

ED UD6

ACTIVIDAD 2

3. Genera en eclipse la documentación JavaDoc del proyecto.

- Haz una captura de la **Consola**: debe verse todo el proceso de generación completo.

The screenshot shows the Eclipse IDE interface. The Project Explorer on the left displays a project named 'Practica2_UD6 [u6p2]'. The Java Editor on the right shows a code snippet for a class named 'AppTest'. The Run View at the bottom shows the command 'Javadoc' being executed, with a list of files being generated, such as 'index-6.html', 'index-7.html', 'search.html', 'overview-summary.html', and 'help-doc.html'. The status bar at the bottom indicates that the 'javadoc' command finished with exit code 0.

```
import ...
/*
 * Unit test for simple App.
 */
public class AppTest {
    /**
     * Constructor de la clase, obligatorio ponerlo ya que si no nos da warning
     */
    public AppTest(){
        new AppTest();
    }
    /**
     * Rigorous Test :-)
     */
}
```

Run: Javadoc

```
Generating /Users/fran/IdeaProjects/Practica2_UD6/javadoc/index-files/index-6.html...
Generating /Users/fran/IdeaProjects/Practica2_UD6/javadoc/index-files/index-7.html...
Generating /Users/fran/IdeaProjects/Practica2_UD6/javadoc/search.html...
Generating /Users/fran/IdeaProjects/Practica2_UD6/javadoc/overview-summary.html...
Generating /Users/fran/IdeaProjects/Practica2_UD6/javadoc/help-doc.html...

"javadoc" finished with exit code 0
```

- Haz una captura en la vista **Package Explorer** del directorio **doc** resultante.

The screenshot shows the IntelliJ IDEA interface with the 'Package Explorer' view on the left and a code editor on the right. The 'Package Explorer' shows a project structure with a 'doc' directory expanded, containing various files like 'index.html', 'copy.svg', and 'script-dir'. The code editor displays a Java test class 'AppTest.java' with the following content:

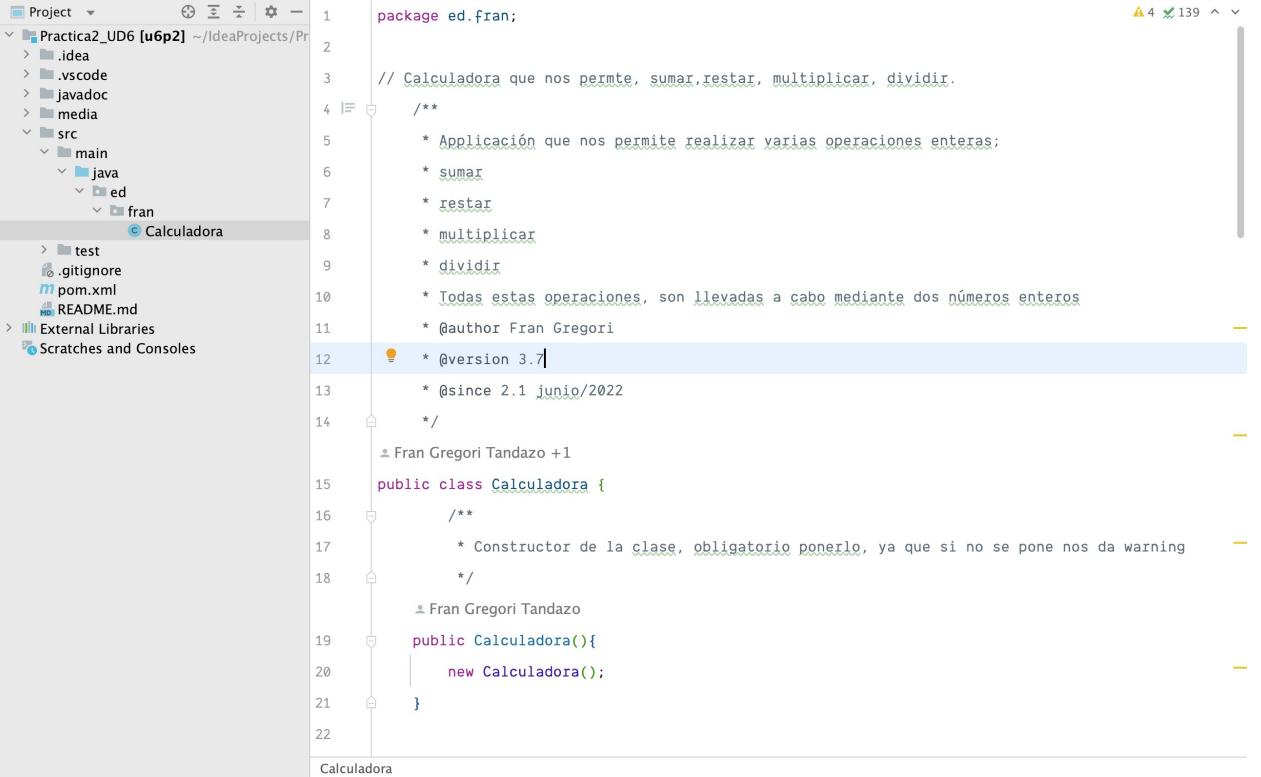
```

3 import ...
6
7 /**
8 * Unit test for simple App.
9 */
10 public class AppTest {
11     /**
12      * Constructor de la clase, obligatorio ponerlo ya que si no nos da warning
13     */
14     public AppTest(){
15         new AppTest();
16     }
17     /**
18      * Rigorous Test :-)
19     */
20     @Test
21     public void shouldAnswerWithTrue(){
22     }
23     assertTrue( condition: true );
24 }
25

