

ciclo: DAW

MÓDULO DE lenguajes de marcas y sistemas de gestion

Tarea Nº 2

Alumno:

Daniel del Valle González

53665340S

Contenido

[1. Documentos que se adjuntan a este informe. 2](#_Toc84303070)

[2. Respuestas de la tarea: 2](#_Toc84303071)

[RA1\_a) 2](#_Toc84303072)

[https://www.pccomponentes.com/msi-x470-gaming-plus-max 2](#_Toc84303073)

[RA1\_b) 3](#_Toc84303074)

(Una vez realizado el informe, no olvidar actualizar esta tabla del índice **(F9 + Actualizar toda la tabla)**, con el fin de que se actualicen todos los epígrafes y números de página)

1. Documentos que se adjuntan a este informe.

A continuación se detallan los documentos que componen la presente entrega de la tarea:

1. Informe de elaboración de la tarea.
2. Archivos HTML: Inicio, Contacto y Artículos.
3. Archivos CSS.
4. Respuestas de la tarea:

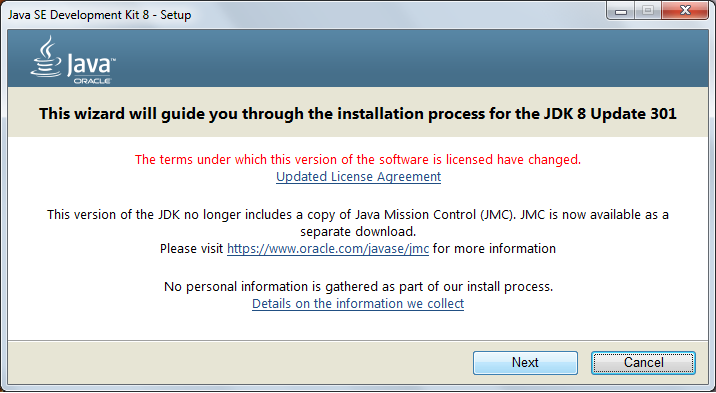
****RA1\_a)**** La tarea pide que se realice usando HTML5.¿cómo indicamos esto en el fichero HTML?

* Con la DTD (Document Type Declaration) : <!DOCTYPE HTML>

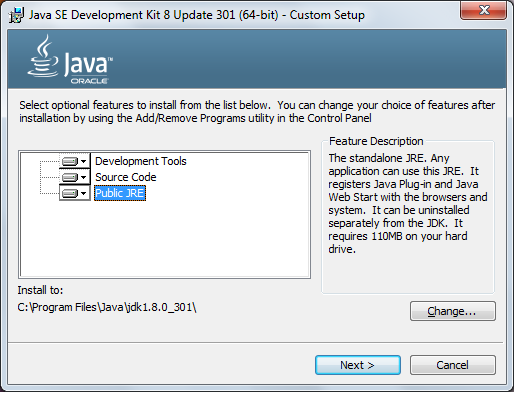
****RA1\_b)****

**Escojo el instalador para Windows x64**

**Pulso next para continuar con el instalador.**

****

**Elijo la instalación estándar, con todos los paquetes.**

****

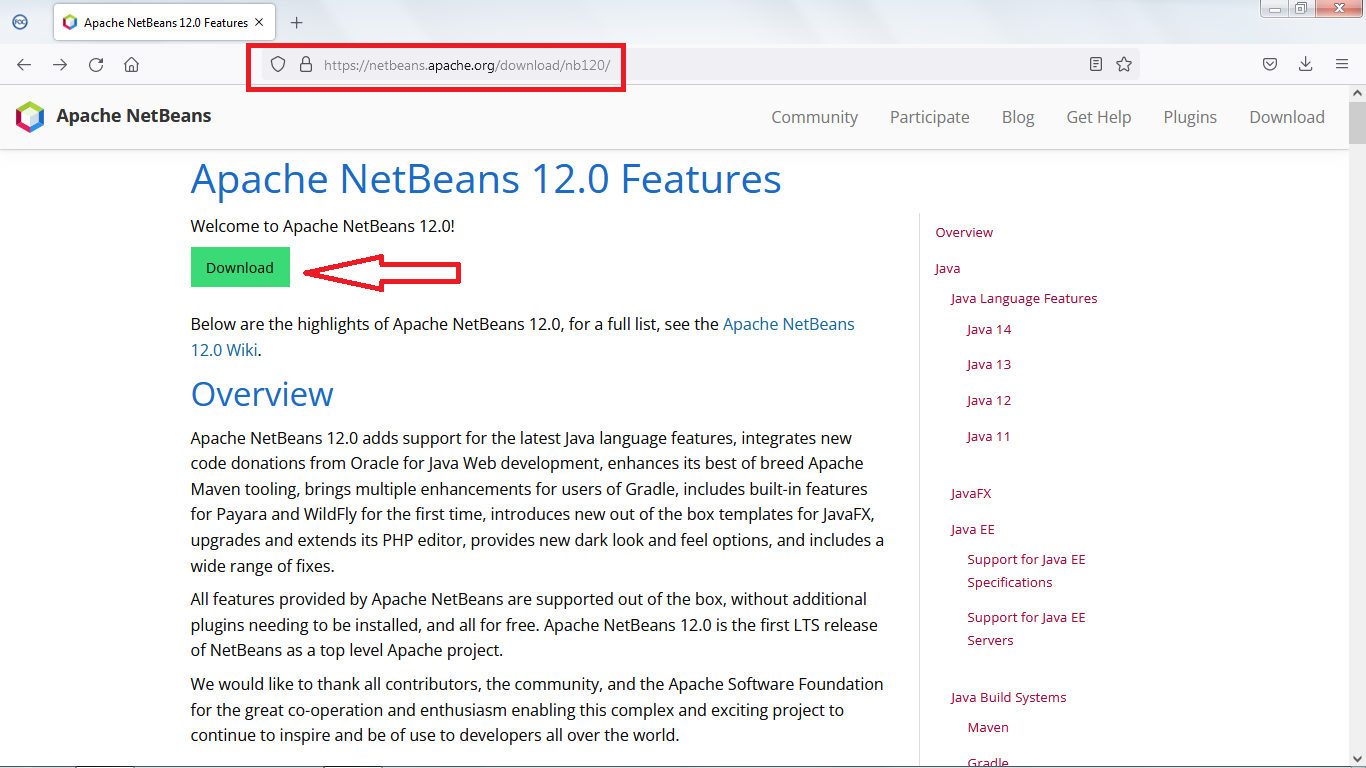
**Tras esto, el JDK se instalará el JDK 8\_301.**

