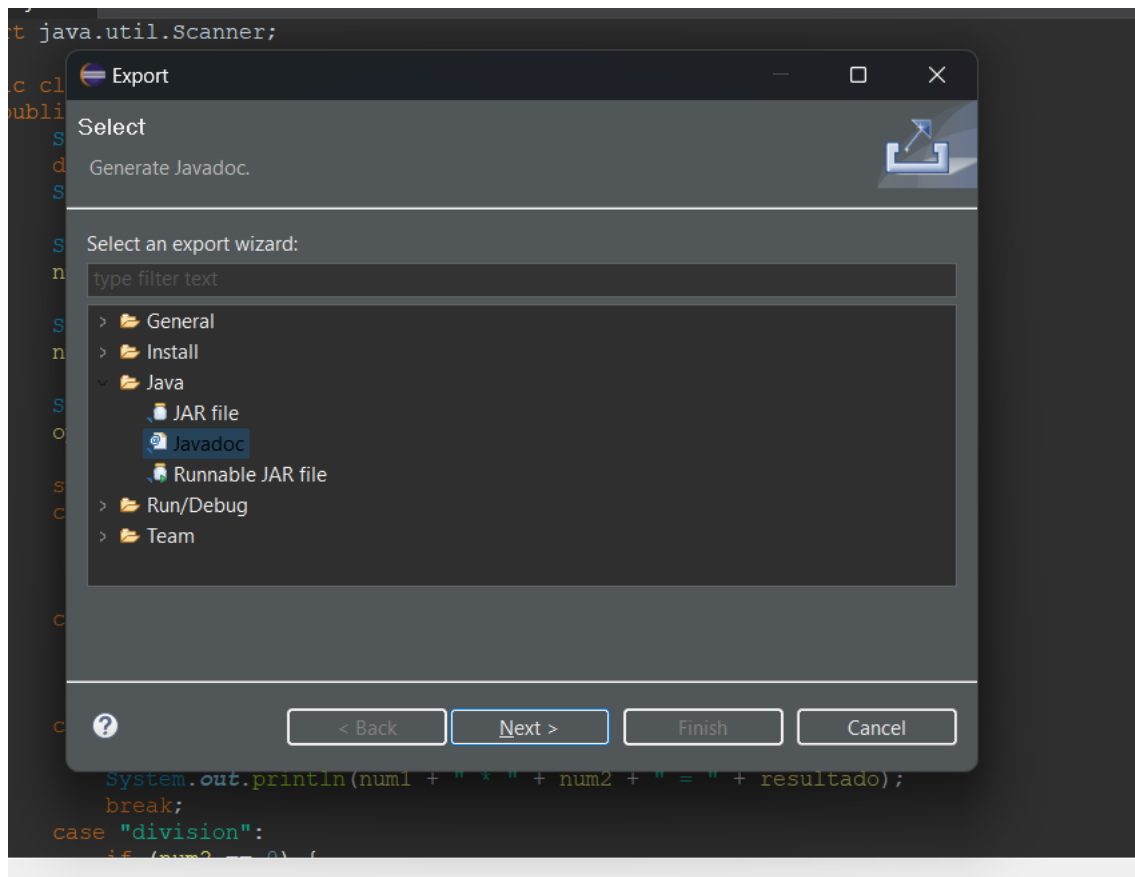
Manual para crear Javadoc desde eclipse

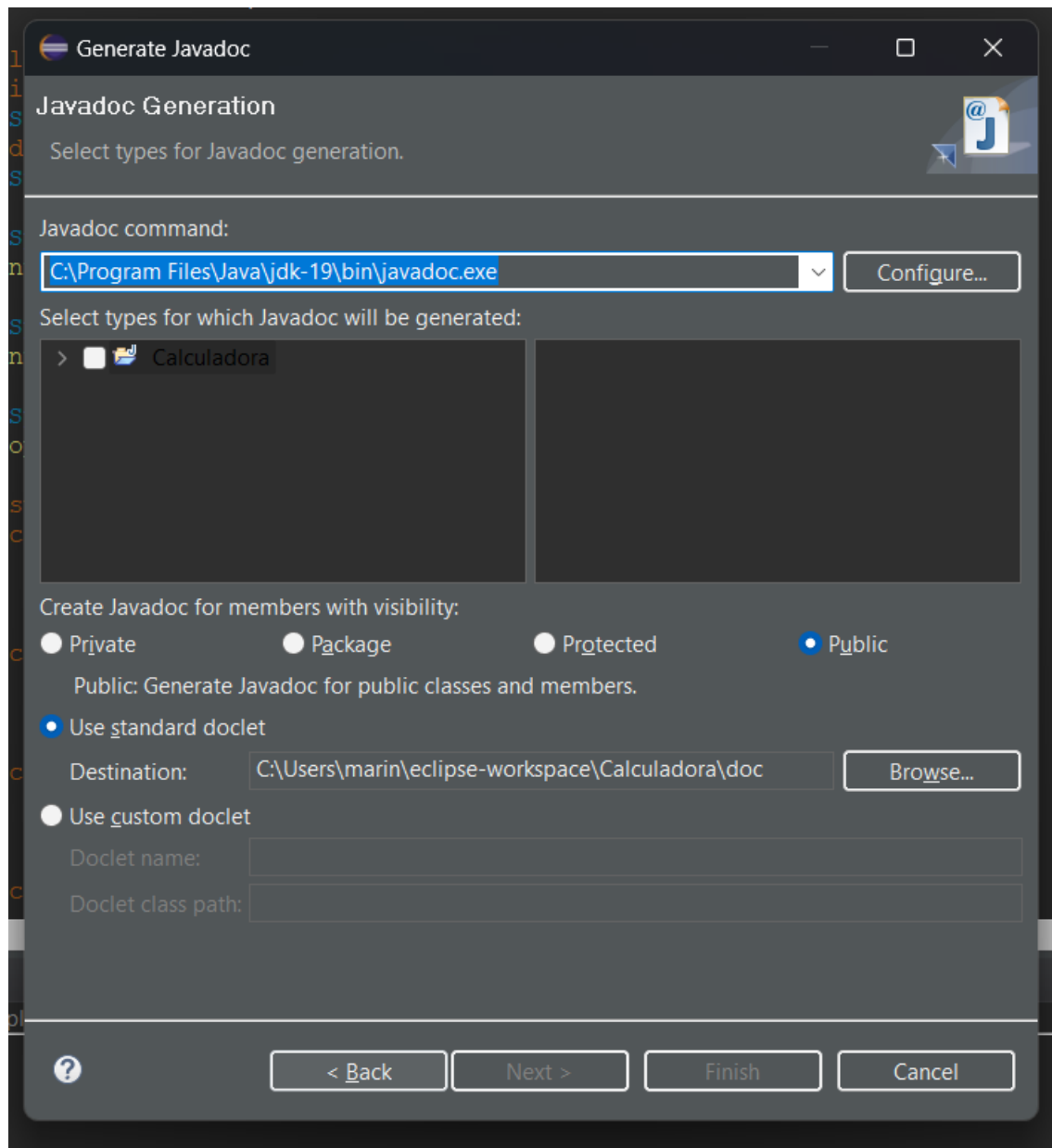








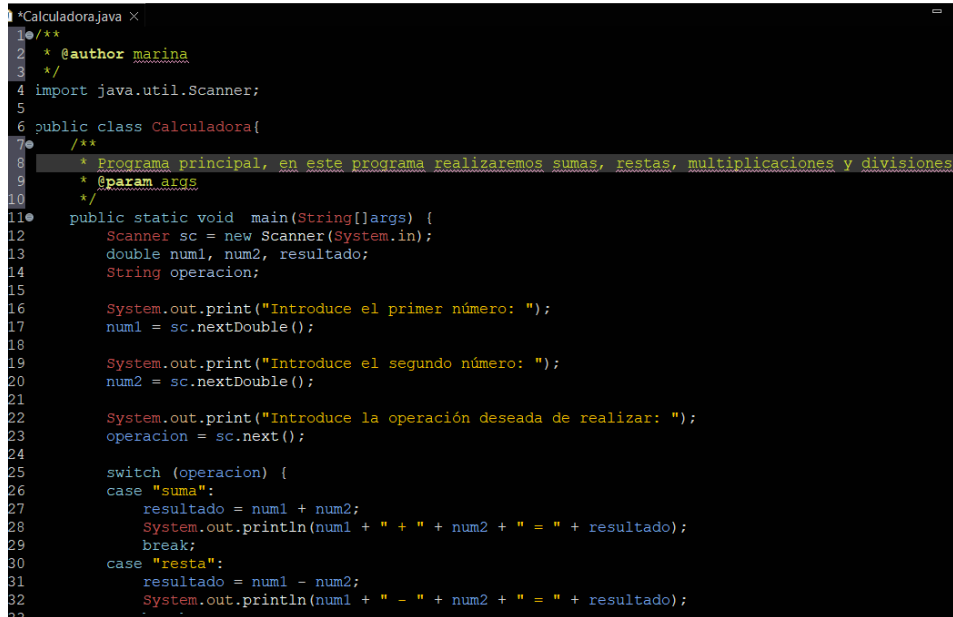




Para crear el Javadoc lo primero que debemos tener es el código del programa del que realizaremos el Javadoc. En el código empezaremos a poner los comentarios que corresponden al Javadoc su sintaxis es la siguiente.

```
/**
 *
 */
```

Cada párrafo nuevo que queramos poner empezara por un *

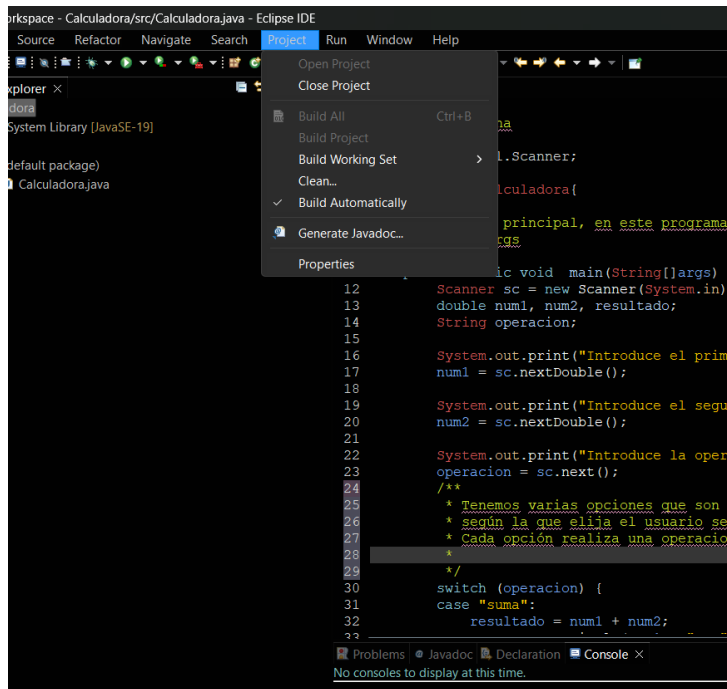


```
1  *Calculadora.java x
2  /**
3   * @author marina
4   */
5  import java.util.Scanner;
6  public class Calculadora{
7      /**
8       * Programa principal, en este programa realizaremos sumas, restas, multiplicaciones y divisiones
9       * @param args
10      */
11     public static void main(String[] args) {
12         Scanner sc = new Scanner(System.in);
13         double num1, num2, resultado;
14         String operacion;
15
16         System.out.print("Introduce el primer número: ");
17         num1 = sc.nextDouble();
18
19         System.out.print("Introduce el segundo número: ");
20         num2 = sc.nextDouble();
21
22         System.out.print("Introduce la operación deseada de realizar: ");
23         operacion = sc.next();
24
25         switch (operacion) {
26             case "suma":
27                 resultado = num1 + num2;
28                 System.out.println(num1 + " + " + num2 + " = " + resultado);
29                 break;
30             case "resta":
31                 resultado = num1 - num2;
32                 System.out.println(num1 + " - " + num2 + " = " + resultado);
33         }
34     }
35 }
```

