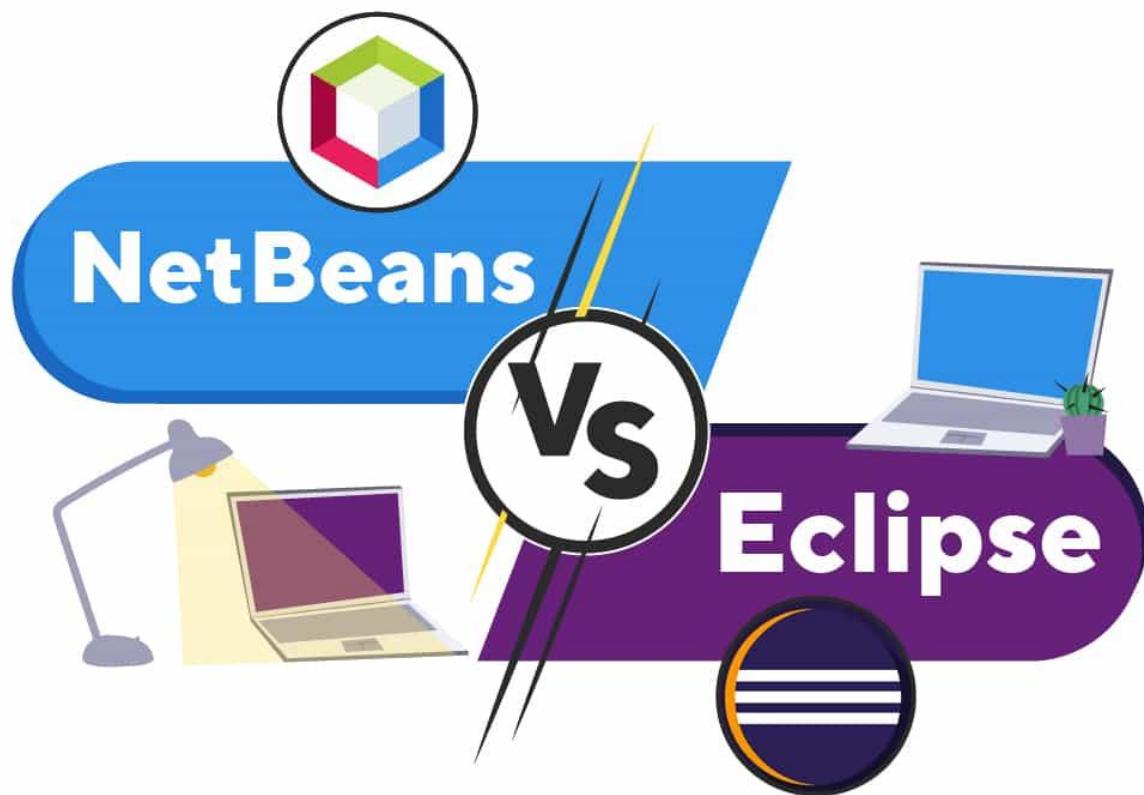


# ENTORNOS DE DESARROLLO

## UNIDAD 2

### TAREA ED02

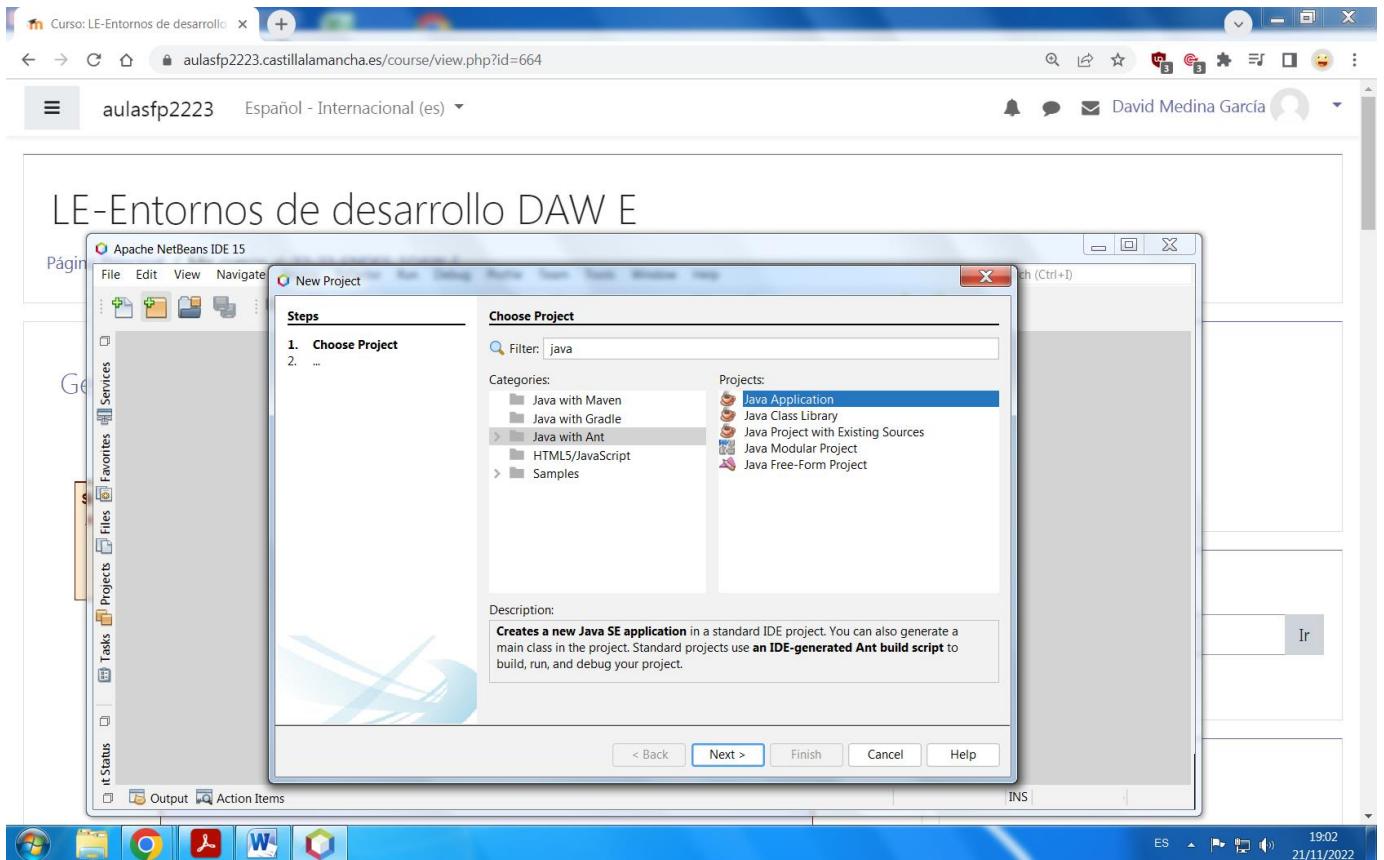


DAVID MEDINA GARCIA

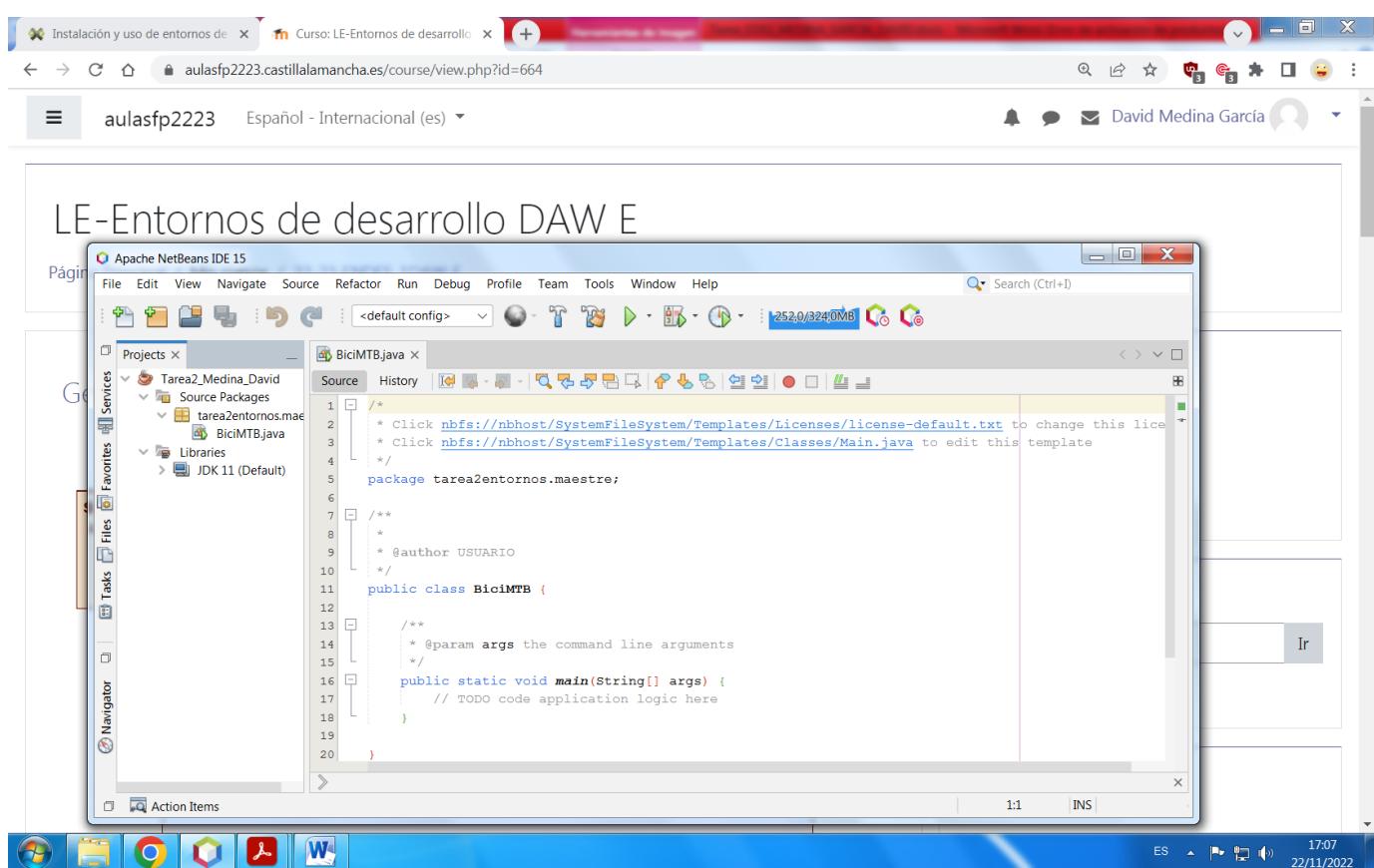
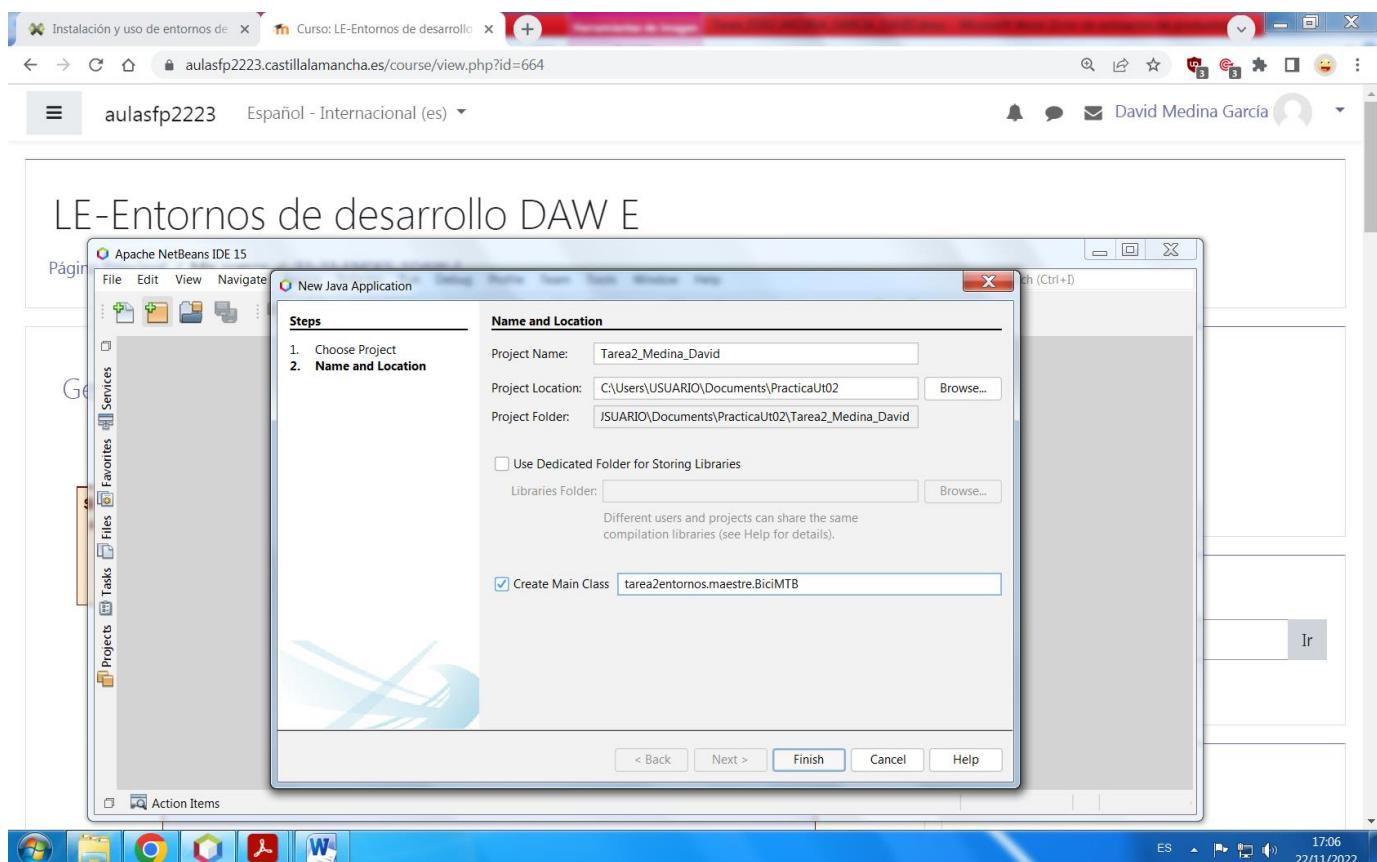
## **1. Creación proyecto**

### **Apartados a/b/c**

Tras habernos instalado Netbeans (en mi caso la version 15) y el JDK (19) ejecutamos Netbeans. Hacemos click en la pestaña **File** y pinchamos en **New Project**. Dentro de la carpeta **Java Ant** seleccionamos **Java Application**.

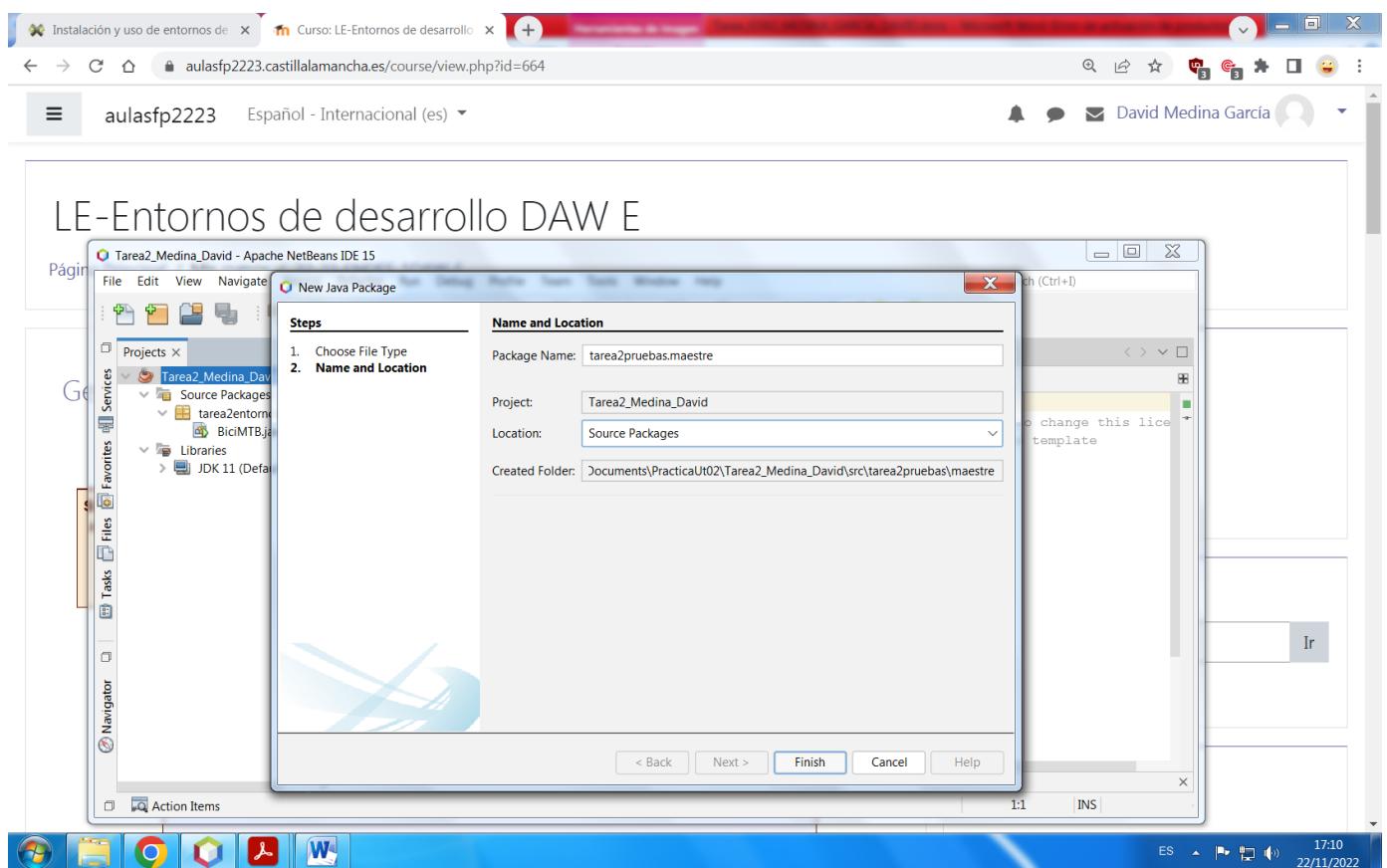


Introducimos el nombre del proyecto, donde se va a ubicar y también damos nombre a la clase **Main**. Tras darle a finish tendremos nuestro proyecto creado.