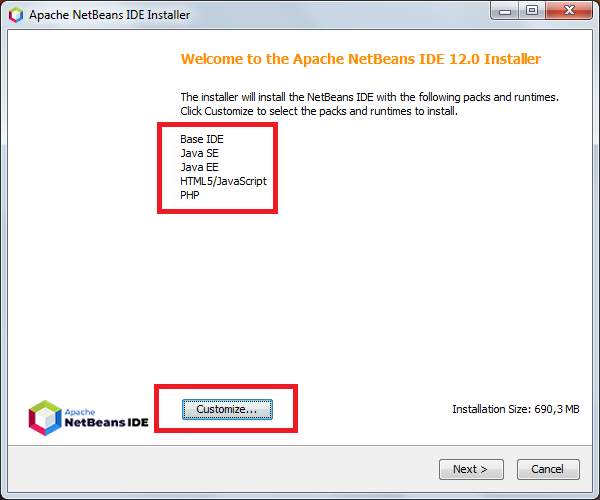
**A continuación, repetiré los mismos pasos para instalar el JDK 11.0.**

**(Sólo sería necesario uno, pero quería probar con ambos para practicar la selección de JDK según probables necesidades futuras).**

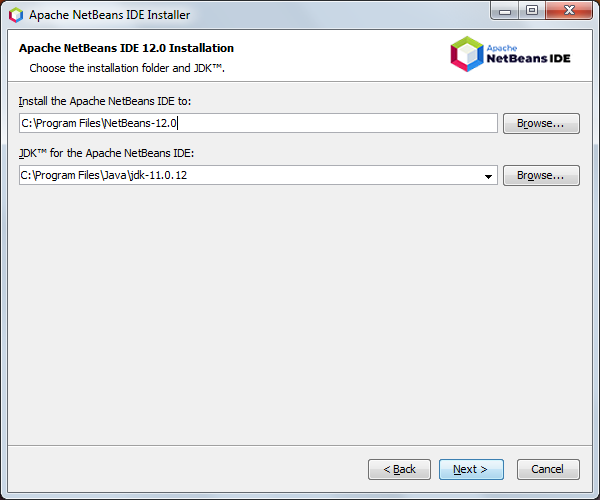
**Instalo Netbeans 12.0 LTS tras haber instalado los JDK**

**Descargo de la web oficial de Apache Netbeans.**

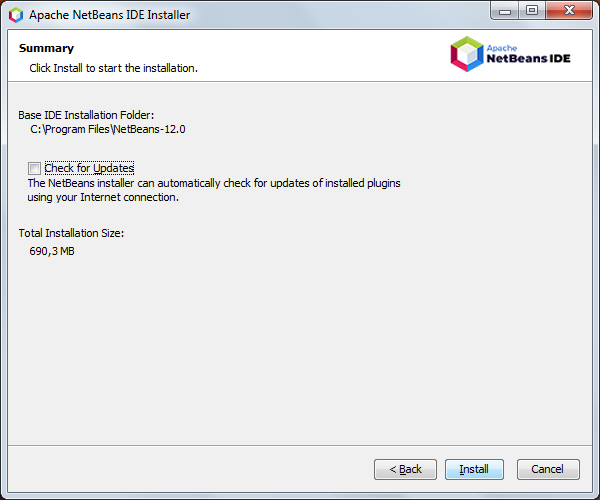
****

**Me da la opción de elegir qué componentes instalar, si bien algunos en el fondo son necesarios. En cualquier caso, instalo todos para necesidades futuras.**

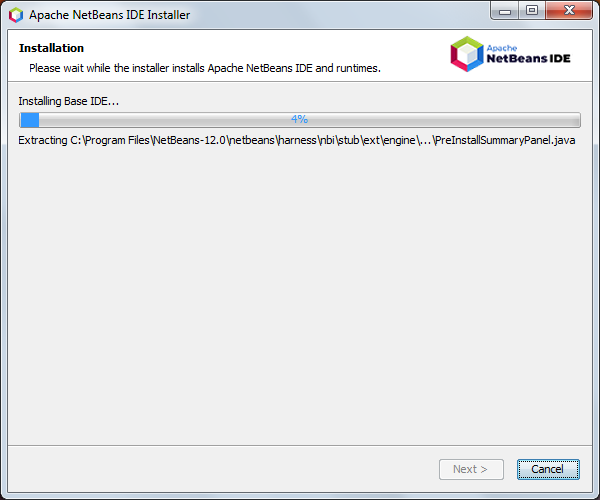
**Escojo la ruta de instalación, dejando la que se propone por defecto**