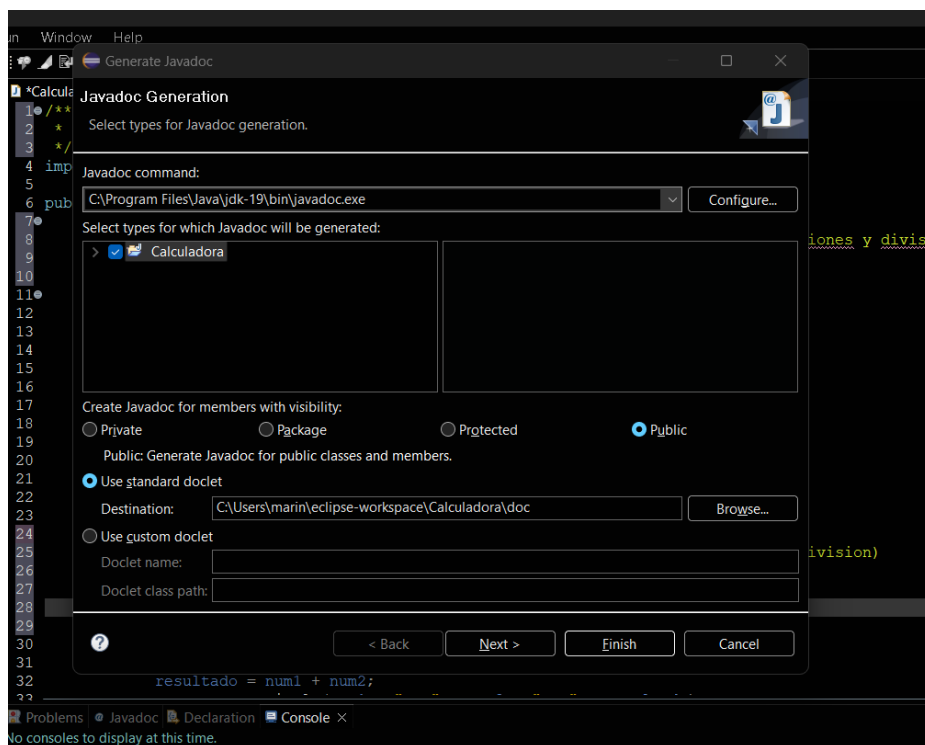
Como observamos el primer comentario que comienza antes que el código del programa contiene algo específico que es el @author.

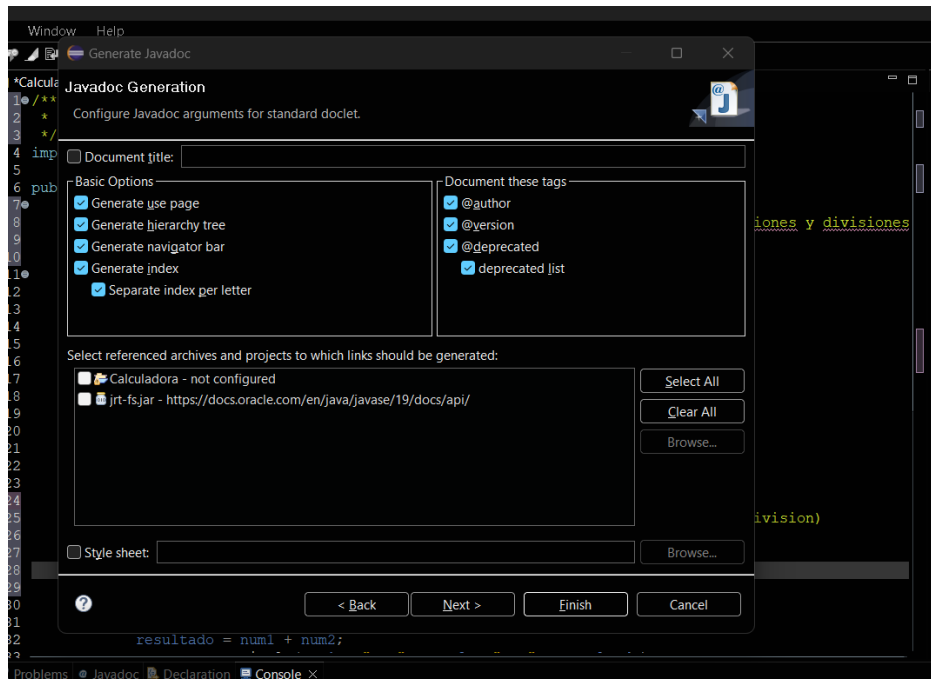
Mientras que el segundo comentario que observamos tiene una pequeña descripción de lo que va a realizar el programa principal y continuadamente vemos que tenemos una línea nueva en la que tenemos @param con esto lo que hacemos es especificar cuales son los parámetros que se le pasa a la función.