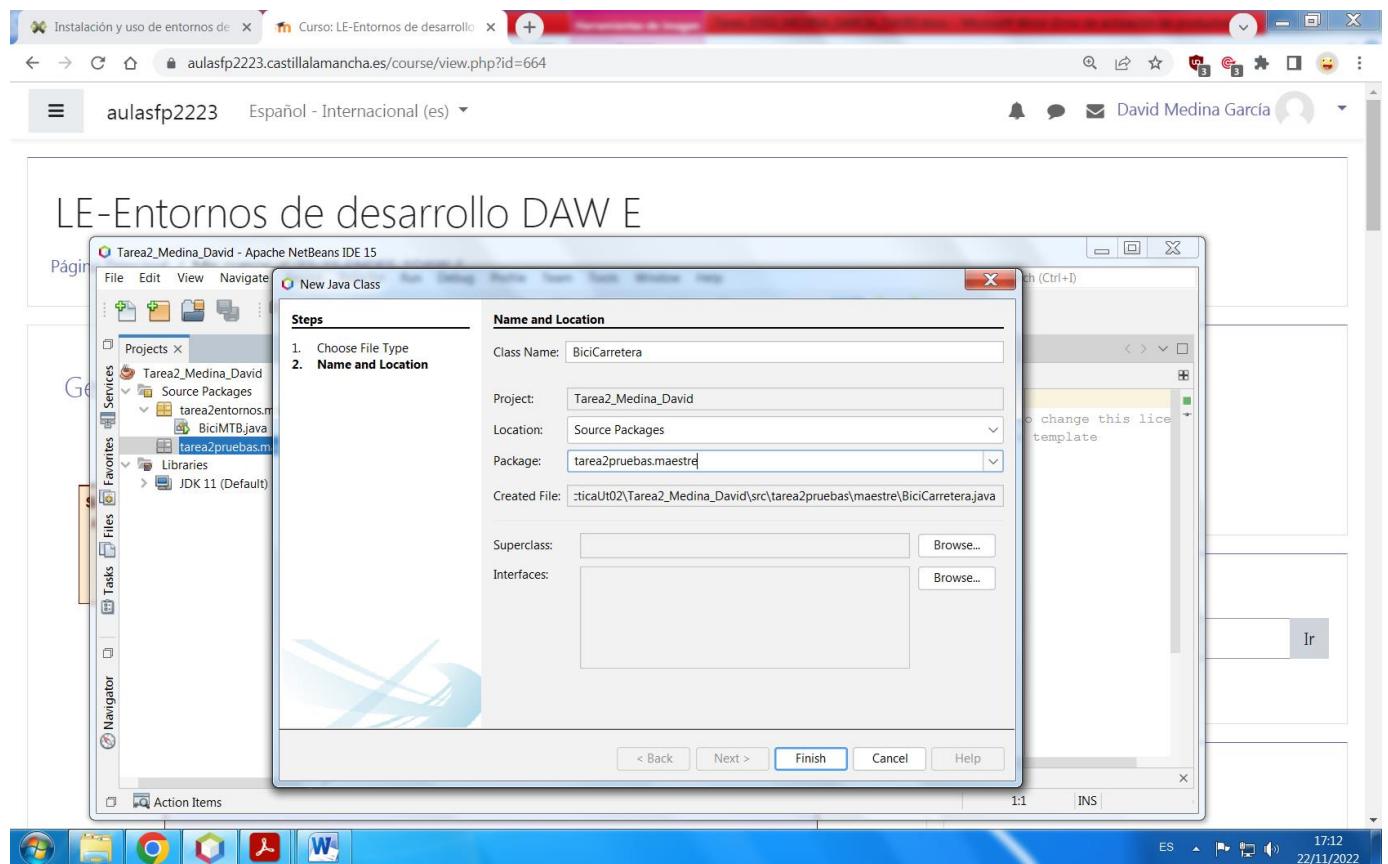


## Apartados d/e

Abrimos la ventana “**Projects**”, hacemos click derecho sobre nuestro proyecto y en **New**, seleccionamos **java package**. Elegimos nombre y ubicación y finalizamos.



Abrimos la ventana “**Projects**”, hacemos click derecho sobre nuestro proyecto y en **New**, seleccionamos **java class**. Elegimos nombre y ubicación y finalizamos.

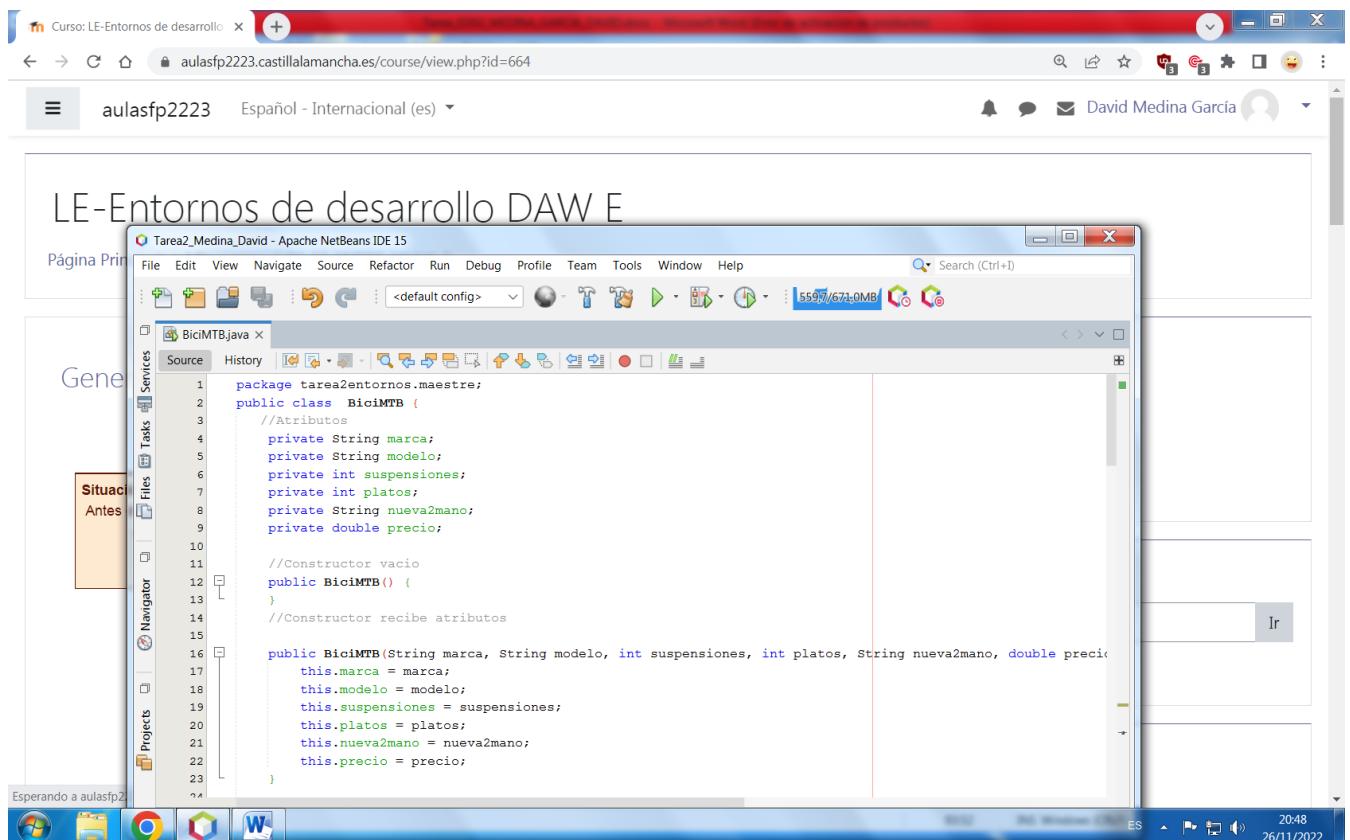


## Apartado f/g

Establecemos las variables que vamos a utilizar. Usamos **String** para cadenas de texto, **int** para números enteros, y **double** para número con dos decimales.

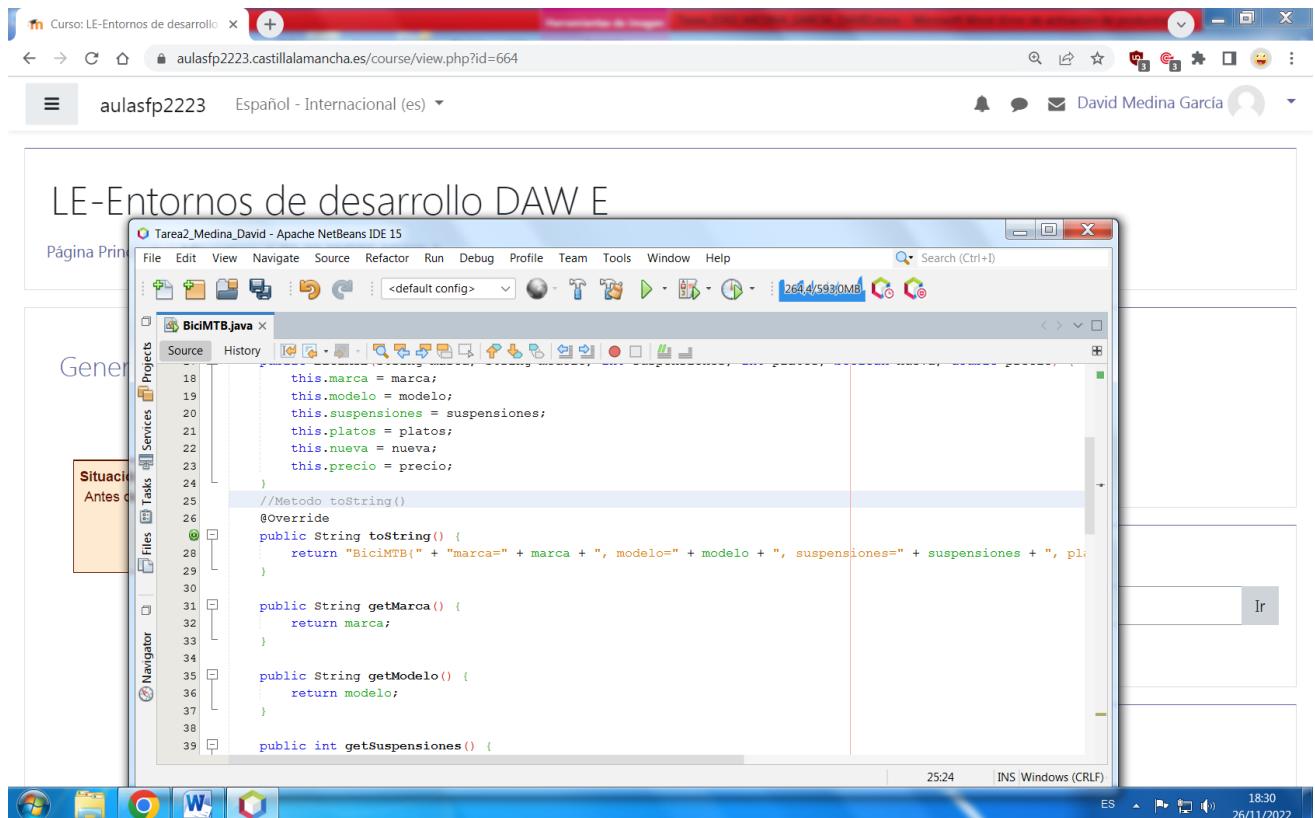
El constructor vacío lo creamos dando click derecho (o **alt + insert**) seleccionamos constructor y finalizamos sin marcar nada.

El constructor “relleno” lo creamos dando click derecho (o **alt + insert**) seleccionamos constructor y marcamos lo que deseamos que recoja el constructor, en este caso todo.



## Apartado h

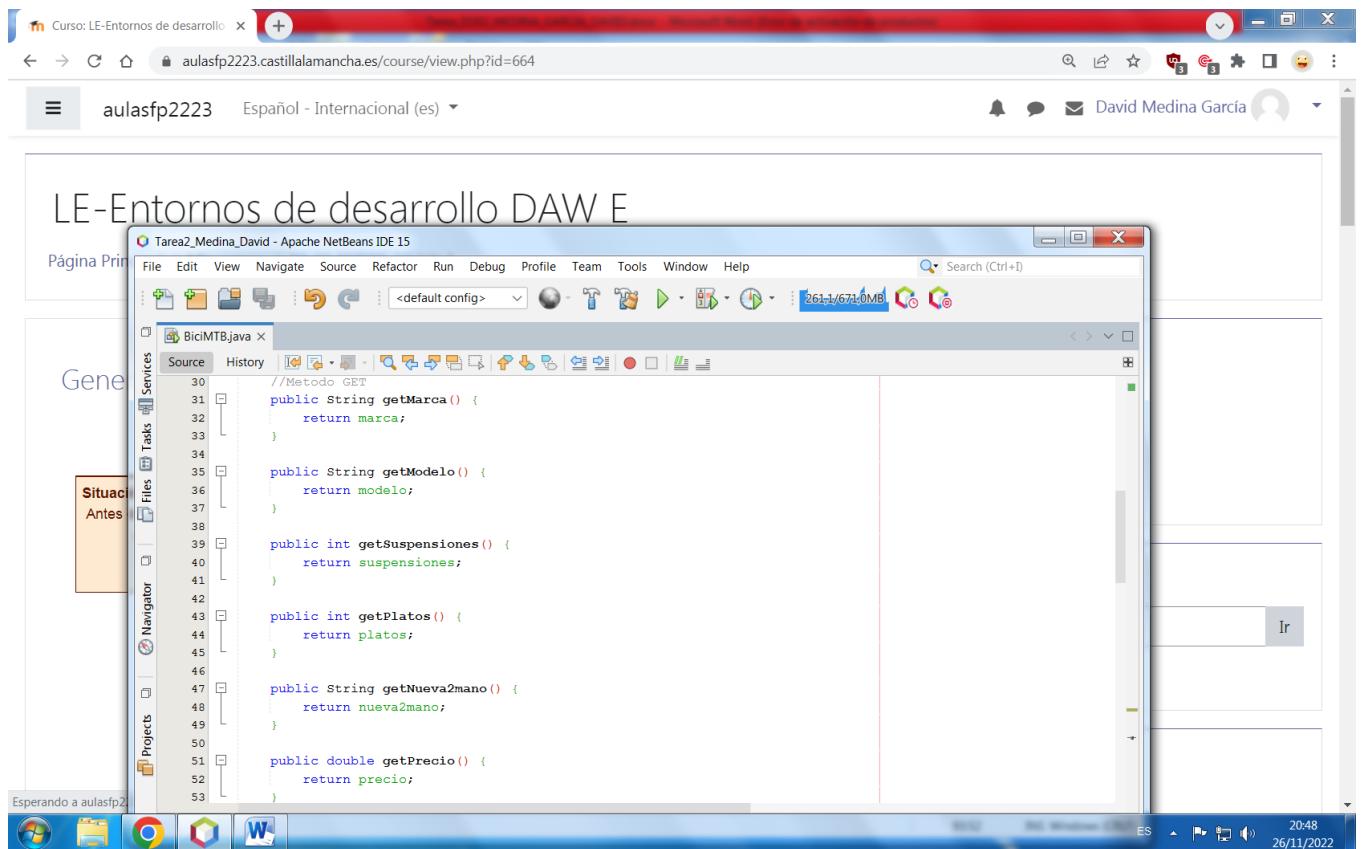
El método **toString ()** lo creamos dando click derecho (o **alt + insert**) seleccionamos **toString()** y marcamos lo que deseamos que recoja el constructor, en este caso todo.