****

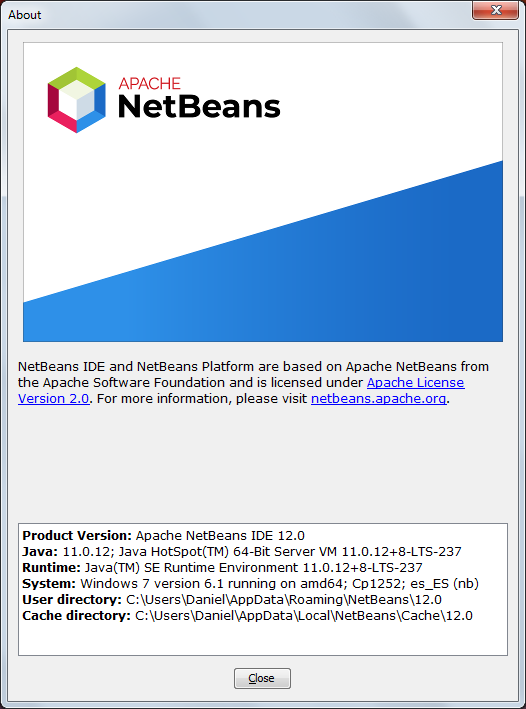
**Desmarco la casilla de “buscar actualizaciones” para poder hacerlo después, en un proceso separado y controlado, y documentarlo.**

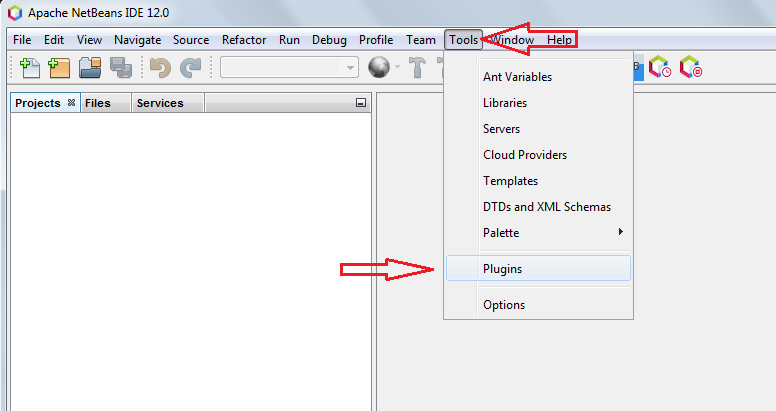
****

**El programa comienza a instalarse tras pulsar “Install”**

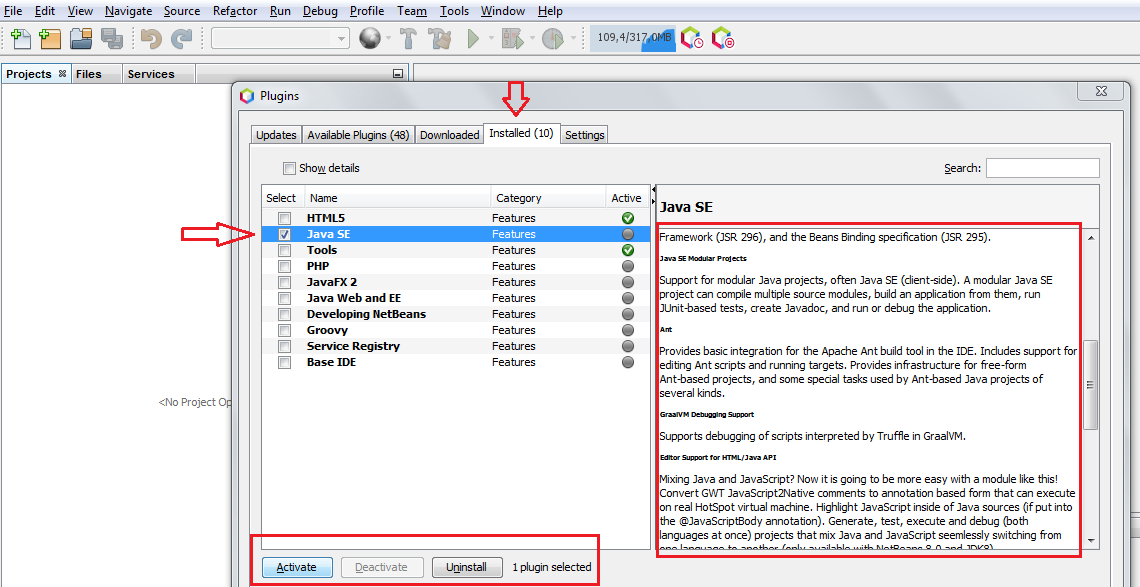
****

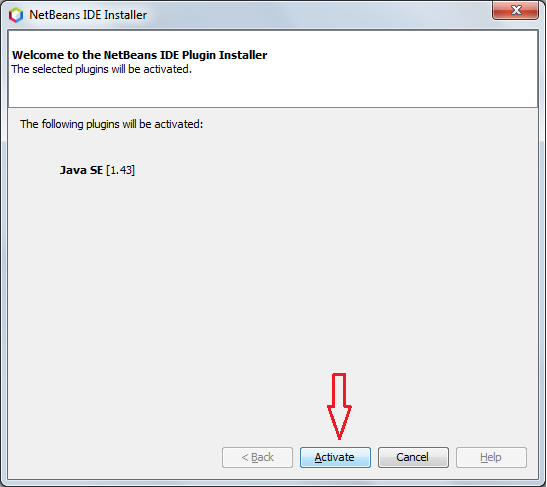
**En la pestaña “about”, se muestran la info básica del programa, ya instalado.**