Una vez que tengamos nuestros comentarios colocados en el programa principal continuaremos con el siguiente paso a realizar, el cual es generar el Javadoc. Desde el IDE de Eclipse es tan fácil como entrar en el menú de Project el cual esta situado en la parte de arriba y veremos que en la penúltima posición nos aparece lo siguiente: Generate Javadoc...

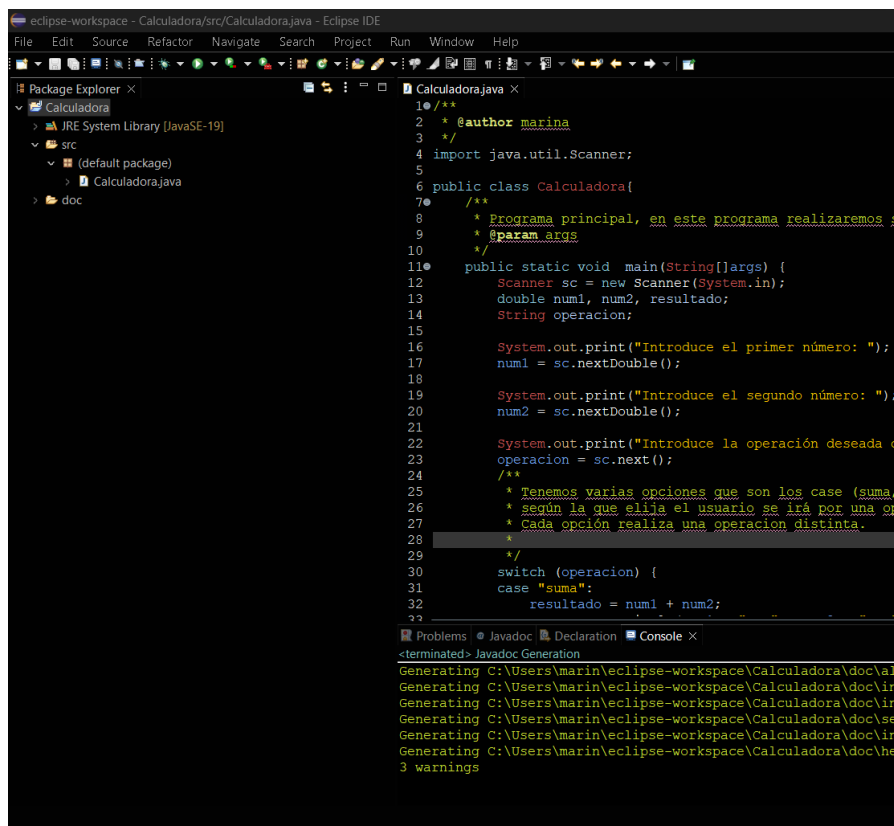


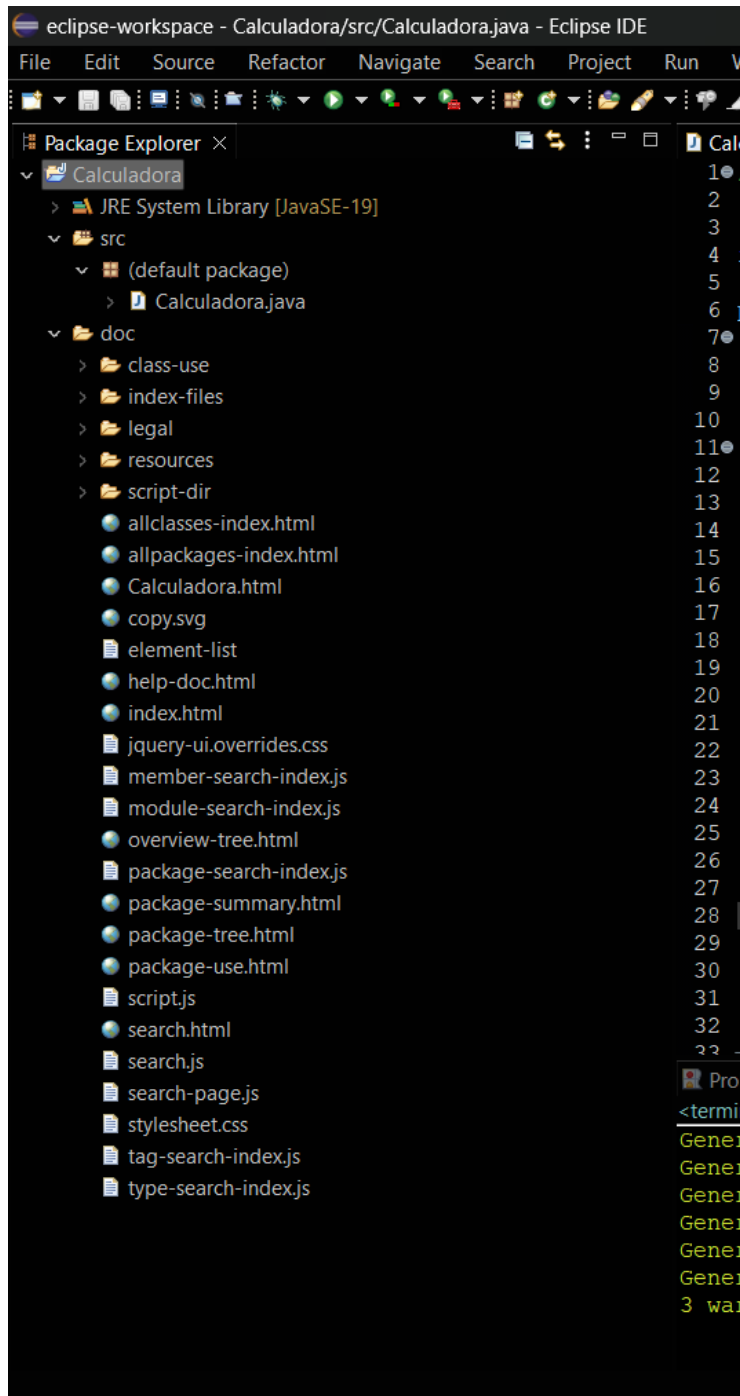
Seleccionaremos esta opción para generarlo y empezaremos a poner unos ajustes para poder generarlo, los cuales dejaremos los que nos vienen por defecto.





Una vez hayamos finalizado, observaremos como empieza a generarse el Javadoc y una vez finalizado encontraremos una nueva carpeta debajo de las carpetas de nuestro proyecto, esta carpeta tendrá el nombre de doc por defecto.





En esta carpeta encontraremos el archivo index.html, este es el archivo del Javadoc, el cual abriremos con el navegador para ver como ha quedado el documento que nos ha generado.

A continuación veremos como nos ha quedado el Javadoc de un programa algo básico.

En la primera página tendremos las clases de nuestro código.

PACKAGE CLASS USE TREE INDEX HELP	
PACKAGE: DESCRIPTION RELATED PACKAGES CLASSES AND INTERFACES	
Unnamed Package	
Classes	
Class	Description
Calculadora	