## Apartado i

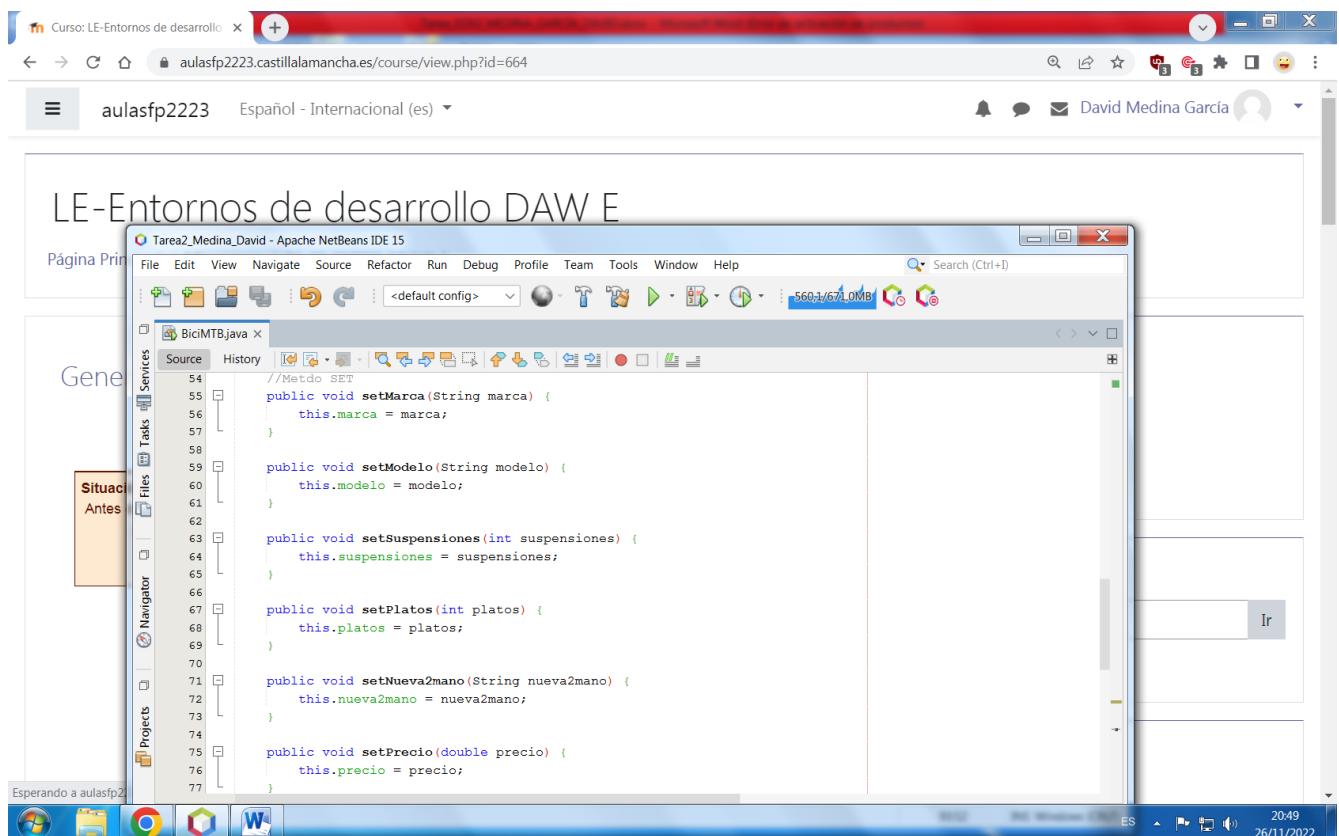
### Método GET

El método **GET** lo creamos dando click derecho (o **alt + insert**) seleccionamos **getter** y marcamos lo que deseamos que recoja el método, en este caso todo. También marcamos la casilla de **encapsular**.



## Método SET

El método **SET** lo creamos dando click derecho (o **alt + insert**) seleccionamos **setter** y marcamos lo que deseamos que recoja el método, en este caso todo. También marcamos la casilla de **encapsular**.

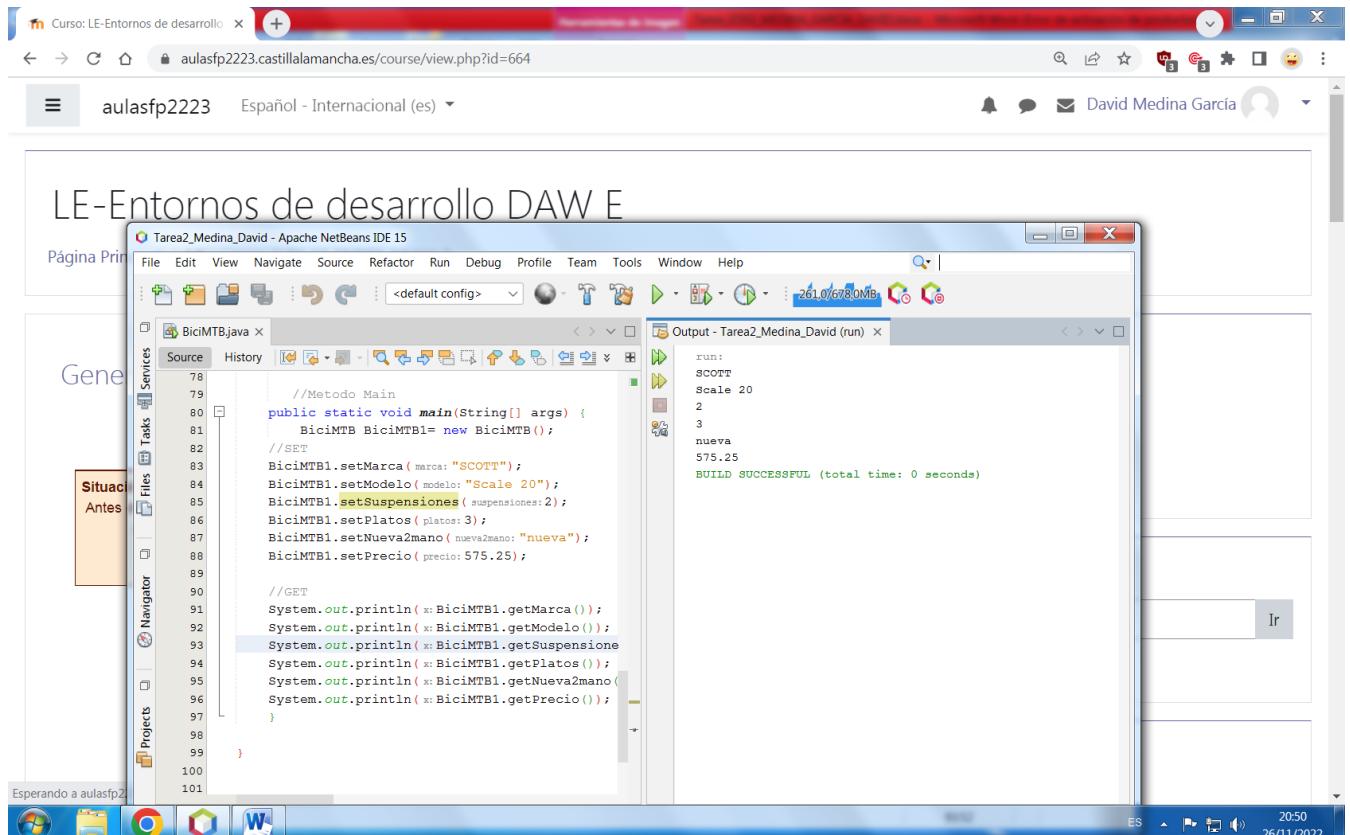


## Apartado j

Creamos el método **main**, escribiendo **main+TAB**. Creamos el objeto **BiciMTB** dándole un nombre y e igualándolo con el comando **new**.

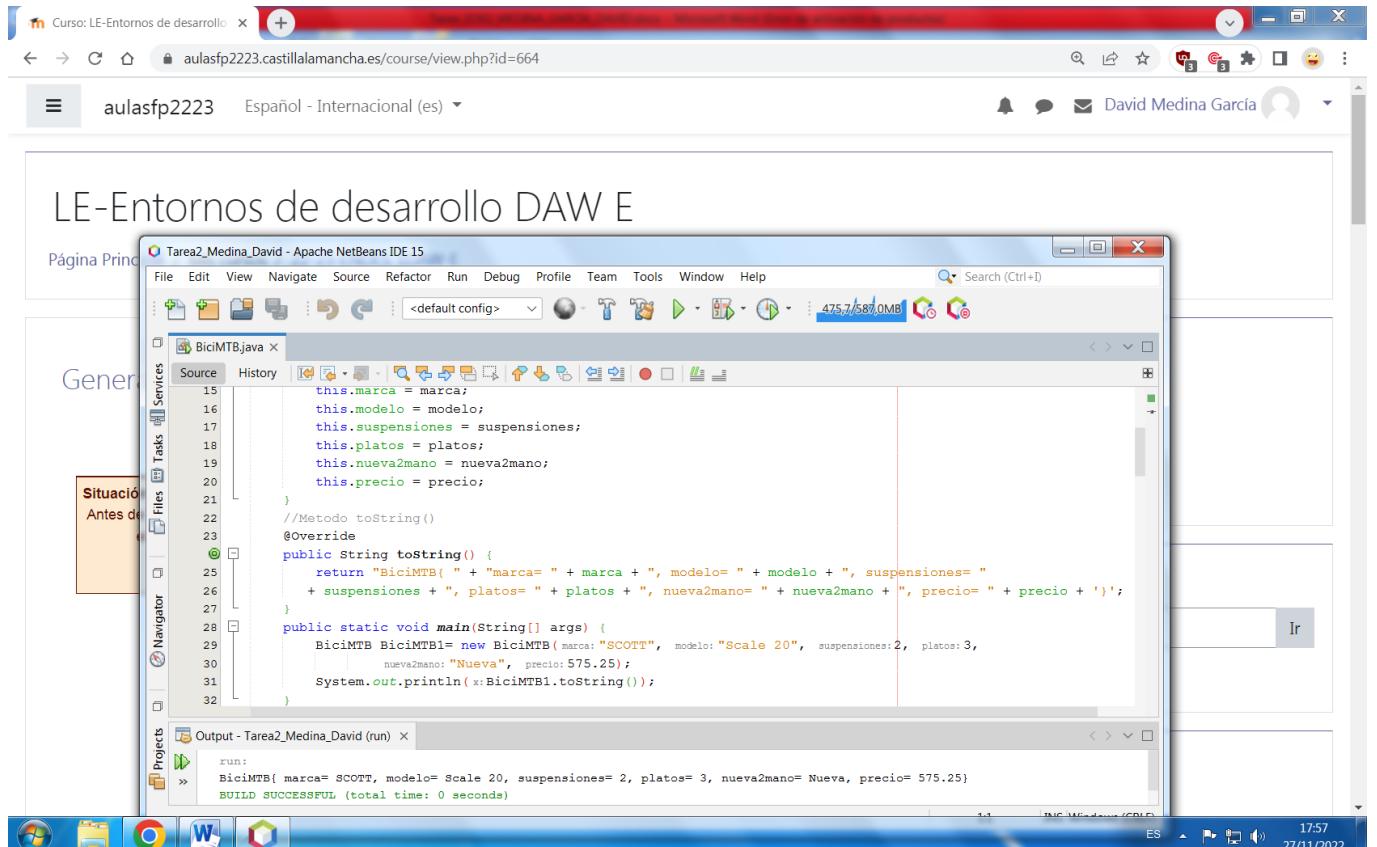
Con los **sets** establecemos los valores de los atributos y con **get** obtenemos dichos valores.

Ejecutamos (**click derecho/Run file**) para ver el resultado en consola.



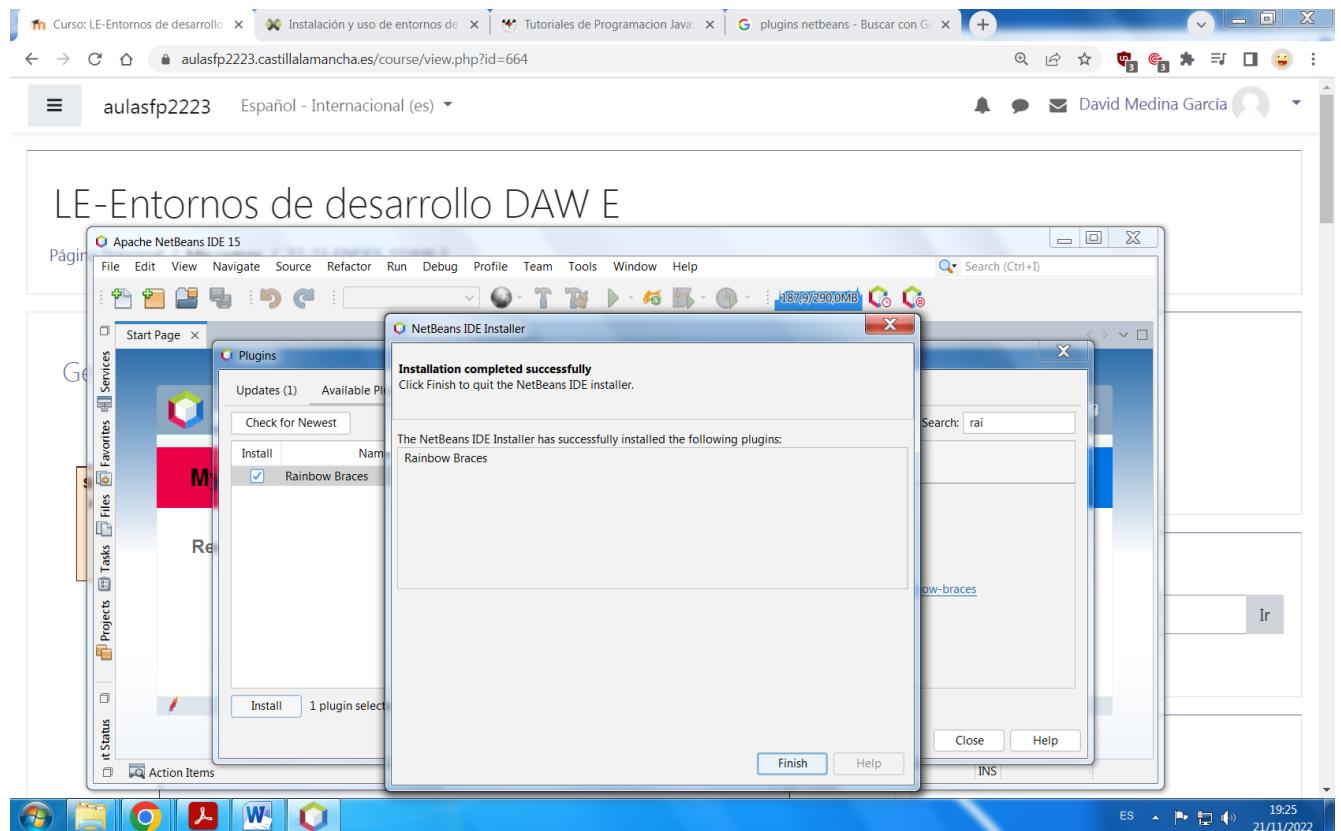
## Apartado k

Usando el método **toString** del apartado h, mostraremos el resultado por consola. Para ello creamos el método **main** con un objeto llamado **BiciMTB1**, al cual le introduciremos los valores de las variables previamente definidas. Utilizamos el **SOUT (System.out.println)** para llamar al método **toString (BiciMTB.toString)** y ejecutamos dándole al símbolo ➤.