****

**En la pestaña tolos, en “plugins”, habilitaré el plugin ya instalado de “Java SE”, para poder crear proyectos en Java. **

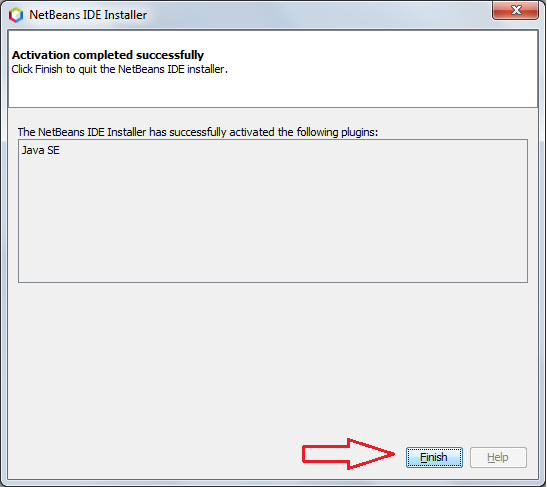
Pulsando “Activate”,lo habilito.

****

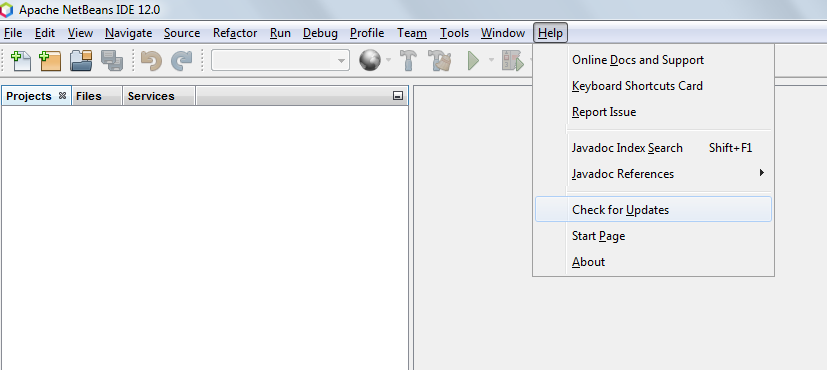
****

Pulso activate de nuevo.

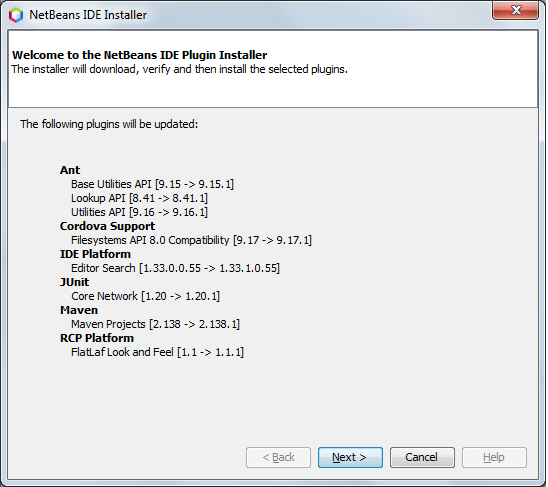
Se muestra el mensaje de consola indicando la instalación satisfactoria.

****

Busco actualizaciones en la pestaña “help”, para poner el IDE al día.

****

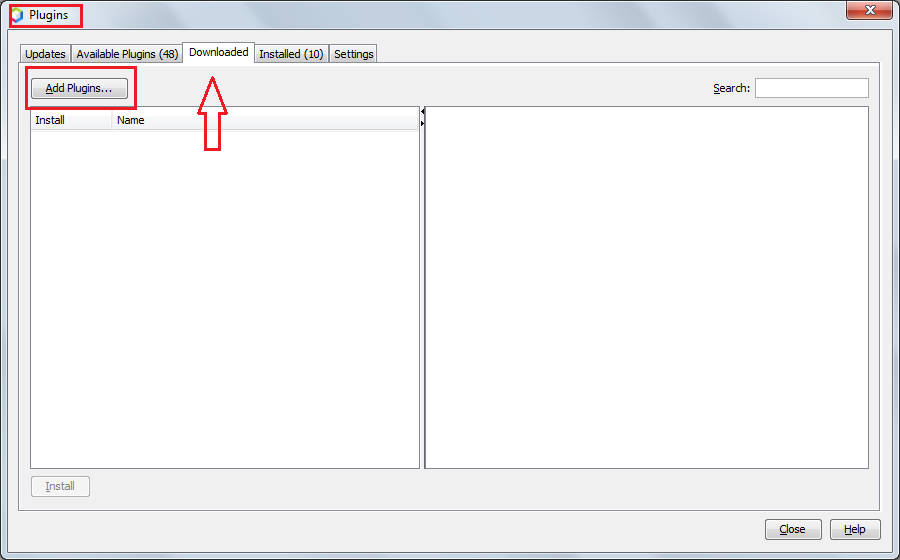
**Se me informa de que los plugins de la lista serán actualizados; pulso “Next”**