```

A tooltip 'Camilo Juan' appears over the comment block at lines 18-19. The status bar at the bottom right shows '3 3 ^ v'.

4. Haz una captura del código fuente documentado resultante de la clase Calculadora



The screenshot shows the IntelliJ IDEA interface with the project 'Practica2_UD6 [u6p2]' open. The 'src/main/java' package contains the 'Calculadora' class. The code is annotated with Javadoc comments describing the class's purpose, operations, and authorship.

```
1 package ed.fran;
2
3 // Calculadora que nos permite, sumar,restar, multiplicar, dividir.
4 /**
5  * Aplicación que nos permite realizar varias operaciones enteras;
6  * sumar
7  * restar
8  * multiplicar
9  * dividir
10 * Todas estas operaciones, son llevadas a cabo mediante dos números enteros
11 * @author Fran Gregori
12 * @version 3.1
13 * @since 2.1 junio/2022
14 */
15 public class Calculadora {
16     /**
17      * Constructor de la clase, obligatorio ponerlo, ya que si no se pone nos da warning
18      */
19     /**
20      * Fran Gregori Tandazo
21     */
22 }
```

```
Project Practica2_UD6 [u6p2] ~/IdeaProjects/Practica2_UD6 [u6p2]
  .idea
  .vscode
  javadoc
  media
  src
    main
      java
        ed
        fran
          Calculadora
            Calculadora
              public Calculadora(){
                new Calculadora();
              }
              /**
               * Devolver la suma de dos números enteros pasados como parámetro
               * @param a de tipo entero
               * @param b de tipo entero
               * @return devuele la suma de dos números enteros
               * @throws RuntimeException Tipo de dato no esperado, es decir,
               * un tipo de dato superior al int o diferente al int, tales como String, char etc...
               * No son de obligada captura
               */
              no usages Camilo Juan
              public int suma(int a, int b) { return a + b; }

              /**
               * Devolver la resta de dos números enteros pasados como parámetro
               * @param a de tipo entero
               * @param b de tipo entero
               * @return devuele la resta de dos números enteros
               * @throws RuntimeException Tipo de dato no esperado, es decir,
               * un tipo de dato superior al int o diferente al int, tales como String, char etc...
               */
              Calculadora
```

```
36    /**
37     * Devolver la resta de dos números enteros pasados como parámetro
38     * @param a de tipo entero
39     * @param b de tipo entero
40     * @return devuele la resta de dos números enteros
41     * @throws RuntimeException Tipo de dato no esperado, es decir,
42     * un tipo de dato superior al int o diferente al int, tales como String, char etc...
43     * No son de obligada captura, por lo tanto son de tipo RunTimeException
44     */
45    public int resta(int a, int b) { return a - b; }
46
47    /**
48     * Devolver la multiplicación de dos números enteros pasados como parámetro
49     * @param a de tipo entero
50     * @param b de tipo entero
51     * @return devuele la multiplicacion de dos números enteros
52     * @throws RuntimeException Tipo de dato no esperado, es decir,
53     * un tipo de dato superior al int o diferente al int, tales como String, char etc...
54     * 0 dividir entre 0, no son de obligada captura
55     */
56    public int multiplica(int a, int b) { return a*b; }
```