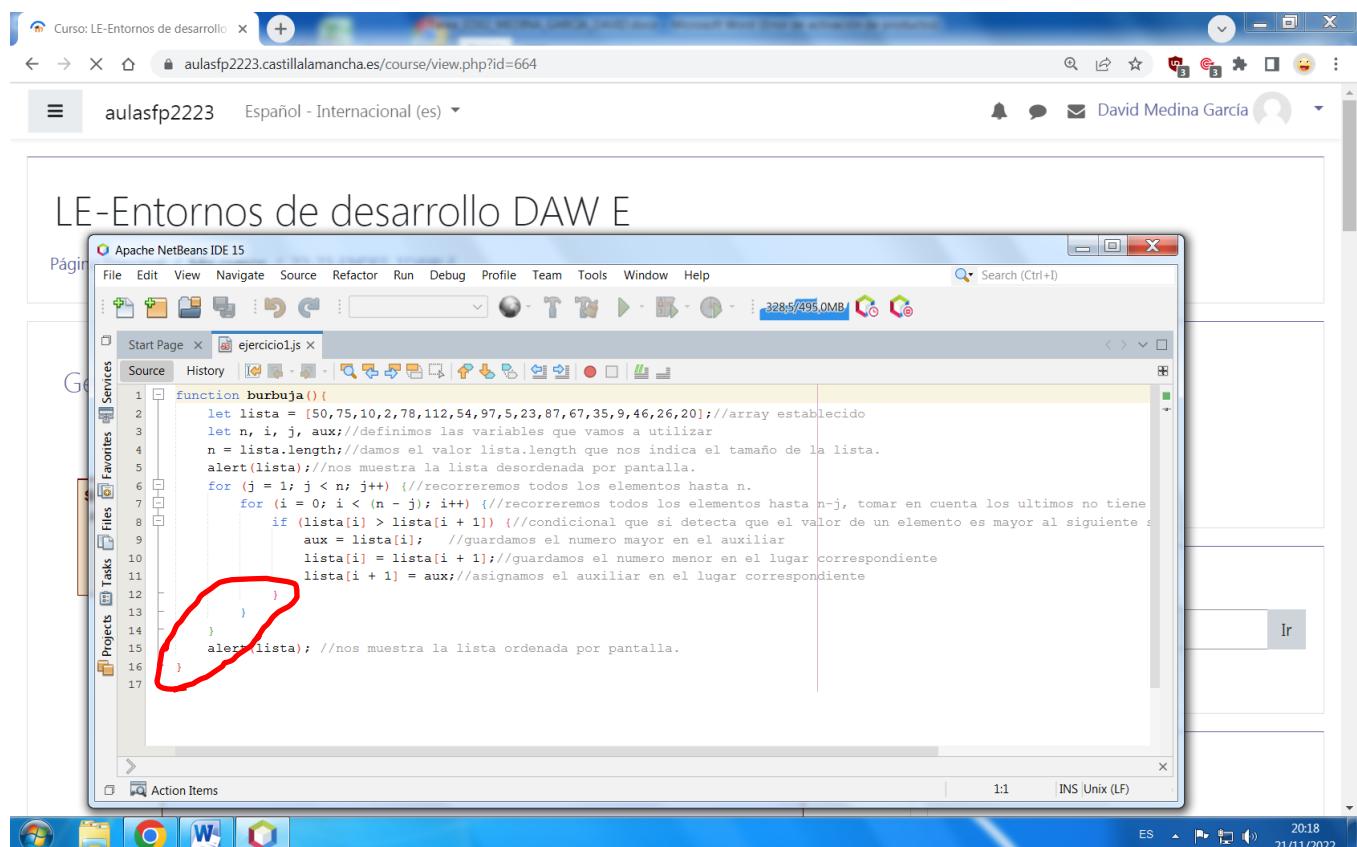


## **2. Instalación plugin Rainbow Braces (On-line)**

Para instalar un plugin online, abrimos la pestaña **tools** y seleccionamos **plugins**. Buscamos el que queramos instalar y procedemos a ello. Tambien podemos seleccionar los que nos interesen de la lista que hay en **available** plugins, la cual se puede actualizar.

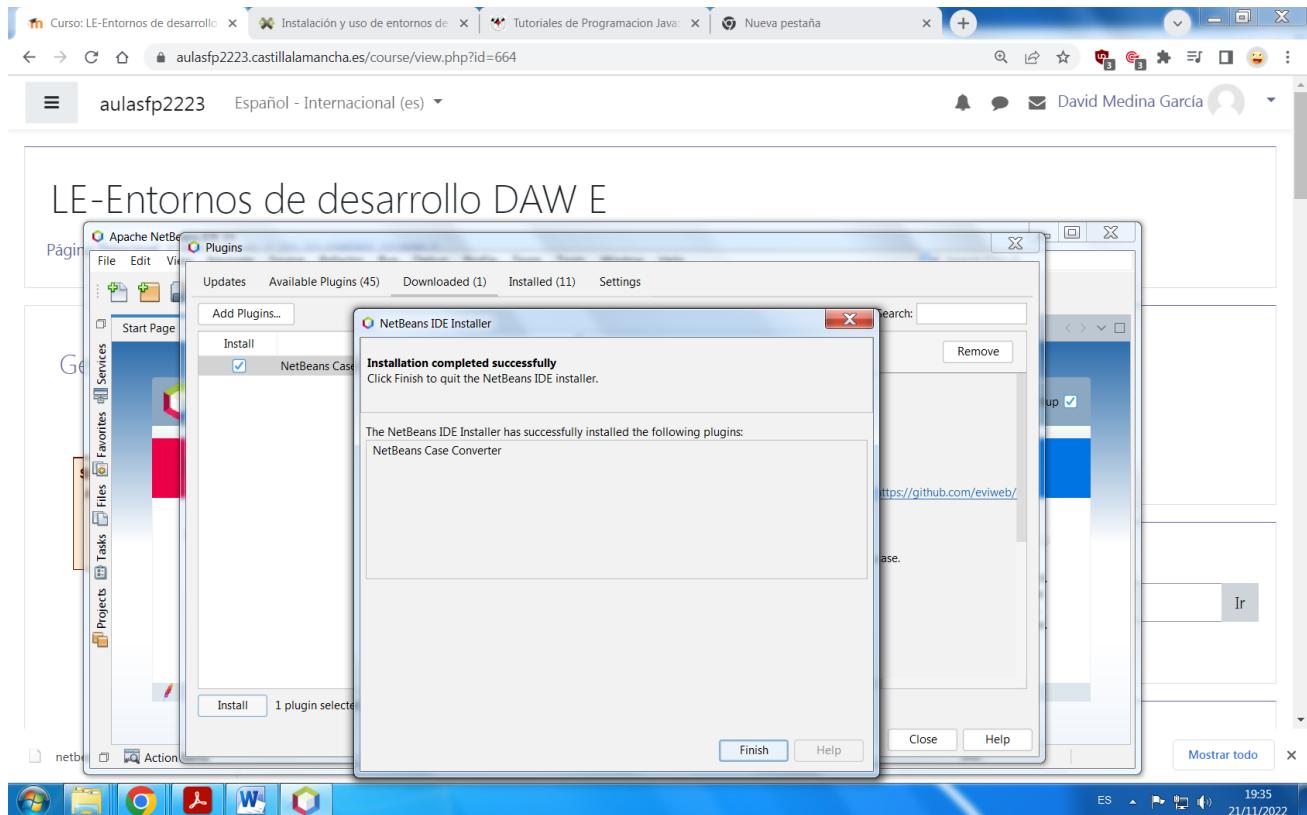


Este plugin basicamente sirve para resaltar de distintos colores los corchetes del codigo ayudando asi a distinguir mejor los distintos comandos establecidos, facilitando asi su revision y corrección de errores.

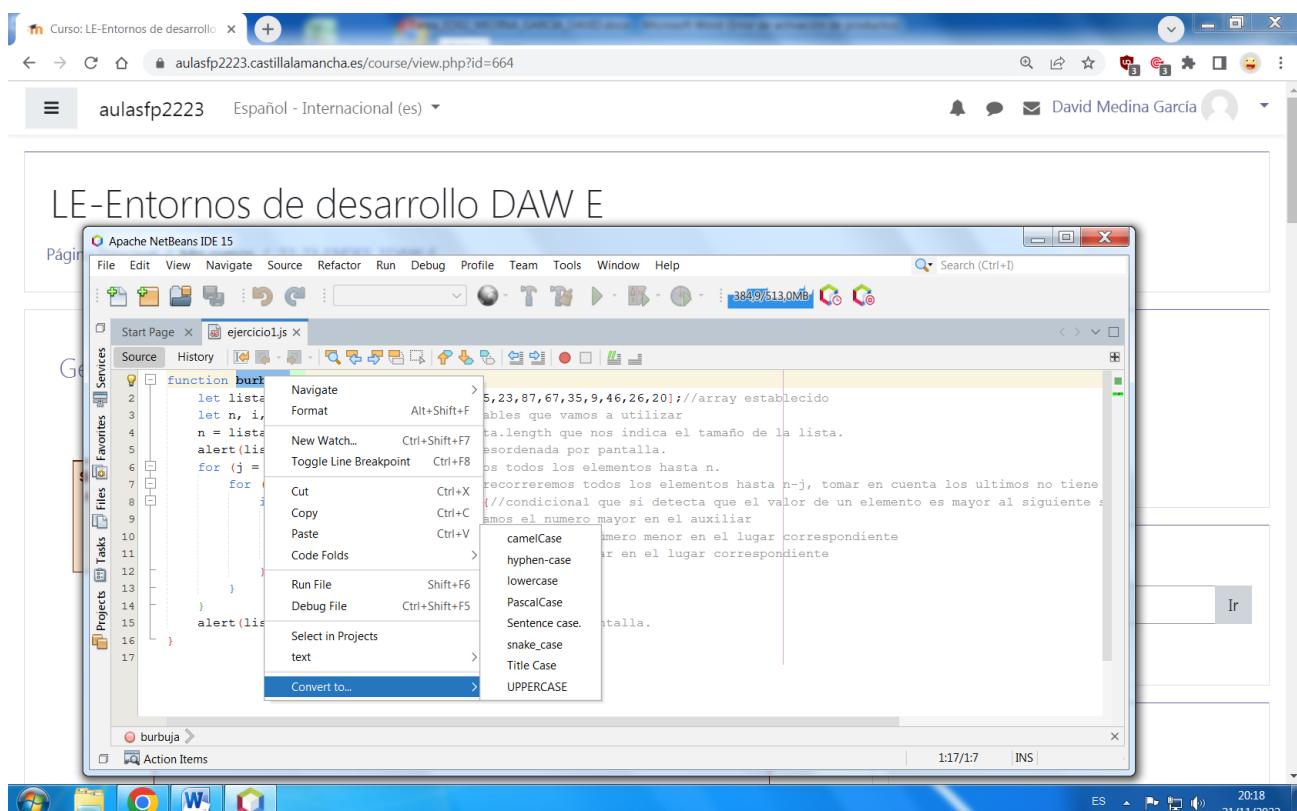


### **3. Instalación plugin Netbeans Case Converter (Off-line)**

Para instalar un plugin de forma offline vamos a la página [plugins.netbeans.apache.org](https://plugins.netbeans.apache.org) (tambien se puede buscar en otros sitios) y buscamos el que necesitemos. Hacemos click en él. Seleccionamos la versión de **Netbeans IDE** que tenemos y lo descargamos. Una vez descargados volvemos a netbeans y nos dirigimos a **tools > plugins**. Hacemos click en la pestaña **Downloaded** y hacemos click en **Add plugin**. Buscamos el archivo descargado y lo instalamos siguiendo el asistente de instalación.



Este plugin sirve para intercambiar facilmente entre mayúsculas y minúsculas, letra Capital y demás modalidades.

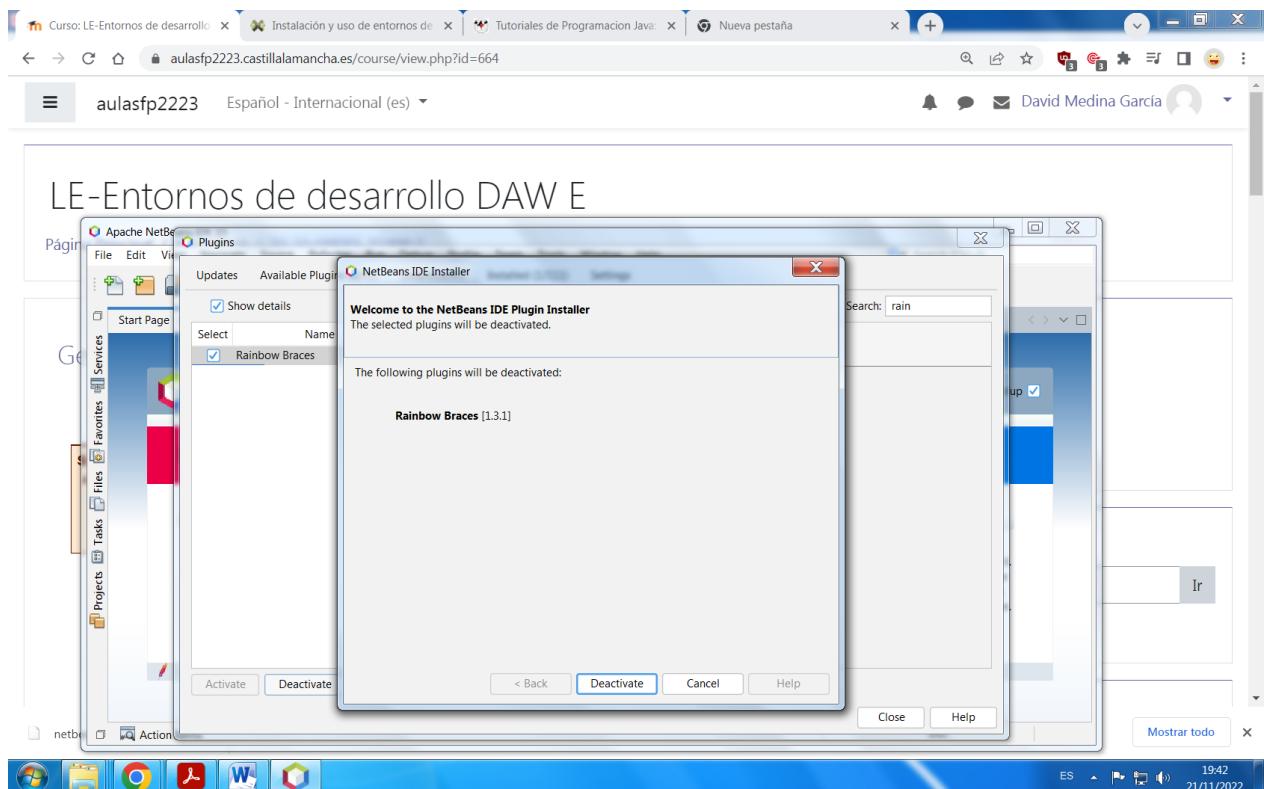


#### **4. Desactivar y desinstalar plugins**

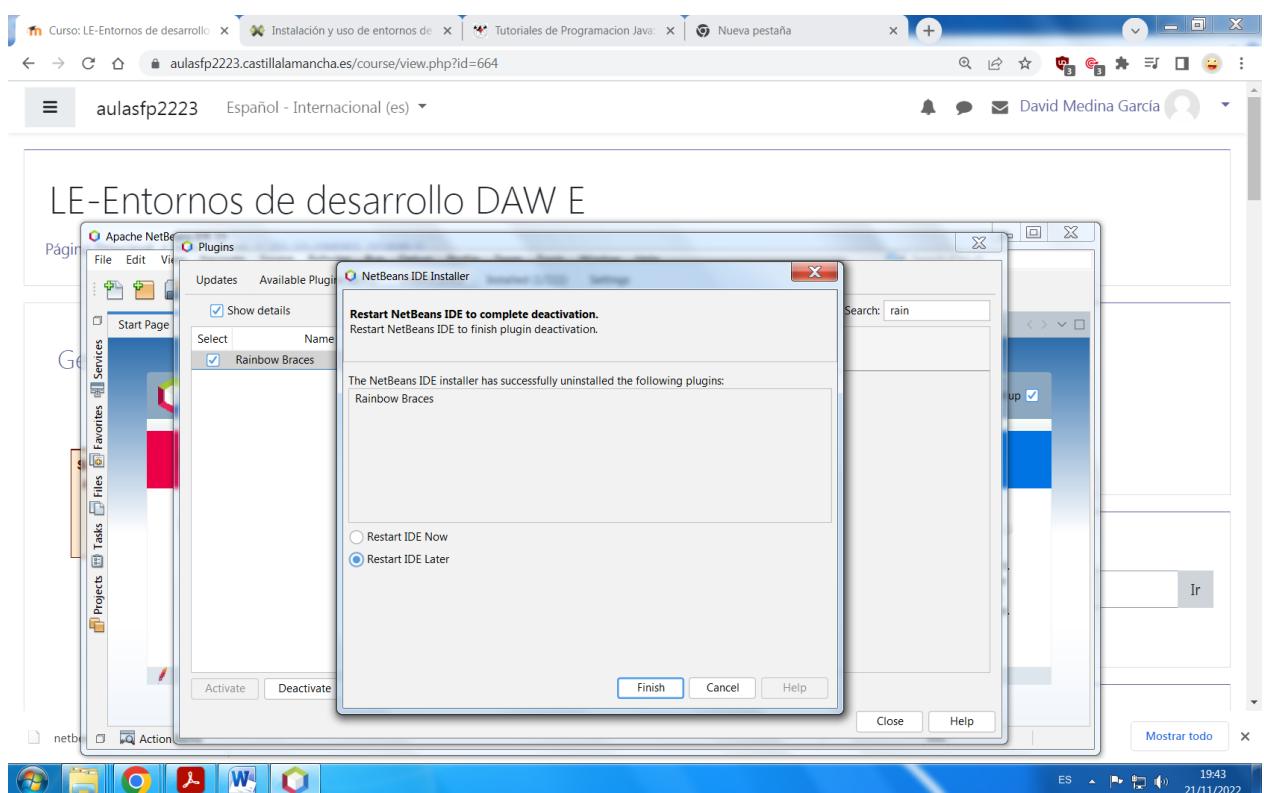
Para desactivar y desinstalar plugins tenemos que irnos a la pestaña **tools** y abrir **plugins**. Allí buscamos el plugin sobre el que queremos hacer la opción y procedemos a ello. Ambas operaciones se hacen de igual manera, solo que si desinstalamos nos pedirá reiniciar el programa para establecer los cambios.