Pinchando en el nombre de la clase entraremos a su contenido. Podremos ver los constructores, métodos que nos encontremos en el código de esta clase.

PACKAGE

CLASS

USE

TREE

INDEX

HELP

SUMMARY: NESTED | FIELD | CONSTR | METHOD

DETAIL: FIELD | CONSTR | METHOD

Class Calculadora

java.lang.Object

Calculadora

public class Calculadora

extends Object

Constructor Summary

Constructors

Constructor	Description
Calculadora()	

Method Summary

All Methods

Static Methods

Concrete Methods

Modifier and Type	Method	Description
static void	main(String[] args)	Programa principal, en este programa realizaremos sumas, restas, multiplicaciones y c

Methods inherited from class java.lang.Object

equals

getClass

hashCode

notify

notifyAll

toString

wait

wait

wait

Constructor Details

PACKAGE
CLASS
USE
TREE
INDEX
HELP

SUMMARY
NESTED
FIELD
CONSTR
METHOD
DETAIL
FIELD
CONSTR
METHOD

SEARCH
Search

All Methods
Static Methods
Concrete Methods

Modifier and Type	Method	Description
static void	main(String[] args)	Programa principal, en este programa realizaremos sumas, restas, multiplicaciones y divisiones

Methods inherited from class java.lang.Object
equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Details

Calculadora

public Calculadora()

Method Details

main

public static void main(String[] args)

Programa principal, en este programa realizaremos sumas, restas, multiplicaciones y divisiones

Parameters:

args -