****

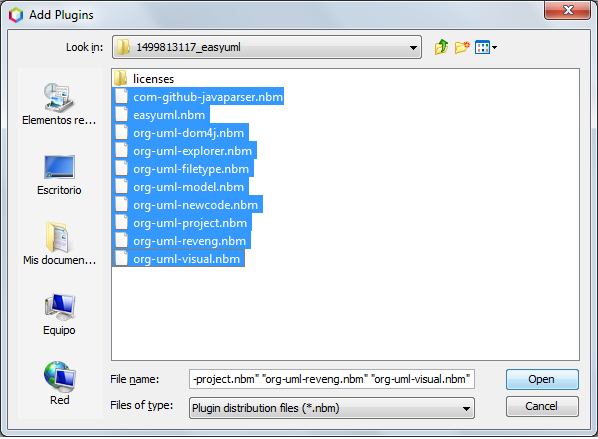
**A continuación, instalaré una herramienta CASE que el tutor nos ha facilitado.**

**Para ello, pulso “Add Plugins” en la pestaña “descargados” de Plugins.**

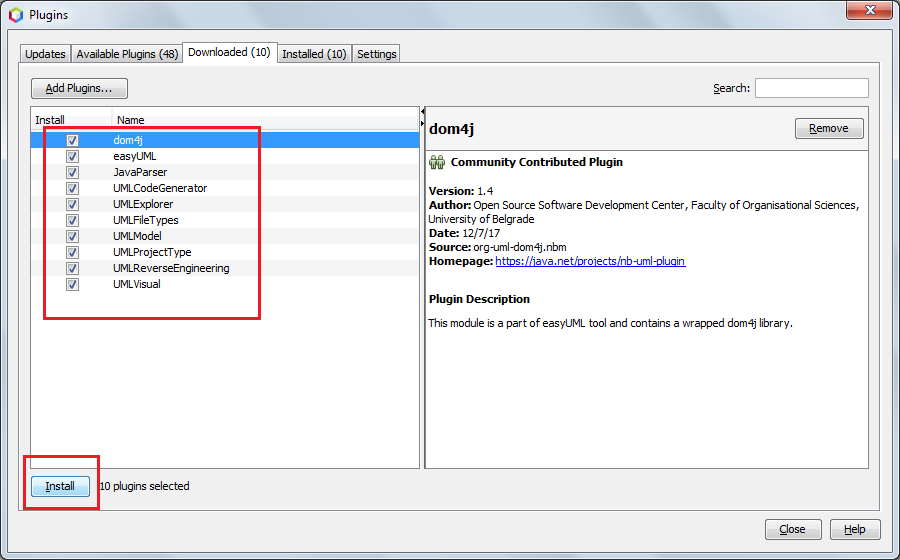
**Esto me permitirá escoger un archivo o carpeta ubicado en mi PC. Es una de las ventajas de ser un IDE de código abierto.**

****

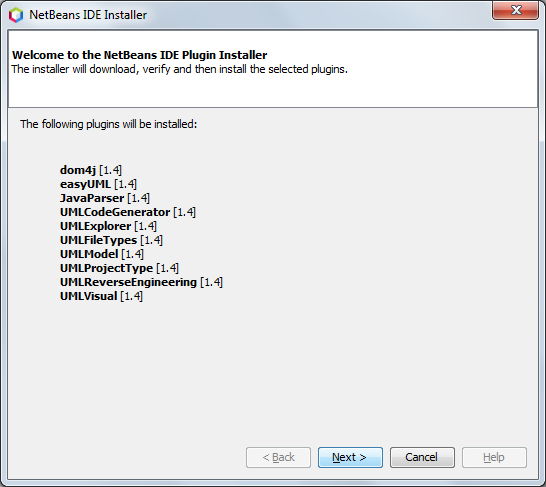
**Selecciono todos los archivos .nbm, cuya extensión significa “NetBeansModule”**

****

**Marco todos y pulso “Install”**

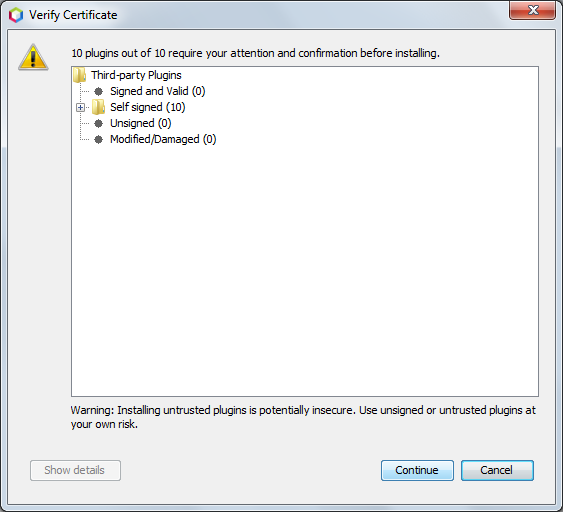
****

**El sistema me confirma los módulos que se instalarán.**

****

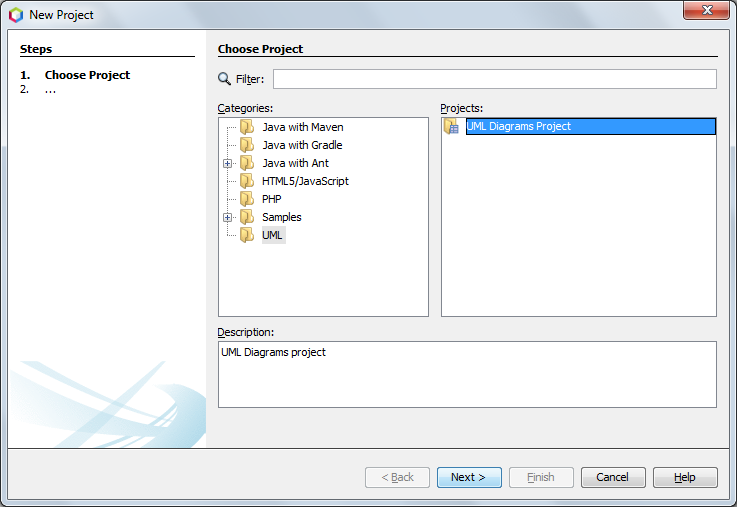
**Salta un aviso de seguridad al tratarse de módulos creados por colaboradores “no oficiales” o confiados de Netbeans (código abierto).**