##### **Rainbow Braces**

- Desactivar

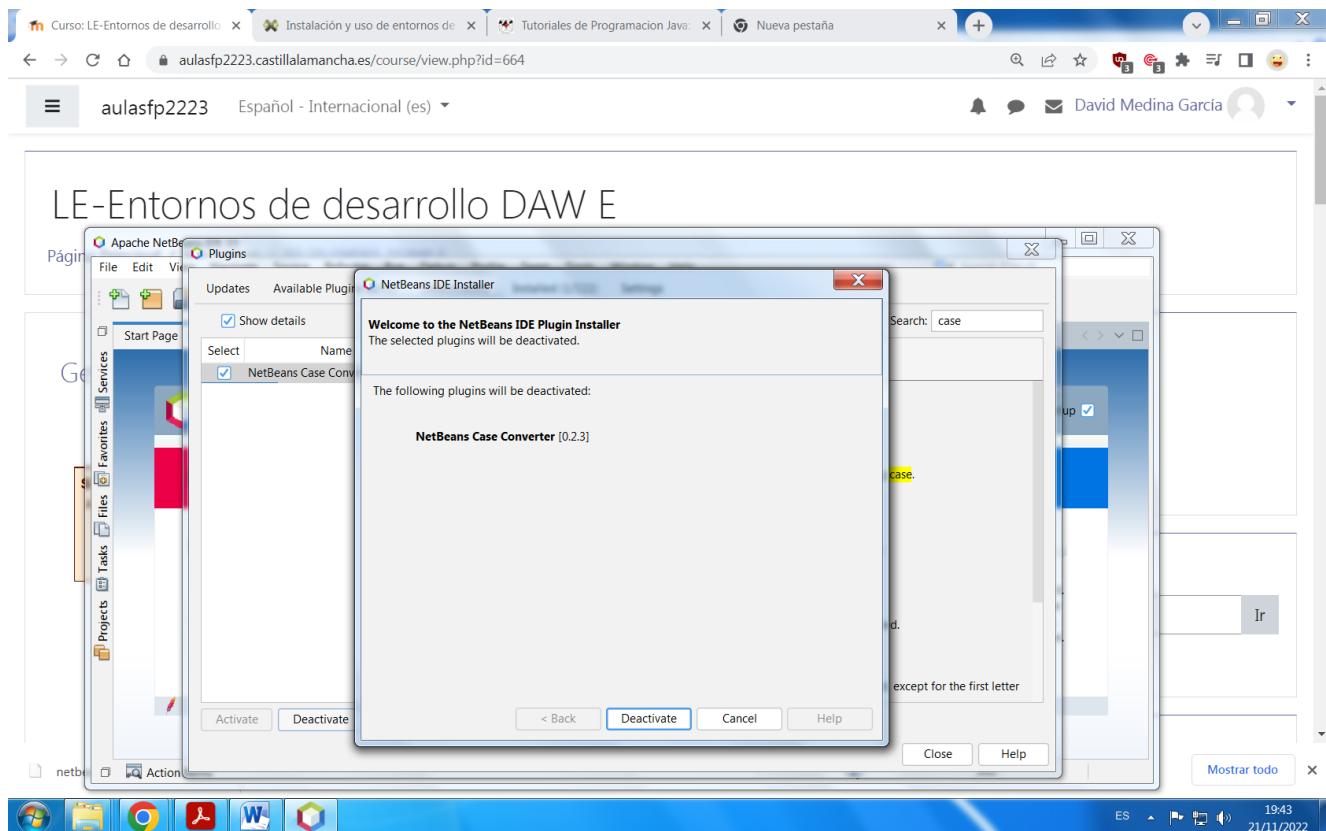


- Desinstalar

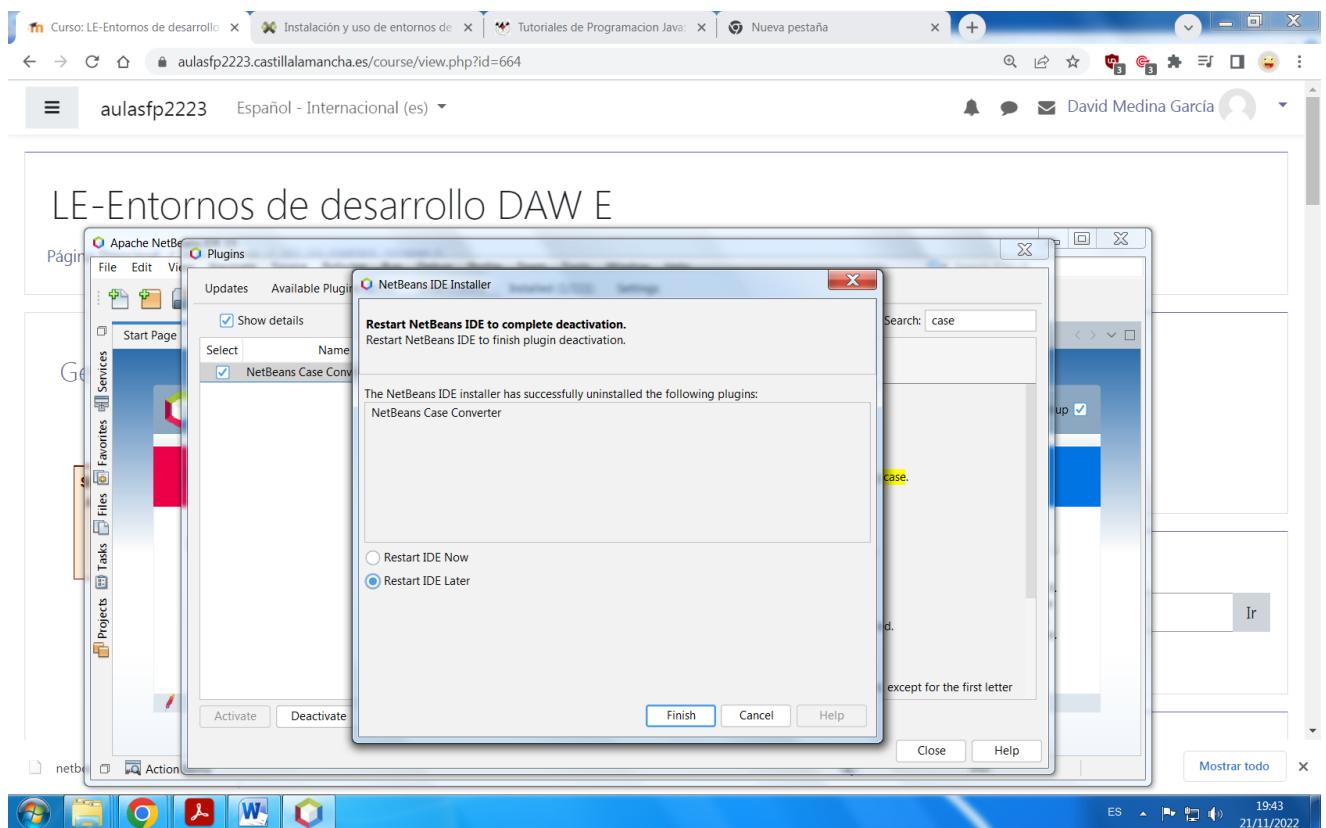


## Netbeans Case Converter

- Desactivar

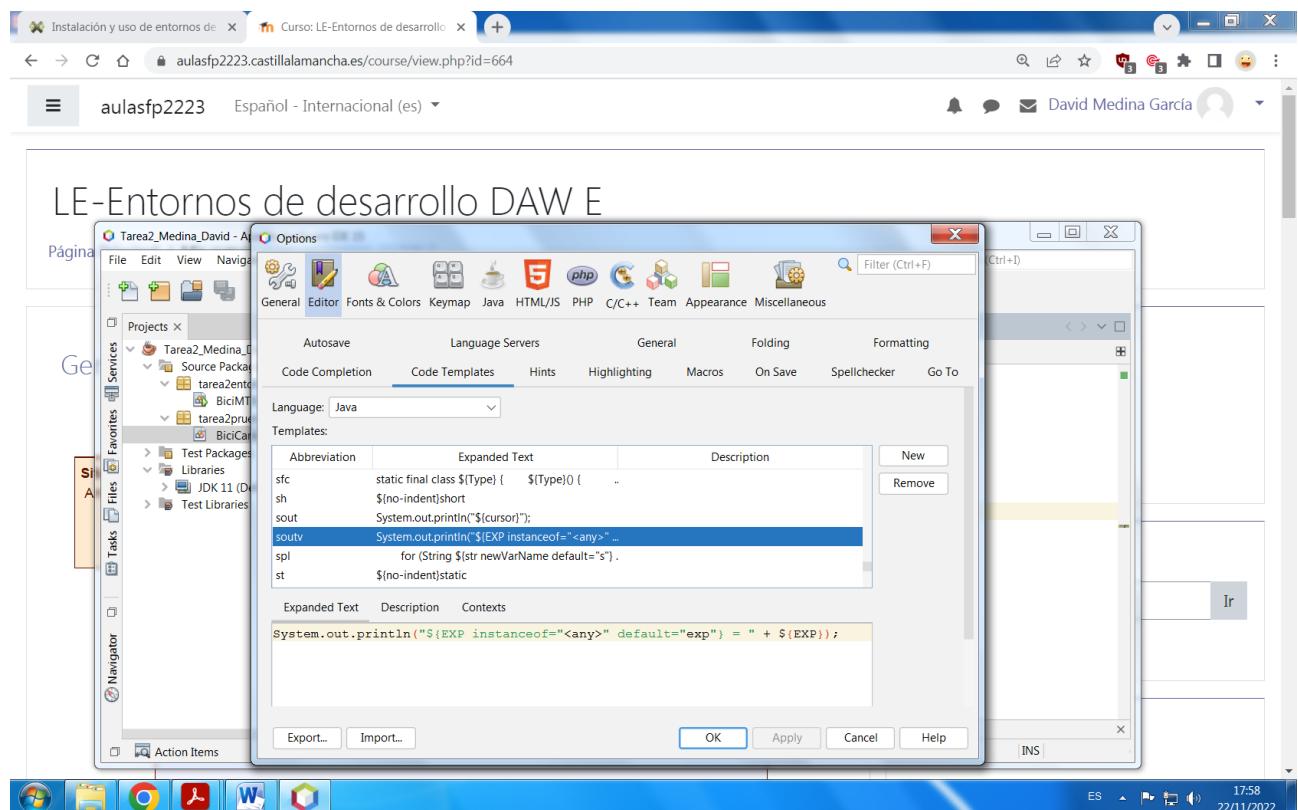


- Desinstalar

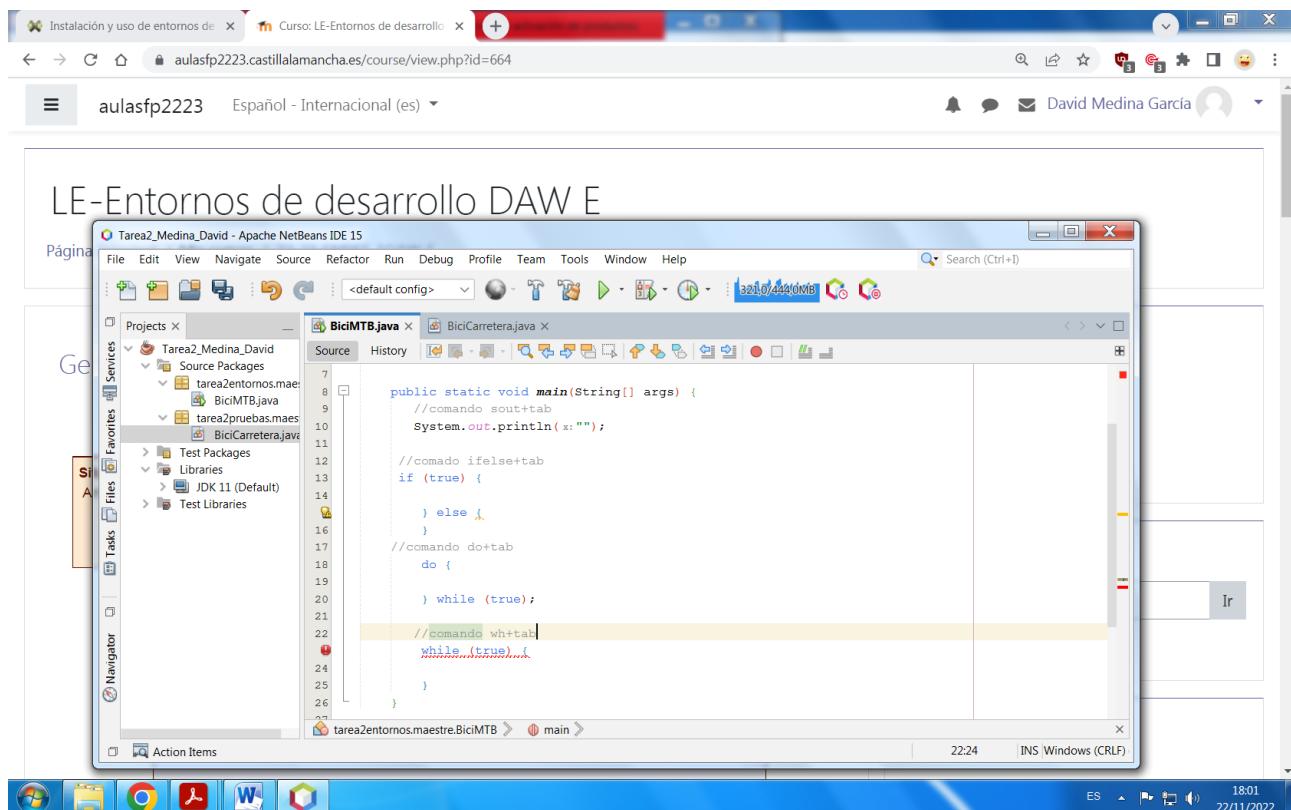


## 5. Atajos del teclado

Si queremos ver los distintos atajos de teclado que abrir la pestaña **tools** y allí irnos a **options**. Nos abrirá una ventana con varios apartados. Dentro del apartado **Editor** nos vamos a la **pestaña Code Templates** y allí podemos ver los distintos atajos que tienen los programas de netbeans.