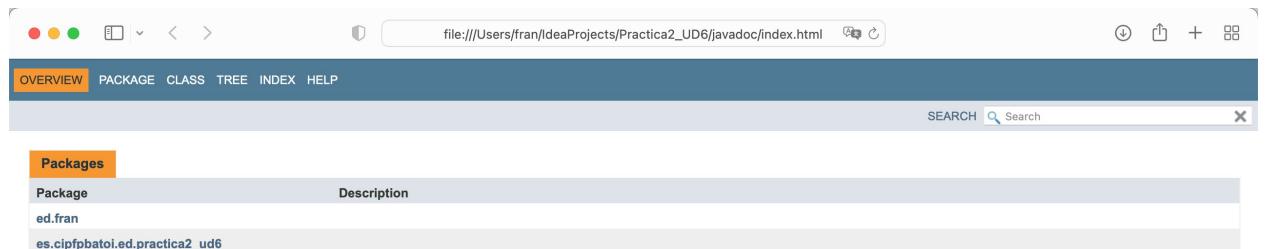
The screenshot shows the IntelliJ IDEA interface with the following details:

- Project:** Practica2_UD6 [u6p2] (~/IdeaProjects/Practica2_UD6)
- File:** Calculadora.java
- Code Content:**

```
57    */
58    no usages  ↳ Camilo Juan
59    public int multiplica(int a, int b) { return a*b; }
60
61    /**
62     * Devolver la división de dos números enteros pasados como parámetro
63     * @param a de tipo entero
64     * @param b de tipo entero
65     * @return devuele la división de dos números enteros
66     * @throws RuntimeException error al introducir un tipo de dato no esperado, es decir,
67     * un tipo de dato superior al int o diferente al int, tales como String, char etc...
68     * Además de una posible opción, dividir un número entero,
69     * esto genera una excepción del tipo ArithmeticException, no es de obligada captura.
70     * No son es obligada captura, por lo tanto son de tipo RunTimeException
71     */
72
73    no usages  ↳ Camilo Juan
74    public int divide(int a, int b) { return a/b; }
75
76    }
```

5. Muestra el resultado de la documentación:

- Muestra el fichero **index.html** en el **Navegador** integrado en **Eclipse** y muestra capturas de **TODO** su contenido. Fíjate en la correspondencia entre las descripciones y tags de las páginas web (*html*) generadas.



The screenshot shows a JavaDoc interface for a package named 'ed.fran'. The title bar includes standard window controls and the URL 'an/ideaProjects/Practica2_UD6/javadoc/ed/fran/package-summary.html'. The menu bar has tabs for 'OVERVIEW', 'PACKAGE' (which is selected), 'CLASS', 'TREE', 'INDEX', and 'HELP'. Below the menu is a search bar with the placeholder 'SEARCH' and a magnifying glass icon. A link 'PACKAGE: DESCRIPTION | RELATED PACKAGES | CLASSES AND INTERFACES' is also present. The main content area is titled 'Package ed.fran' and contains the package declaration 'package ed.fran'. A 'Classes' tab is selected, showing a table with one row. The table has two columns: 'Class' and 'Description'. The single entry is 'Calculadora', which is described as an application that allows performing various integer operations: add, subtract, multiply, and divide. All operations are carried out using two integer numbers.