**Continúo ya que tengo la certeza de que no perjudicarán mi sistema.**

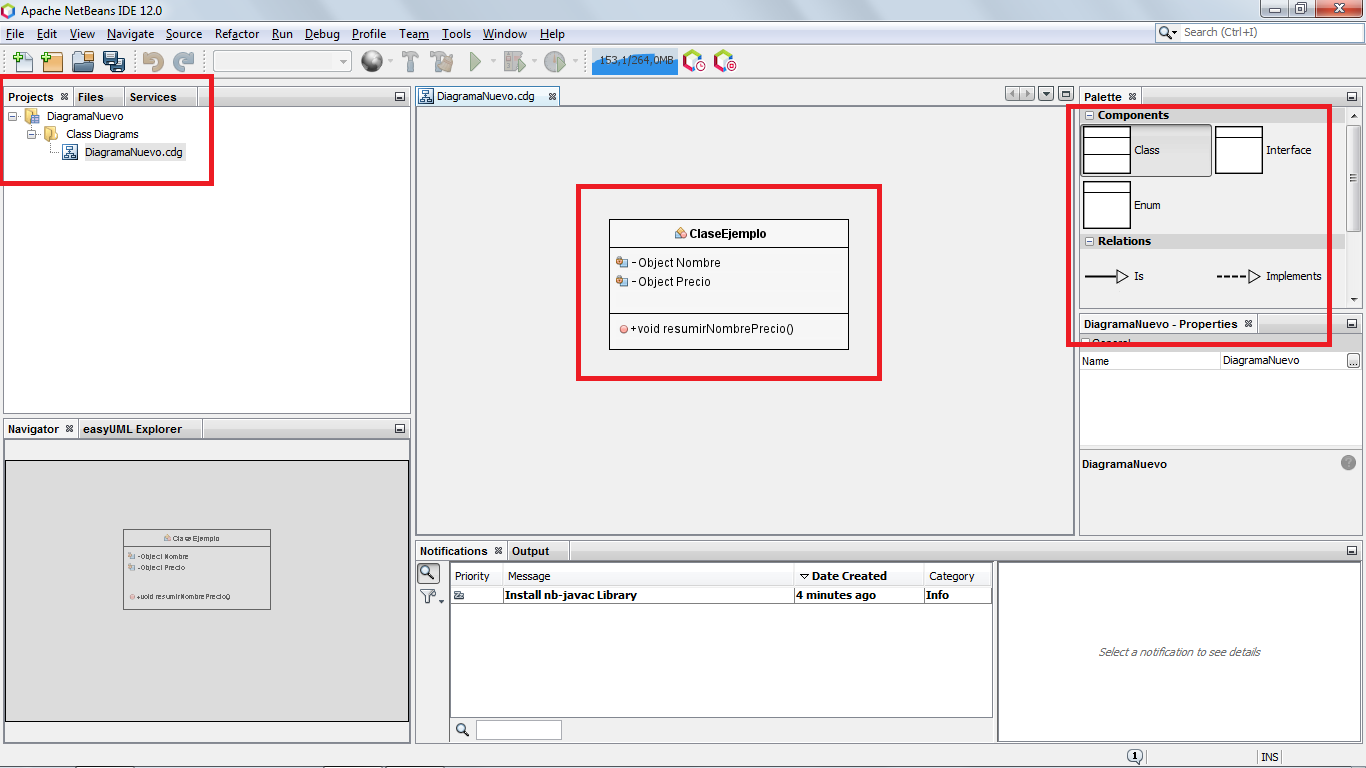
****

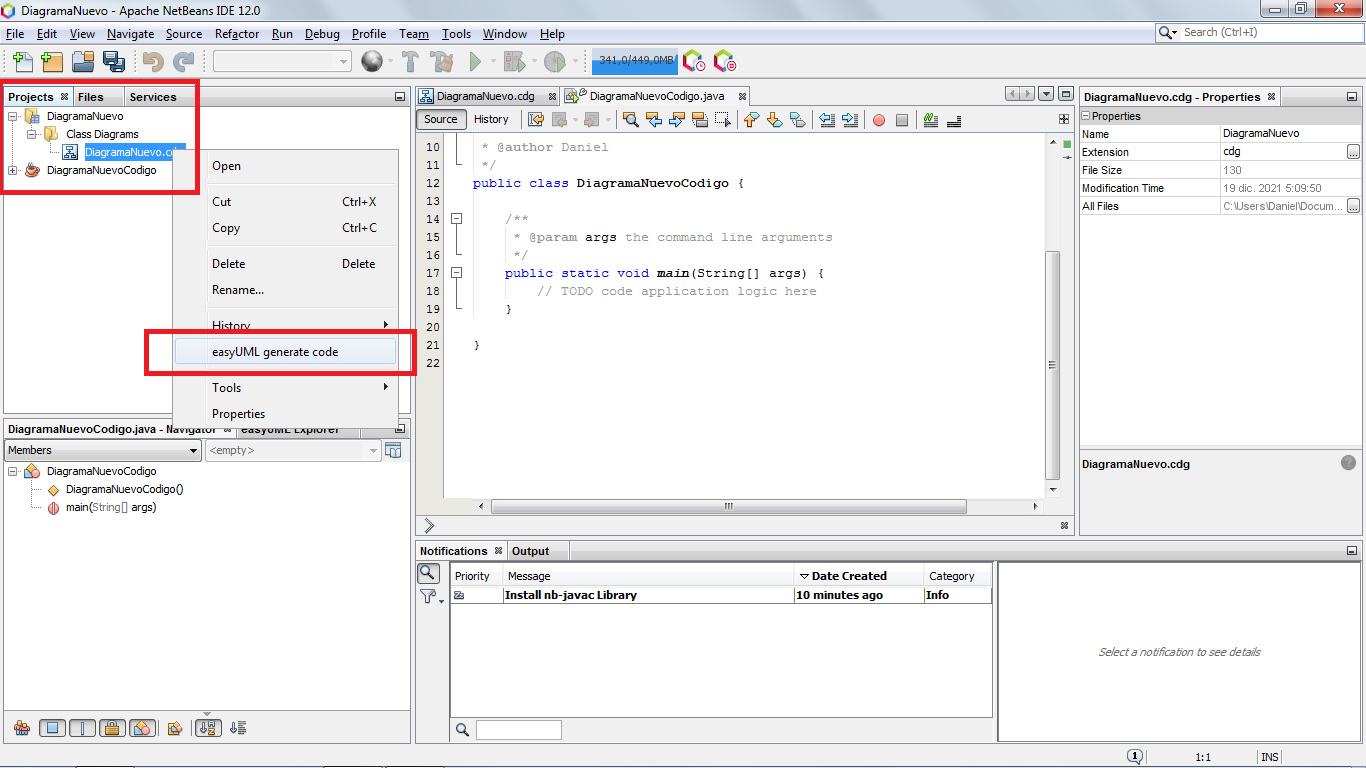
**Ahora, ya puedo usar esta herramienta CASE, que me permite crear diagramas parecidos a los de Entidad – Relación, donde podré transformar cada tabla que yo cree, en código del programa. Una utilidad muy buena.**

**Se trata de una herramienta CASE de tipo DORSAL. Otra herramienta CASE Dorsal que incluye el IDE NetBeans es el Javadoc, que permite crear documentación a partir de los comentarios incluidos en el código.**