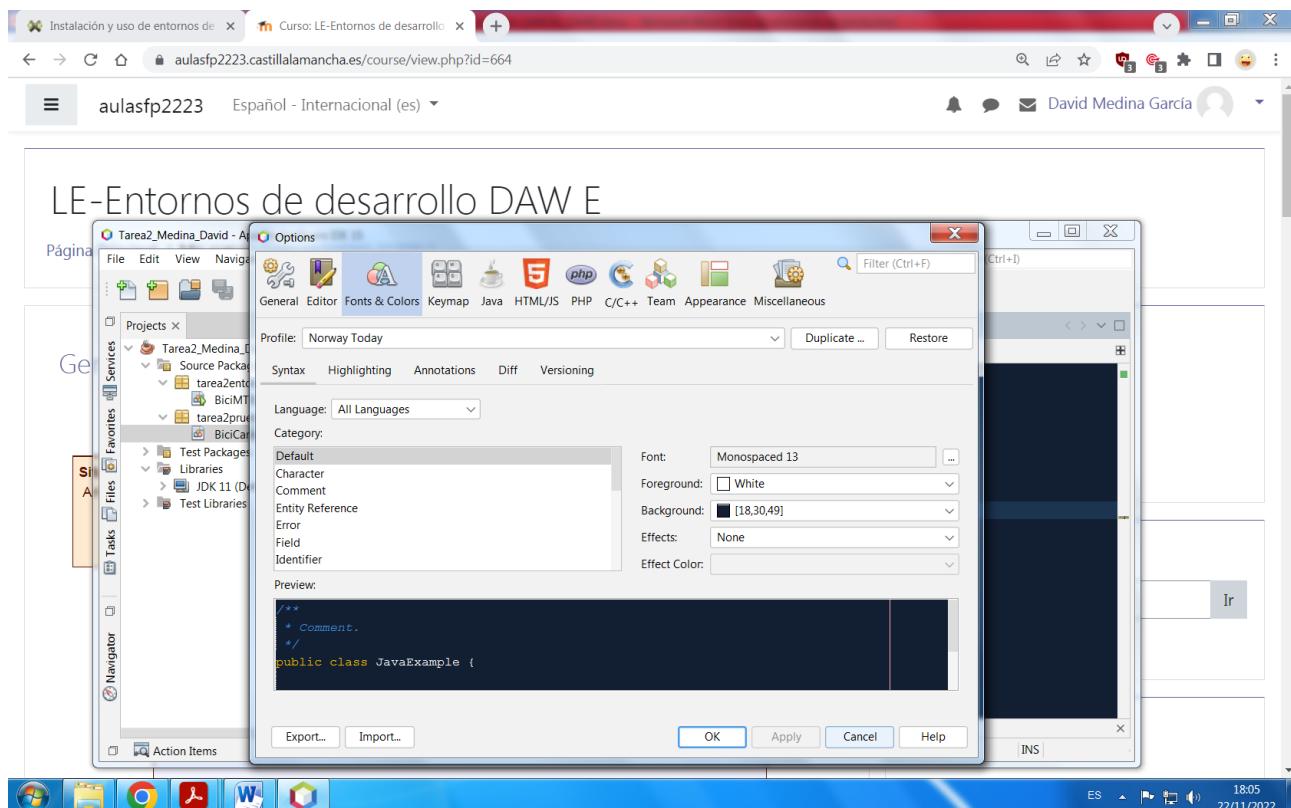


## Apartados a/b/c/d



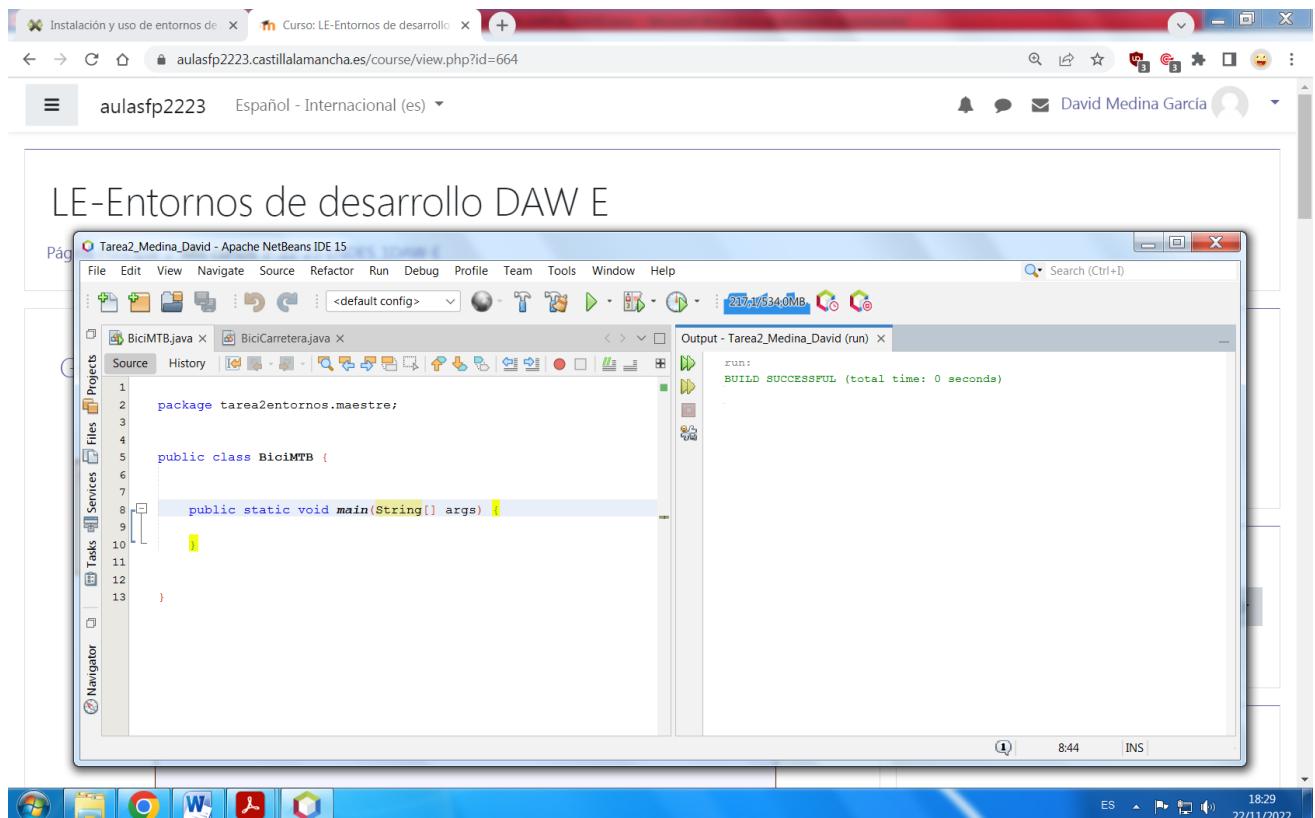
## 6. Cambiar el editor a “Norway today”

Para modificar el editor tenemos que abrir la pestaña **tools** y allí irnos a **options**. Nos abrirá una ventana con varios apartados. Dentro del apartado **FONT & COLORS** podemos elegir el **perfil** que queramos y modificarlo a nuestro gusto.



## 7. Configurar entorno

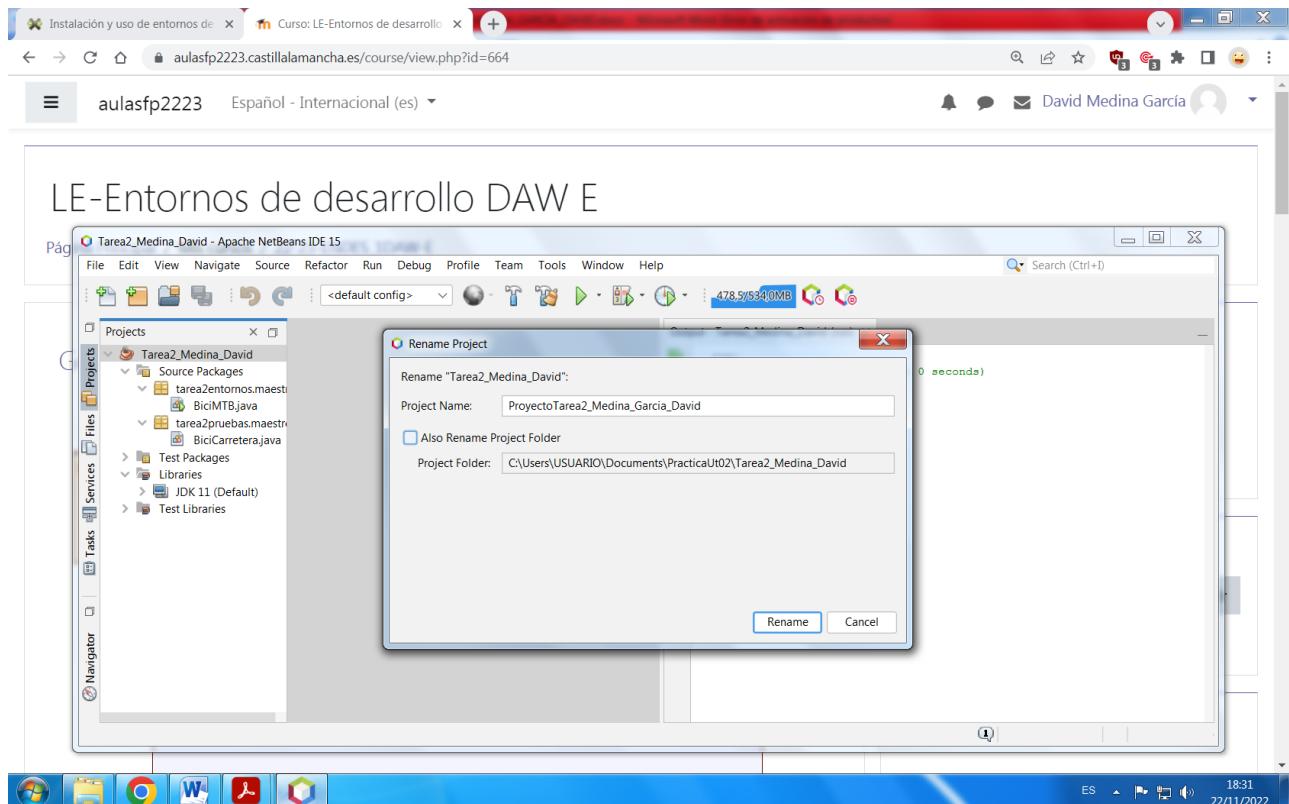
Para conseguir que el editor se disponga como la imagen de abajo, debes clicar y mantener pulsado la ventana de **Output**, arrastrarla hacia las otras ventanas abiertas y cuando nos aparezca una silueta de ventana en vertical soltar la ventana para que se queden divididas verticalmente. Si la prefieres horizontal, lo mismo pero cuando nos salga la silueta horizontal. También puedes modificar las ventanas minimizadas que te aparecen a izquierda y derecha de las principales. Añadir o quitar por ejemplo.



## 8. Operaciones sobre el proyecto

### Apartado a

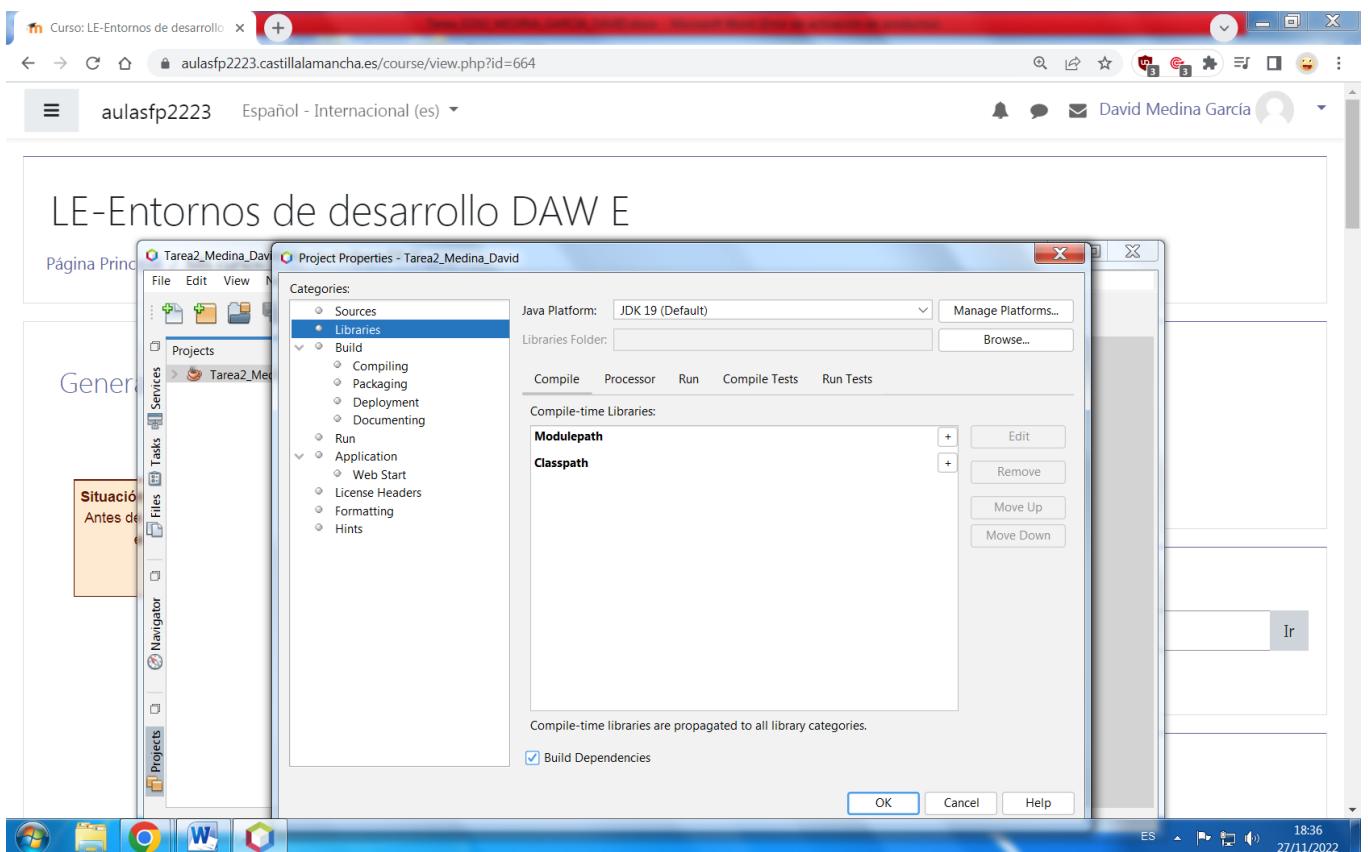
Cambiar el nombre al proyecto es muy sencillo, solo tienes que desplegar la pestaña **Projects**, que se sitúa a la izquierda, hacer click derecho sobre nuestro proyecto y seleccionar **rename**.



## Apartado b

Para entrar a las propiedades del proyecto, abrimos la venta **Projects**, click derecho a nuestro proyecto, y seleccionamos **properties**. Hay diversas categorías, nosotros vamos a centrarnos en cuatro como ejemplo, serán **Libraries**, **Compiling**, **Packaging** y **Run**.

- **Libraries**

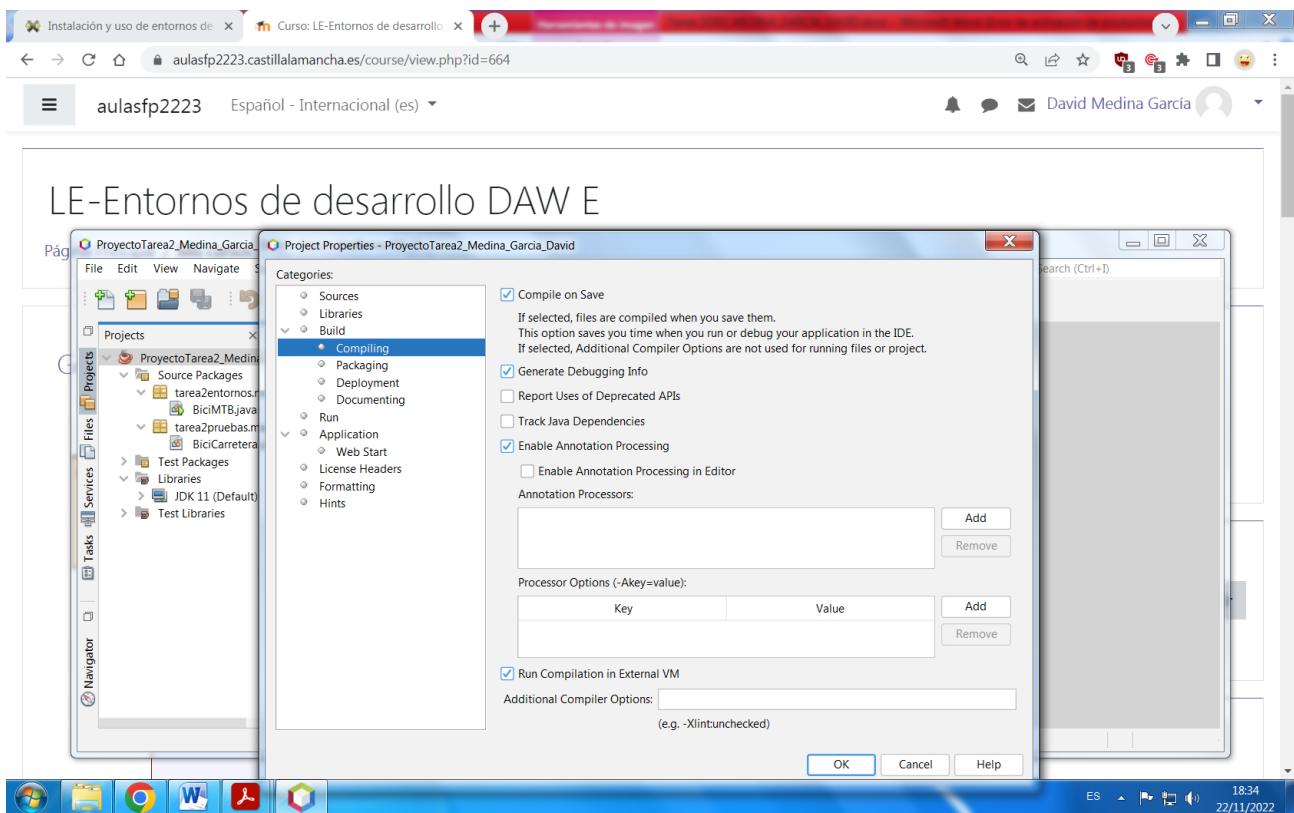


Aquí podemos agregar las **librerías** que nos hagan falta para ayudarnos con nuestros proyectos.

Las **librerías** en programación son aquellos paquetes creados por terceros que podemos agregar a nuestros proyectos para enriquecer el sistema o aprovechar ciertas funcionalidades.

Una **librería** en Java se puede entender como un conjunto de clases, que poseen una serie de métodos y atributos. Lo realmente interesante de estas librerías para Java es que facilitan muchas operaciones. De una forma más completa, las librerías en Java nos permiten reutilizar código, es decir que podemos hacer uso de los métodos, clases y atributos que componen la librería evitando así tener que implementar nosotros mismos esas funcionalidades.