Class	Description
Calculadora	Aplicación que nos permite realizar varias operaciones enteras; sumar restar multiplicar dividir Todas estas operaciones, son llevadas a cabo mediante dos números enteros

The screenshot shows a Java documentation interface with the following details:

- Toolbar:** Includes icons for back, forward, search, and other navigation.
- Header:** Shows the path "C:\Users\fran\IdeaProjects\Practica2_UD6\javadoc\ed\fran\Calculadora.html" and various menu options: OVERVIEW, PACKAGE, CLASS (highlighted in orange), TREE, INDEX, HELP.
- Search Bar:** A search input field with a magnifying glass icon and a clear button.
- Summary Section:** Contains the package name "Package ed.fran" and the class name "Class Calculadora". It also lists the superclass "java.lang.Object" and the package "ed.fran.Calculadora".
- Description:** A brief description: "Aplicación que nos permite realizar varias operaciones enteras; sumar restar multiplicar dividir. Todas estas operaciones, son llevadas a cabo mediante dos números enteros".
- Since:** The date "2.1 junio/2022".
- Constructor Summary:** A table titled "Constructors" with one row: "Constructor" (Calcularadora()) and "Description" (Constructor de la clase, obligatorio ponerlo, ya que si no se pone nos da warning).
- Method Summary:** A table titled "All Methods" with three tabs: All Methods (selected), Instance Methods, and Concrete Methods. It lists three methods:

Modifier and Type	Method	Description
int	diveix(int a, int b)	Devolver la división de dos números enteros pasados como parámetro
int	multiplica(int a, int b)	Devolver la mutiplicación de dos números enteros pasados como parámetro
int	resta(int a, int b)	Devolver la resta de dos números enteros pasados como parámetro

The screenshot shows a Java Javadoc interface with the following sections:

- Method Summary**: A table listing methods categorized by modifier and type. It includes four columns: Modifier and Type, Method, and Description. Methods listed include `diveix(int a, int b)`, `multiplica(int a, int b)`, `resta(int a, int b)`, and `suma(int a, int b)`.

Modifier and Type	Method	Description
int	<code>diveix(int a, int b)</code>	Devolver la división de dos números enteros pasados como parámetro
int	<code>multiplica(int a, int b)</code>	Devolver la multiplicación de dos números enteros pasados como parámetro
int	<code>resta(int a, int b)</code>	Devolver la resta de dos números enteros pasados como parámetro
int	<code>suma(int a, int b)</code>	Devolver la suma de dos números enteros pasados como parámetro
- Methods inherited from class java.lang.Object**: A list of methods inherited from the Object class: `clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, and `wait`.
- Constructor Details**: A section for the constructor `Calculadora()`. It contains the code

```
public Calculadora()
```

 and a note: "Constructor de la clase, obligatorio ponerlo, ya que si no se pone nos da warning".
- Method Details**: A section for the method `suma`.

The screenshot shows a Java documentation interface with the following details:

Method Details

suma

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

Devolver la suma de dos números enteros pasados como parámetro

Parameters:

- a - de tipo entero
- b - de tipo entero

Returns:

devuelve la suma de dos números enteros

Throws:

RuntimeException¹² - Tipo de dato no esperado, es decir, un tipo de dato superior al int o diferente al int, tales como String, char etc... No son de obligada captura

resta

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

Devolver la resta de dos números enteros pasados como parámetro

Parameters:

- a - de tipo entero
- b - de tipo entero

Returns:

devuelve la resta de dos números enteros

The screenshot shows a Java documentation interface with the following details:

METHODS

- multiplica**
public int multiplicar(int a, int b)
Devolver la multiplicación de dos números enteros pasados como parámetro
Parameters:
a - de tipo entero
b - de tipo entero
Returns:
devuelve la multiplicación de dos números enteros
Throws:
RuntimeException² - Tipo de dato no esperado, es decir, un tipo de dato superior al int o diferente al int, tales como String, char etc... O dividir entre 0, no son de obligada captura
- divide**
public int divide(int a, int b)
Devolver la división de dos números enteros pasados como parámetro
Parameters:
a - de tipo entero
b - de tipo entero
Returns:
devuelve la división de dos números enteros
Throws:
RuntimeException² - error al introducir un tipo de dato no esperado, es decir, un tipo de dato superior al int o diferente al int, tales como String, char etc... Además de una posible opción, dividir un número entero, esto genera una excepción del tipo ArithmeticException, no es de obligada captura. No son de tipo RunTimeException