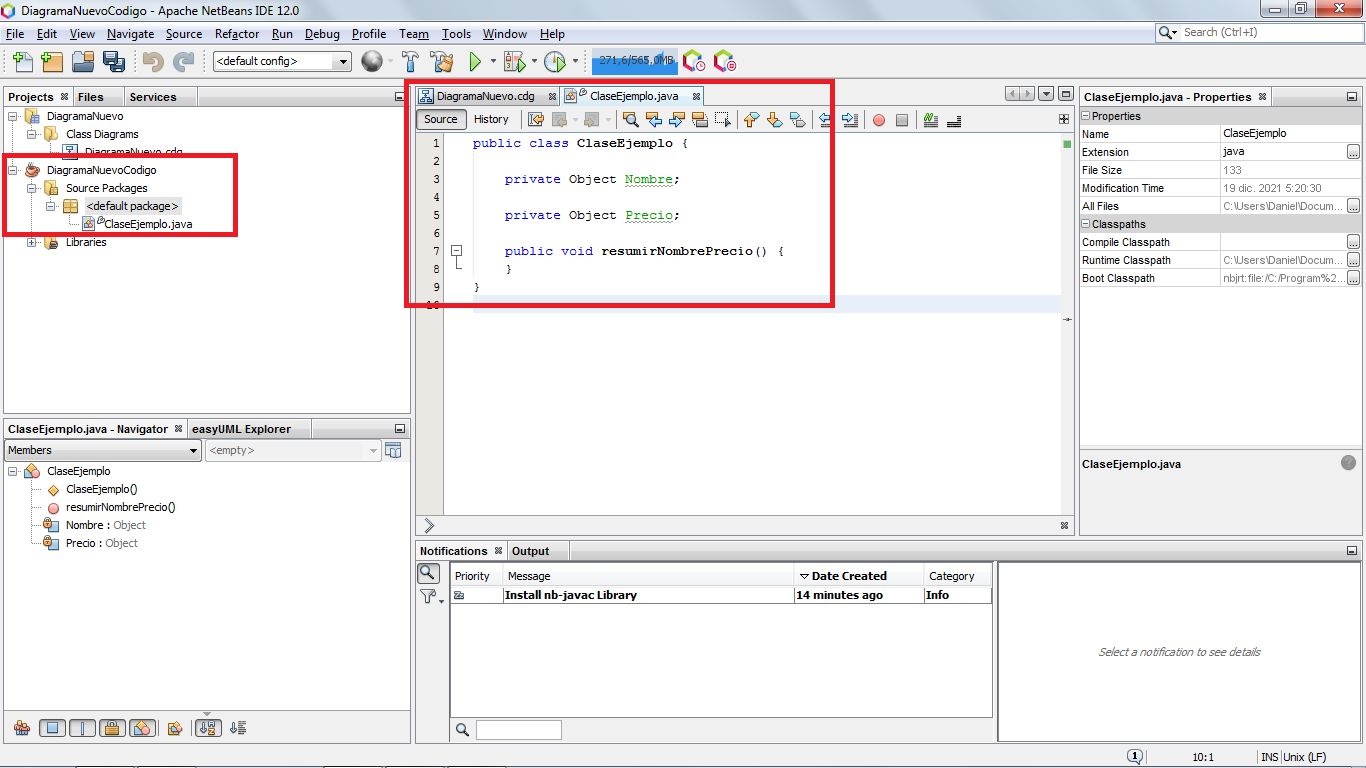
****

**Ejemplo de uso de herramienta UML**

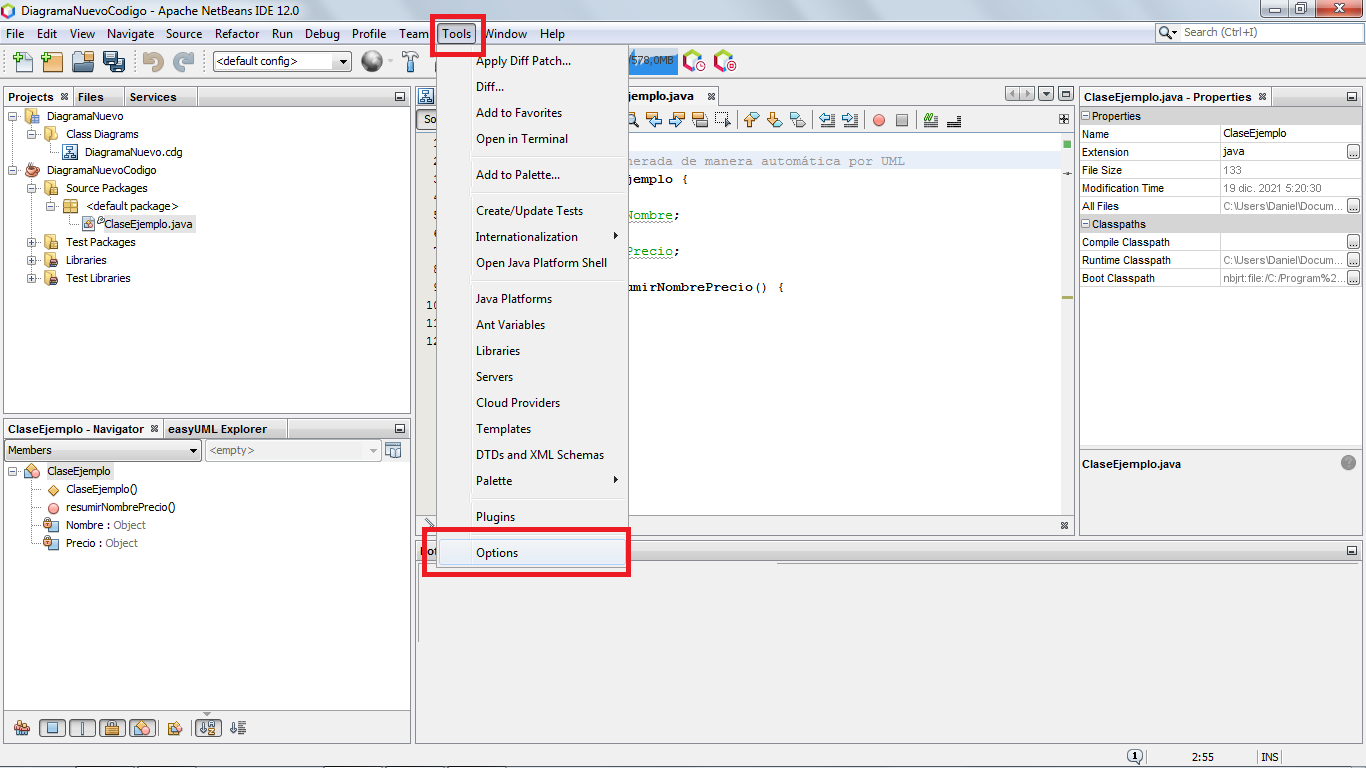


**Hay que crear un proyecto java para