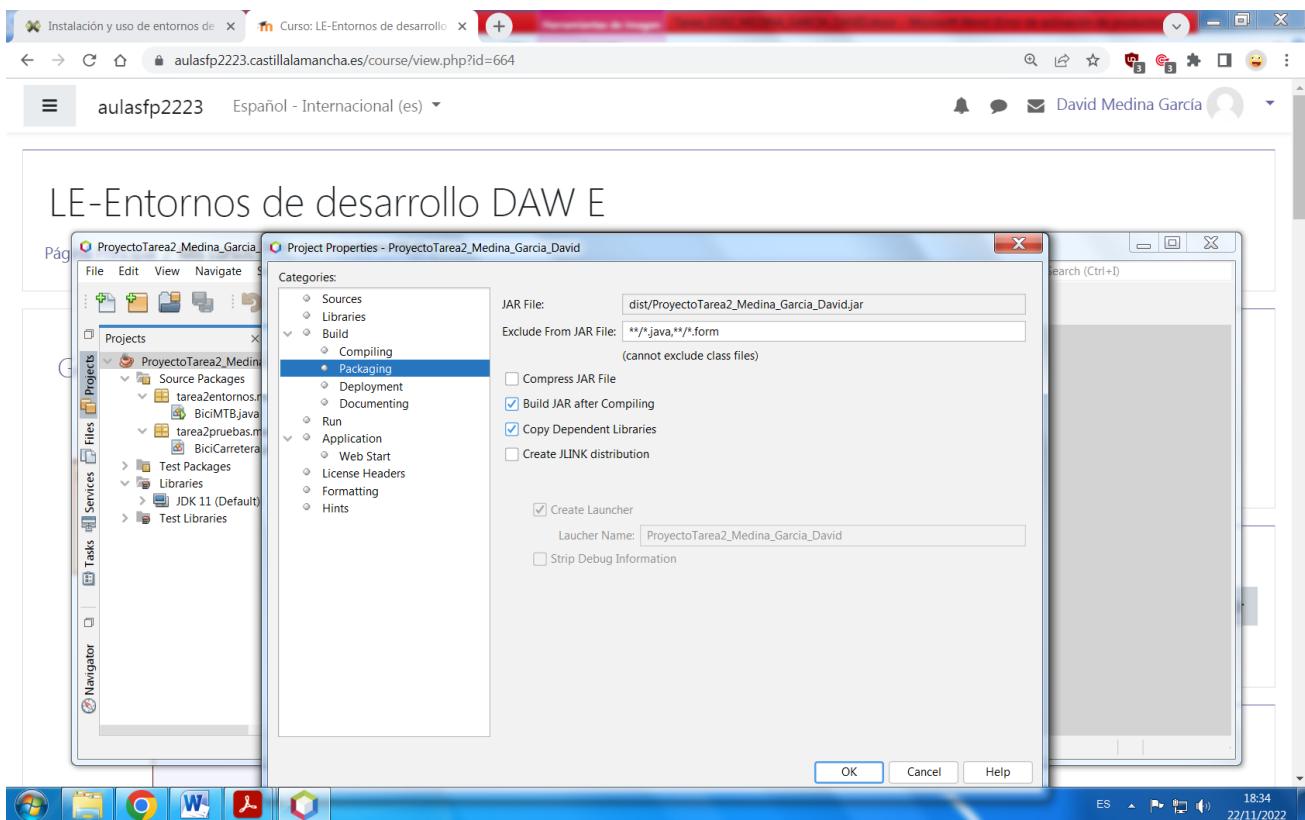
- **Compiling**



En este apartado podemos configurar un poco la **compilación** de nuestro código. Por ejemplo elegir si queremos que se compile el código cada vez que se guarde.

**Compilar** es el proceso que traduce un lenguaje de programación, por ejemplo un código java, al lenguaje de máquina que solo entiende el procesador. Es decir un código fuente a un ejecutable.

- **Packaging.**

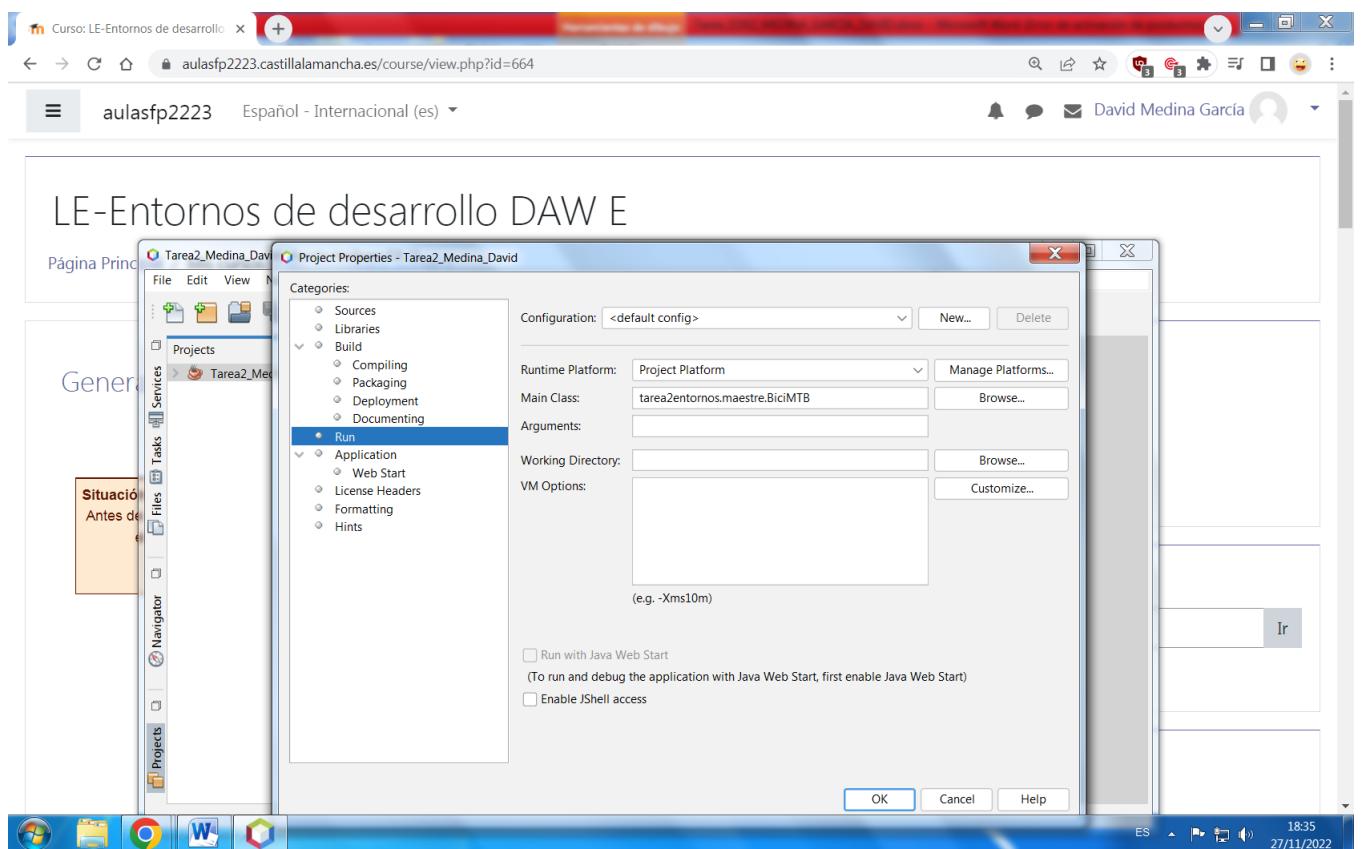


En este apartado podemos modificar algunas opciones relativas a los **paquetes** del proyecto, como el comprimir su archivo.

Los **paquetes** son el mecanismo que usa Java para facilitar la modularidad del código. Un **paquete** puede contener una o más definiciones de interfaces y clases, distribuyéndose habitualmente como un archivo. Para utilizar los elementos de un **paquete** es necesario importar este en el módulo de código en curso, usando para ello la sentencia `import`.

Los **paquetes** son una forma de organizar grupos de clases. Un **paquete** contiene un conjunto de clases relacionadas bien por finalidad, por ámbito o por herencia. Los **paquetes** resuelven el problema del conflicto entre los nombres de las clases. Al crecer el número de clases crece la probabilidad de designar con el mismo nombre a dos clases diferentes.

- **Run**

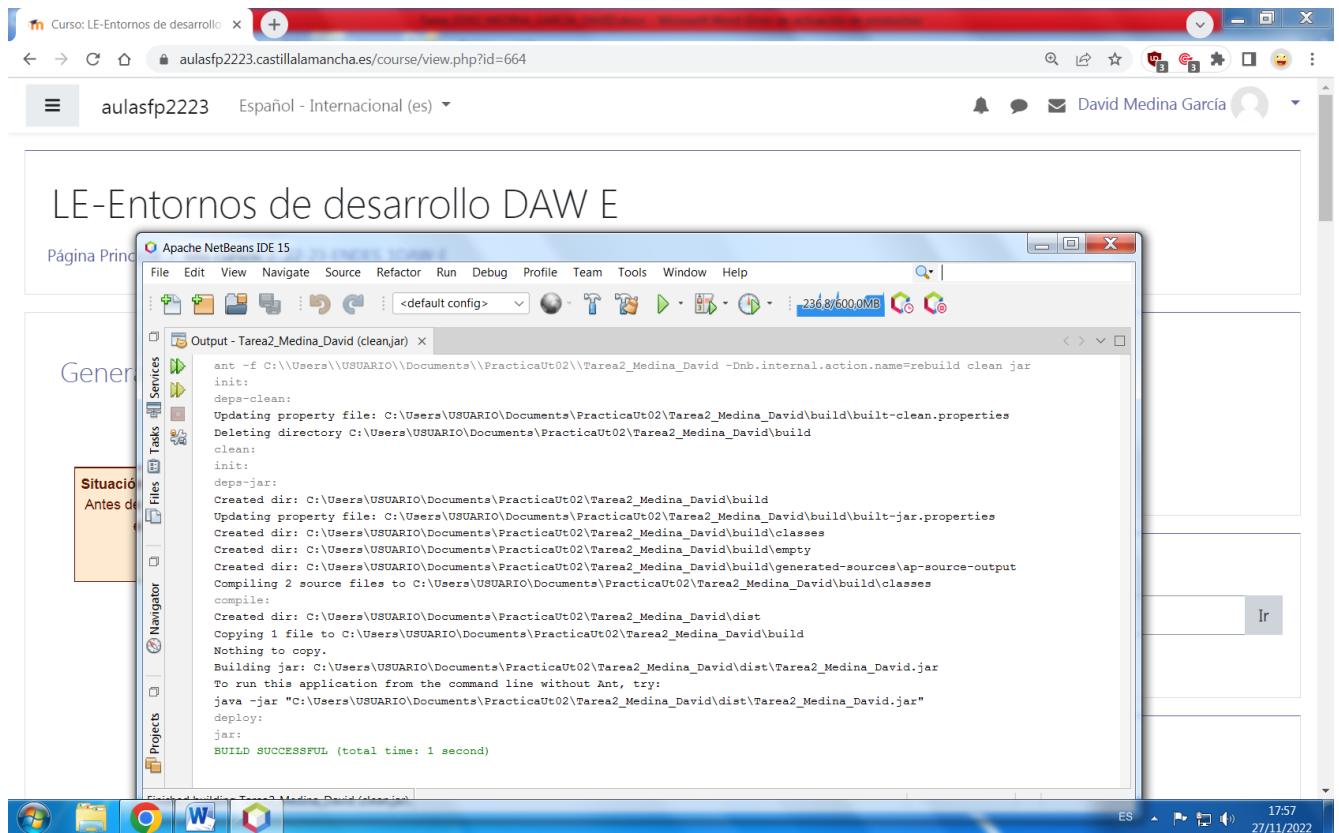


En este apartado podemos cambiar algunas condiciones de la ejecución del proyecto, como la plataforma donde se ejecuta o su main class.

Con run ejecutamos el proyecto. Se puede hacer de distintas maneras, pulsando **F6**, abriendo la pestaña de **Run** clicando en **Run Project** o con el botón que viene por defecto en el programa ➤.

## 9. Generar el .jar

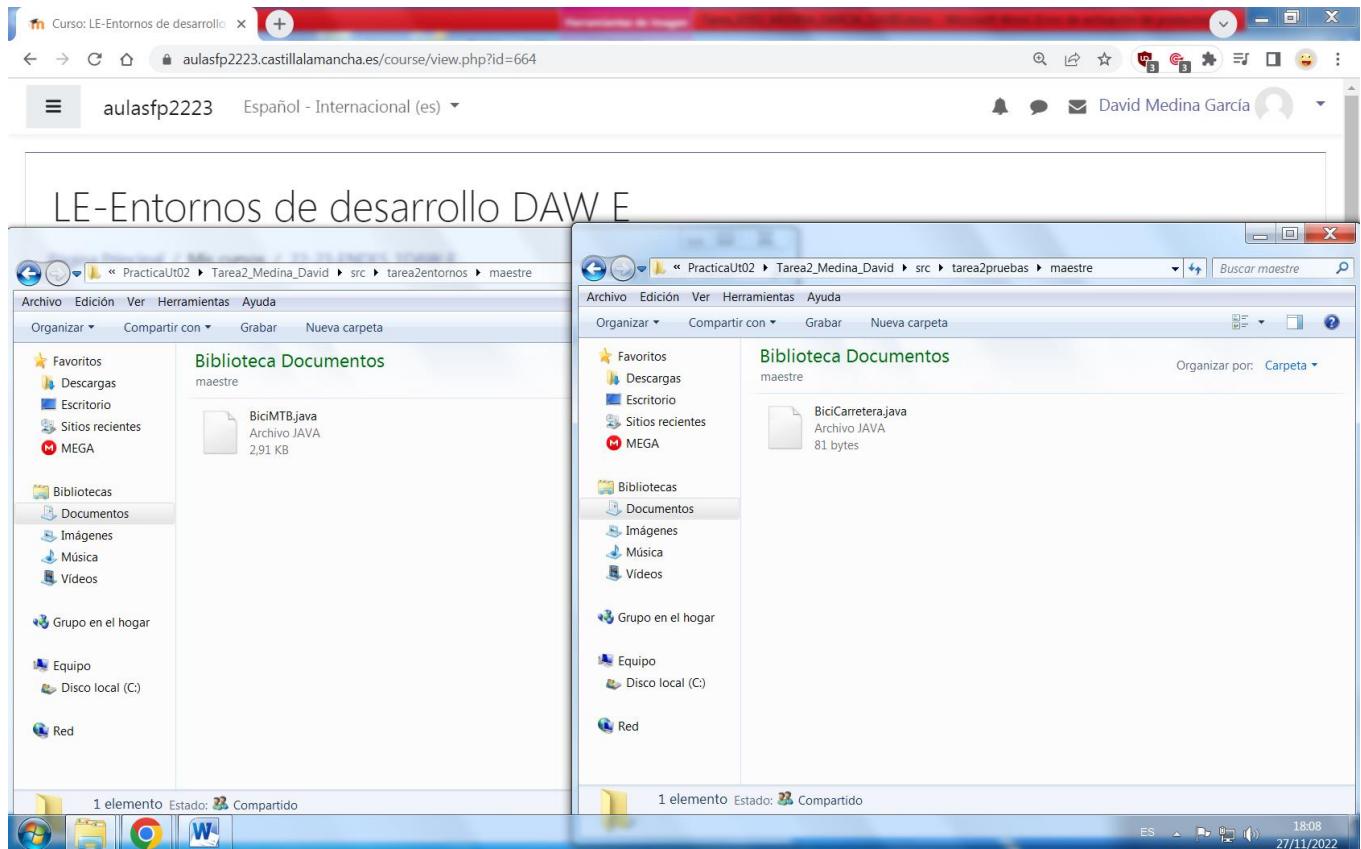
Para generar el .jar solamente clicamos en el símbolo del **martillo** y la **brocha**, al lado del símbolo **Run**, y ya se crearía, mostrando el resultado por el **Output**.