The screenshot shows a Java documentation interface with the following details:

- Toolbar:** Includes icons for back, forward, search, and other navigation functions.
- Header:** Shows the URL as `i2_UD6/javadoc/es/cipfpbatoi/ed/practica2_ud6/package-summary.html`.
- Menu Bar:** Contains links for OVERVIEW, PACKAGE, CLASS, TREE, INDEX, and HELP. The PACKAGE link is highlighted.
- Search Bar:** A search input field with placeholder text "Search" and a clear button.
- Content Area:**
 - Section Header:** **Package es.cipfpbatoi.ed.practica2_ud6**
 - Text:** package es.cipfpbatoi.ed.practica2_ud6
 - Table:** A table titled "Classes" showing one class entry.

Class	Description
AppTest	Unit test for simple App.

The screenshot shows a Java documentation page with the following details:

- Header:** Includes navigation icons (back, forward, search, etc.), a title bar with the URL `ts/Practica2_UD6/javadoc/es/cipfpbatoi/ed/practica2_ud6/AppTest.html`, and a search bar.
- Menu Bar:** OVERVIEW, PACKAGE, CLASS (highlighted in orange), TREE, INDEX, HELP.
- Summary Bar:** SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD.
- Package:** Package `es.cipfpbatoi.ed.practica2_ud6`
- Class:** Class `AppTest`
- Superclasses:** `java.lang.Object`
- Code Snippet:**

```
public class AppTest  
extends Object
```
- Description:** Unit test for simple App.
- Constructor Summary:** A table with one row:

Constructors
<code>AppTest()</code>

`AppTest()` Constructor de la clase, obligatorio ponerlo ya que si no nos da warning
- Method Summary:** A table with three tabs: All Methods (selected), Instance Methods, and Concrete Methods.

All Methods	Instance Methods	Concrete Methods
Modifier and Type	Method	Description
void	<code>shouldAnswerWithTrue()</code>	Rigorous Test :-)

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

The screenshot shows a Java documentation interface with the following details:

- Header:** Includes back/forward navigation, a search bar with placeholder "Search", and various window control icons.
- Top Bar:** Navigation links: OVERVIEW, PACKAGE, CLASS (highlighted), TREE, INDEX, HELP.
- Summary Bar:** SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD
- Method Summary Section:** Contains tabs: All Methods (selected), Instance Methods, Concrete Methods. A table lists methods:

Modifier and Type	Method	Description
void	shouldAnswerWithTrue()	Rigorous Test :-)
- Inherited Methods Section:** Lists methods inherited from java.lang.Object: clone(), equals(), finalize(), getClass(), hashCode(), notify(), notifyAll(), toString(), wait(), wait(), wait().
- Constructor Details Section:** Shows the constructor: public AppTest(). A note states: Constructor de la clase, obligatorio ponerlo ya que si no nos da warning.
- Method Details Section:** Shows the method: public void shouldAnswerWithTrue(). A note states: Rigorous Test :-).

6. Añade 3 nuevos métodos a la clase Calculadora

- Muestra **capturas** de:
- **código documentado** (fomatea el código si es necesario)
- directorio Doc y
- la documentación resultante en el navegador

The screenshot shows the IntelliJ IDEA interface. The code editor displays a Java method named `duplicarValor` with detailed Javadoc comments. The run tool window at the bottom shows the command "javadoc" being executed, generating various HTML files for the project's documentation.

```
/**  
 * Duplicar el valor un número de veces  
 *  
 * @param valor de tipo entero, valor a duplicar  
 * @param iteraciones de tipo entero, veces que queramos que se duplique  
 * @return Devuelve el valor duplicado  
 * @throws RuntimeException error al introducir un tipo de dato no esperado, es decir,  
 * un tipo de dato superior al int o diferente al int, tales como Str  
 * No son es obligada captura, por lo tanto son de tipo RunTimeExcept  
 */  
no usages  ↴ Fran Gregori Tandazo  
public int duplicarValor(int valor, int iteraciones) {  
    int resultado = 1;  
    for ( int i = 0; i < iteraciones; i++ ) {  
        resultado *= valor;  
    }  
    return resultado;  
}  
*/
```