## 10. Subcarpetas .java / .class / .jar

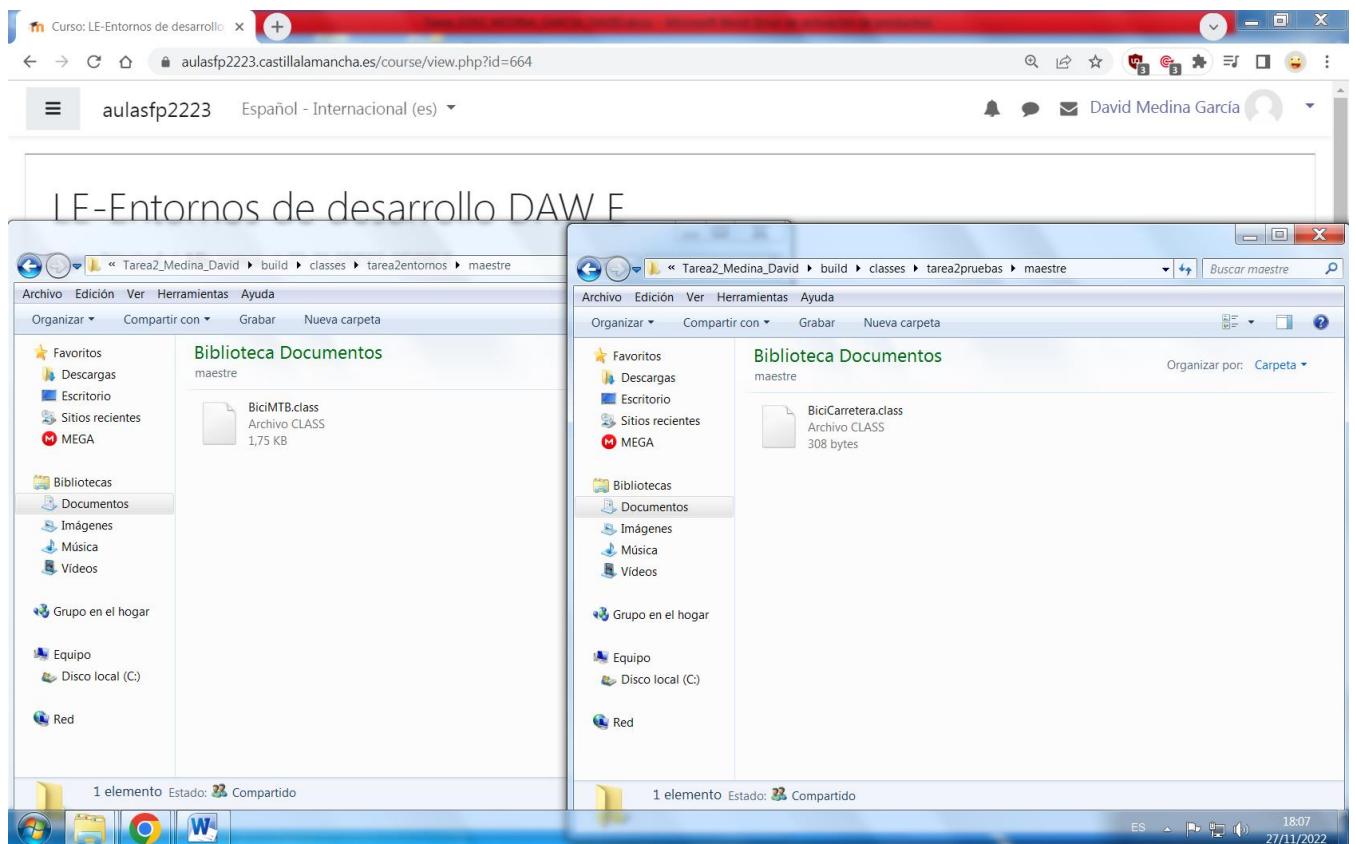
### JAVA

Buscamos el directorio de nuestro proyecto y allí nos vamos a la subcarpeta de los archivos **.java** y hacemos pantallazo.



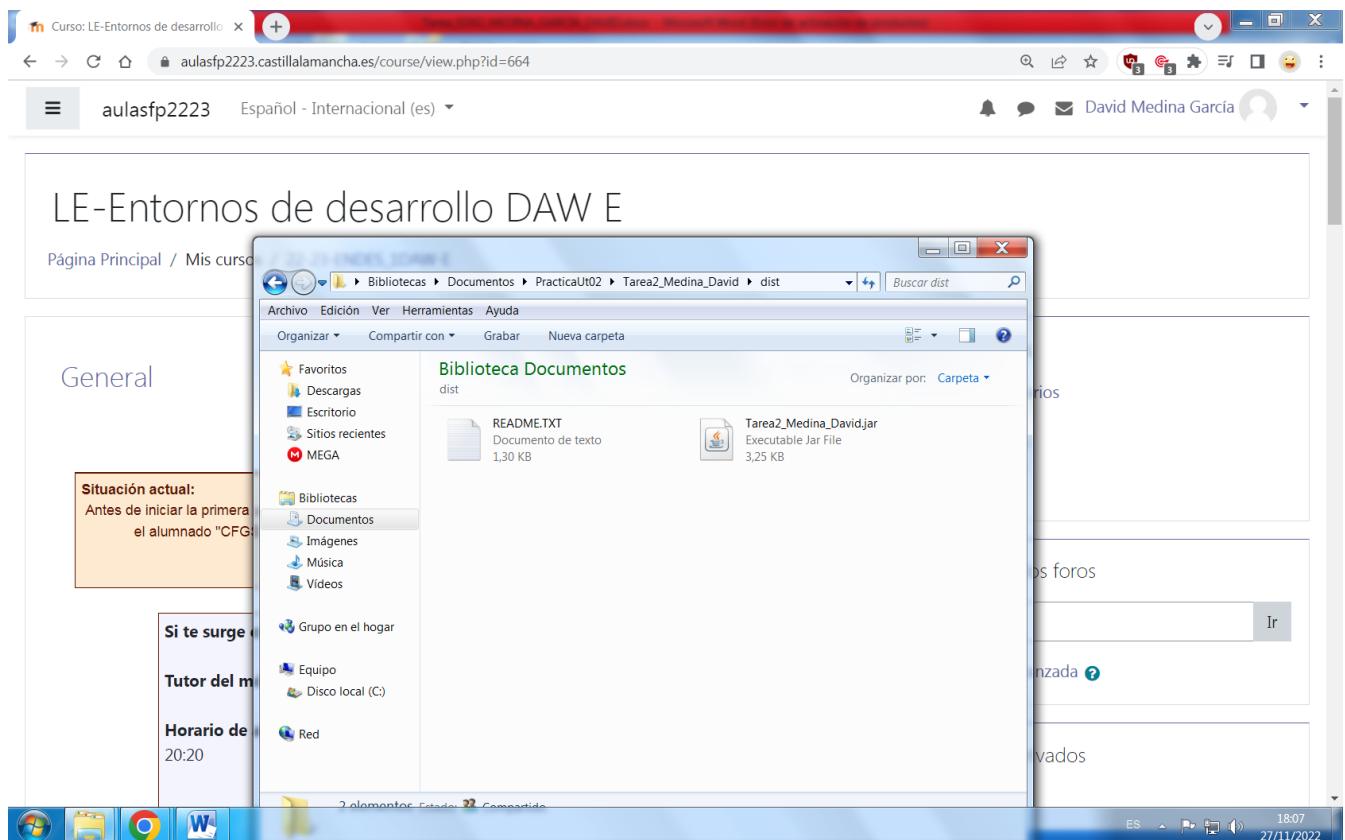
## .CLASS

Buscamos el directorio de nuestro proyecto y allí nos vamos a la subcarpeta de los archivos .class y hacemos pantallazo.



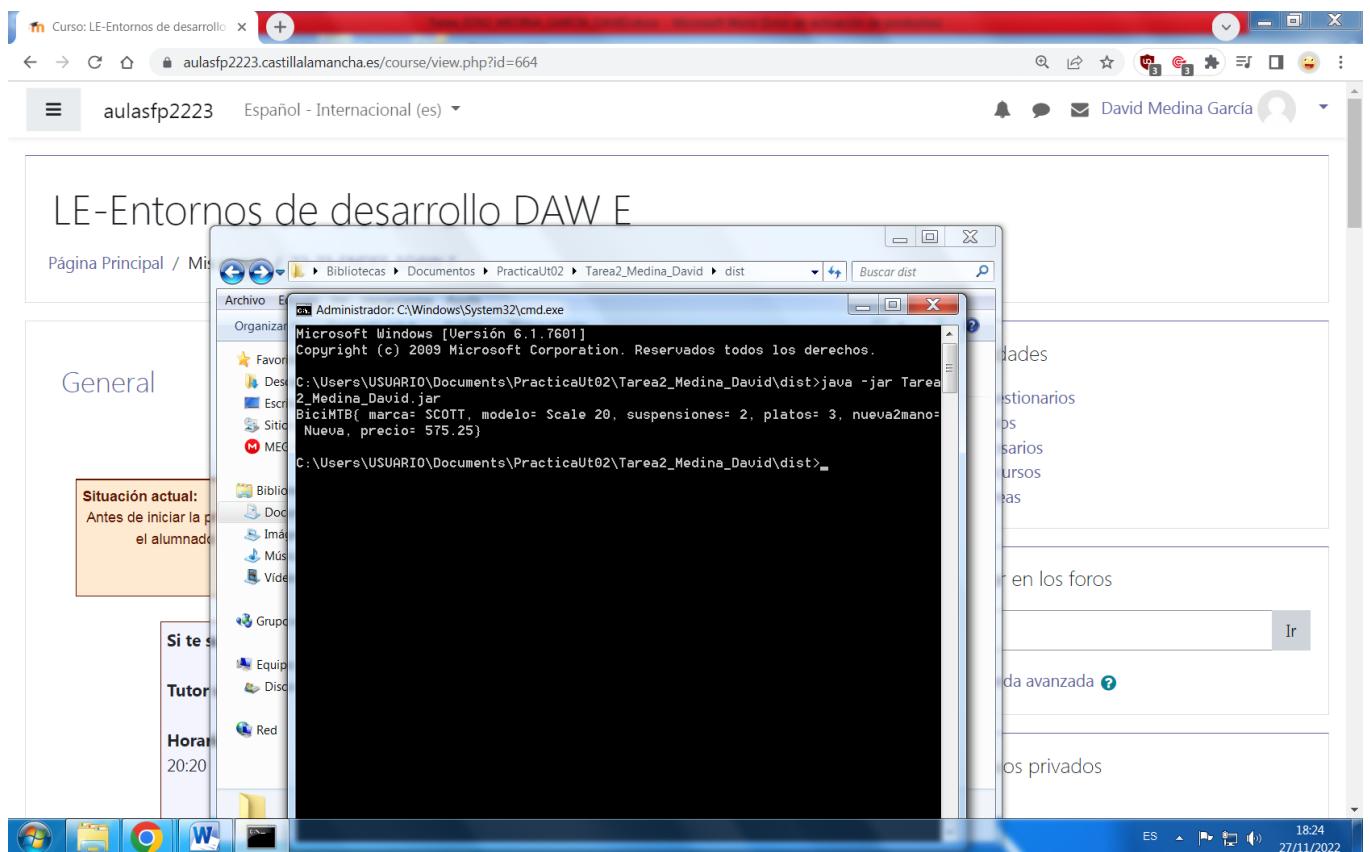
## JAR

Buscamos el directorio de nuestro proyecto y allí nos vamos a la subcarpeta de los archivos .jar y hacemos pantallazo.



## **11. Ejecutar archivo .jar desde consola**

Ejecutamos la consola desde la subcarpeta del archivo **.jar** introduciendo su directorio (C:\Windows\System32\cmd.exe). Una vez allí escribimos “**java -jar**” seguido del nombre del archivo con su extensión (**.jar**), en este caso: **Tarea2\_Medina\_David.jar** y pulsamos **enter**.

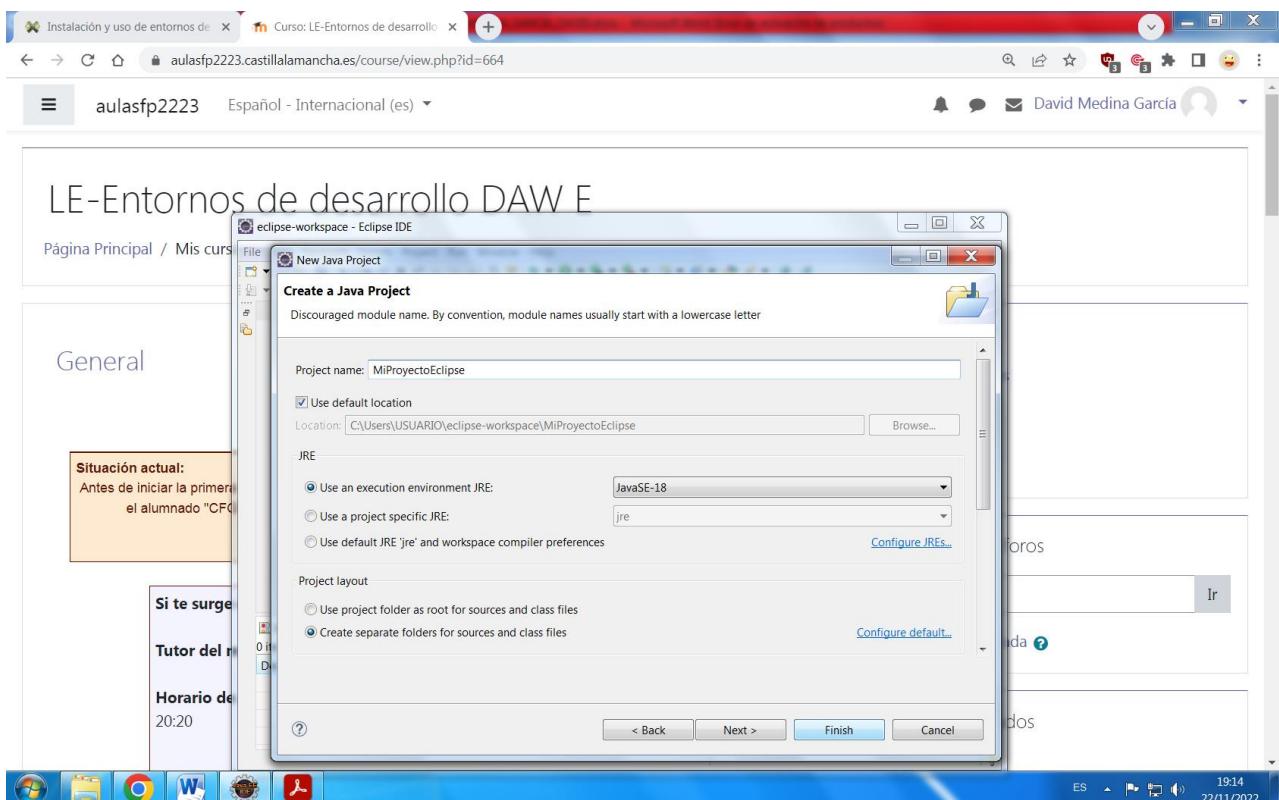


## 12. Eclipse

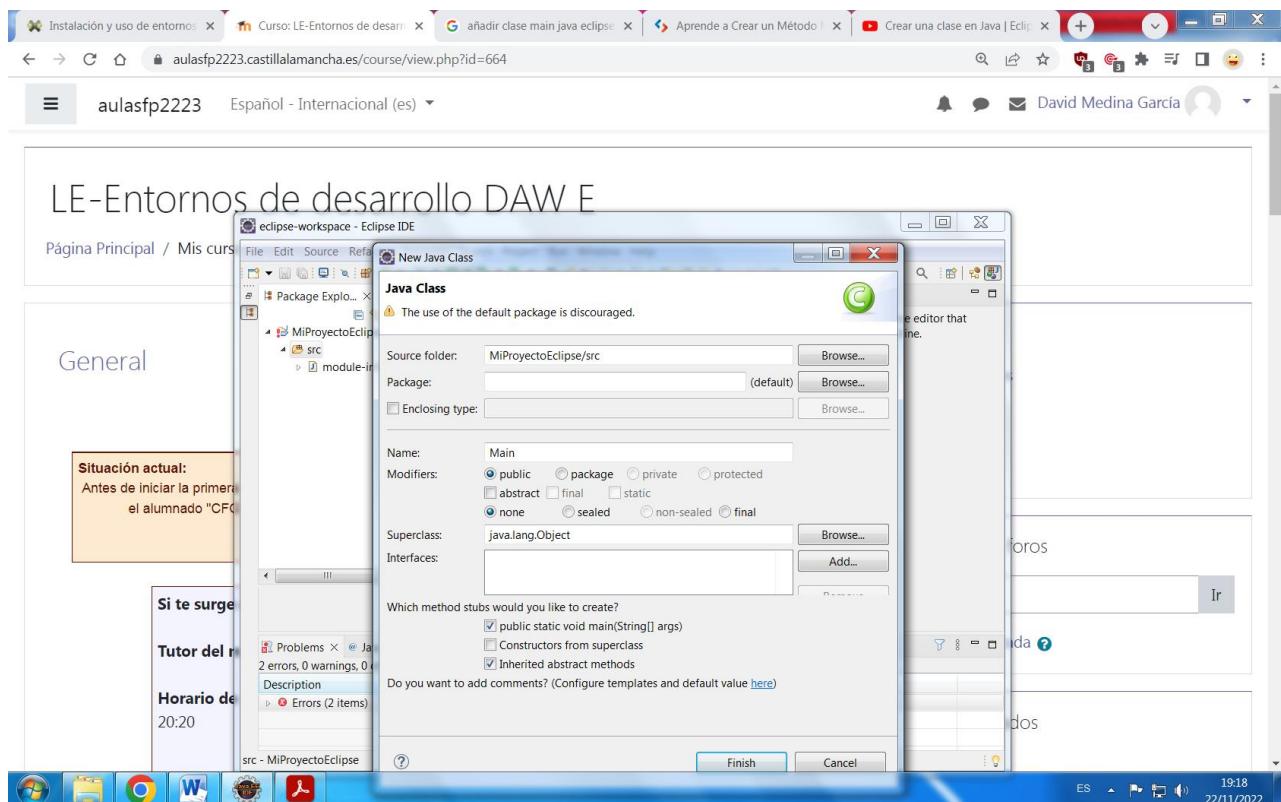
Primero instalamos **Eclipse** en nuestro equipo tras descargarlo de su página web. Lo abrimos y nos pedirá que seleccionemos un **workspace** (lo podremos cambiar luego).

Nos vamos a la pestaña **File > new** y seleccionamos **new java Project**. Le damos nombre y ubicación y finalizamos. Desplegamos la pestaña **package explorer** de la izquierda y hacemos click derecho en nuestro proyecto. Seleccionamos **New > Java class**, le damos nombre y ubicación y finalizamos. Finalmente escribimos el sencillo código del ejemplo y ejecutamos para demostrar su funcionamiento.

### Creación proyecto



## Creación clase main



## Ejecución código