Run: Javadoc x
Generating /Users/fran/IdeaProjects/Practica2_UD6/javadoc/index-files/index-7.html...
Generating /Users/fran/IdeaProjects/Practica2_UD6/javadoc/index-files/index-8.html...
Generating /Users/fran/IdeaProjects/Practica2_UD6/javadoc/search.html...
Generating /Users/fran/IdeaProjects/Practica2_UD6/javadoc/overview-summary.html...
Generating /Users/fran/IdeaProjects/Practica2_UD6/javadoc/help-doc.html...
"javadoc" finished with exit code 0

Externally added files can be added to Git
View Files Always Add Don't Ask Again

The screenshot shows the IntelliJ IDEA interface. The left panel displays the project structure for 'Practica2_UD6 [u6p2]'. The 'src/main/java' directory contains a package 'ed.fran' with a class 'Calculadora'. The code editor shows the implementation of the 'duplicarValor' method:

```
no usages ± Fran Gregori Tandazo
public int duplicarValor(int valor, int iteraciones) {
    int resultado = 1;
    for (int i = 0; i < iteraciones; i++) {
        resultado = resultado * valor;
    }
    return resultado;
}
/***
 * Saber si un número es divisible entre 0
 */

```

The right panel shows the run output for 'Javadoc'.

Run: Javadoc

```
Generating /Users/fran/IdeaProjects/Practica2_UD6/javadoc/index-files/index-7.html...
Generating /Users/fran/IdeaProjects/Practica2_UD6/javadoc/index-files/index-8.html...
Generating /Users/fran/IdeaProjects/Practica2_UD6/javadoc/search.html...
Generating /Users/fran/IdeaProjects/Practica2_UD6/javadoc/overview-summary.html...
Generating /Users/fran/IdeaProjects/Practica2_UD6/javadoc/help-doc.html...

"javadoc" finished with exit code 0
```

Externally added files can be added to Git
View Files Always Add Don't Ask Again

The screenshot shows the IntelliJ IDEA interface with the following details:

- Project Structure:** The project is named "Practica2_UD6 [u6p2]". It contains several packages like ".idea", ".vscode", "javadoc", "media", "apartado3", "apartado4", "apartado5", "apartado6", "src", "main", "java", "ed", and "fran".
- Code Editor:** The code is written in Java. The class "Calculadora" has a method "esDivisibleEntre2" that checks if a number is divisible by 2.
- Code Snippet:**

```
105      /**
106      * Saber si un número es divisible entre 0
107      *
108      * @param numero de tipo entero, número que queremos saber si es divisible
109      * @return Devuelve un booleano, true si es divisible o false si no lo es.
110      * @throws RuntimeException error al introducir un tipo de dato no esperado, es decir,
111      *         un tipo de dato superior al int o diferente al int, tales como Str
112      *
113      * No son es obligada captura, por lo tanto son de tipo RunTimeExcept
114      */
115     public boolean esDivisibleEntre2(int numero) {
116
117         boolean esDivisible;
118
119         if ( numero % 2 == 0 ) {
120             esDivisible = true;
121         } else {
122             esDivisible = false;
123         }
124
125         return esDivisible;
126     }
127 }
```
- Toolbars and Status:** The top right shows status icons (0 1 9 212) and a dropdown menu. A tooltip at the bottom right says: "Externally added files can be added to Git View Files Always Add Don't Ask Again".

```
128
129     /**
130      * No se muy bien el funcionamiento de este método
131      *
132      * @param a de tipo entero
133      * @return Devuelve el String, aunque nunca se llega a devolver
134      * @throws RuntimeException error al introducir un tipo de dato no esperado, es decir,
135      *          un tipo de dato superior al int o diferente al int, tales como String, char etc...
136      *
137      */
138     public String oper3(int a) {
139
140         String r = "";
141         int v = a;
142         int v1;
143
144         while (v > 0) {
145             v1 = v % 2;
146             r = v1 + r;
147             v = v / 2;
148         }
149     }
150
Calculadora > oper3()
```

no usages ± Fran Gregori Tandazo

1 Externally added files can be added to Git
View Files Always Add Don't Ask Again

The screenshot shows the IntelliJ IDEA interface with the following details:

- Project Tree:** The project is named "Practica2_UD6 [u6p2]". It contains several packages like ".idea", ".vscode", "javadoc", "resources", "script-dir", "media", and "src". The "src" directory contains files such as ".gitignore", "pom.xml", and "README.md".
- Code Editor:** The code editor displays a Java method named `oper3` with the following code:

```
128
129     /**
130      * No se muy bien el funcionamiento de este método
131      *
132      * @param a de tipo entero
133      * @return Devuelve el String, aunque nunca se llega a devolver
134      * @throws RuntimeException error al introducir un tipo de dato no esperado, es decir,
135      *         un tipo de dato superior al int o diferente al int, tales como Str
136      *
137      */
138     public String oper3(int a) {
139         String r = "";
140         int v = a;
141         int v1;
142
143         while (v > 0) {
144             v1 = v % 2;
145             r = v1 + r;
146             v = v / 2;
147         }
148     }
149
150 }
```
- Status Bar:** Shows icons for 1 error, 9 warnings, 212 code issues, and a search term.
- Bottom Right Corner:** A tooltip message: "Externally added files can be added to Git" with options: "View Files", "Always Add", and "Don't Ask Again".

The screenshot shows a Java documentation interface with the following details:

duplicarValor

```
public int duplicarValor(int valor,
                         int iteraciones)
```

Duplicar el valor un número de veces

Parameters:

- valor - de tipo entero, valor a duplicar
- iteraciones - de tipo entero, veces que queramos que se duplique

Returns:

Devuelve el valor duplicado

Throws:

RuntimeException² - error al introducir un tipo de dato no esperado, es decir, un tipo de dato superior al int o diferente al int, tales como String, char etc... No son es obligada captura, por lo tanto son de tipo RunTimeException

esDivisibleEntre2

```
public boolean esDivisibleEntre2(int numero)
```

Saber si un número es divisible entre o

Parameters:

- numero - de tipo entero, número que queremos saber si es divisible

Returns:

Devuelve un booleano, true si es divisible o false si no lo es.

Throws:

RuntimeException² - error al introducir un tipo de dato no esperado, es decir, un tipo de dato superior al int o diferente al int, tales como String, char etc... No son es obligada captura, por lo tanto son de tipo RunTimeException

The screenshot shows a Java documentation interface with the following details:

Toolbar: Includes icons for back, forward, search, and other navigation.

Header: Shows the path: /fran/IdeaProjects/Practica2_UD6/javadoc/ed/fran/Calculadora.html

Menu Bar: OVERVIEW PACKAGE CLASS TREE INDEX HELP

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

SEARCH: Search bar with a clear button.

Method 1: esDivisibleEntre2

```
public boolean esDivisibleEntre2(int numero)
```

Saber si un número es divisible entre 0

Parameters:
numero - de tipo entero, número que queremos saber si es divisible

Returns:
Devuelve un booleano, true si es divisible o false si no lo es.

Throws:
RuntimeException¹² - error al introducir un tipo de dato no esperado, es decir, un tipo de dato superior al int o diferente al int, tales como String, char etc... No son es obligada captura, por lo tanto son de tipo RunTimeException

Method 2: oper3

```
public String13 oper3(int a)
```

No se muy bien el funcionamiento de este método

Parameters:
a - de tipo entero

Returns:
Devuelve el String, aunque nunca se llega a devolver

Throws:
RuntimeException¹² - error al introducir un tipo de dato no esperado, es decir, un tipo de dato superior al int o diferente al int, tales como String, char etc... No son es obligada captura, por lo tanto son de tipo RunTimeException