poder transformar la tabla en código.** 

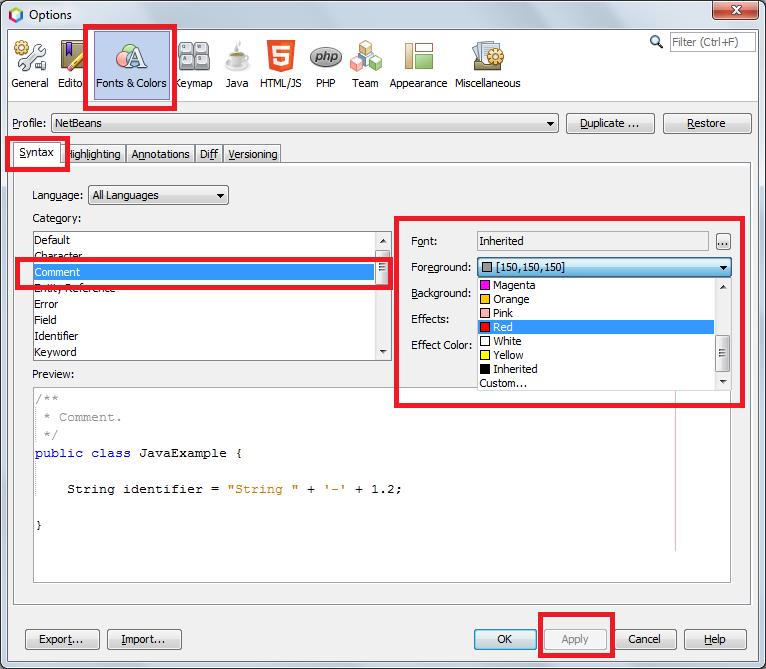
**Efectivamente, la herramienta ha generado código a partir de la tabla. Muy útil.**

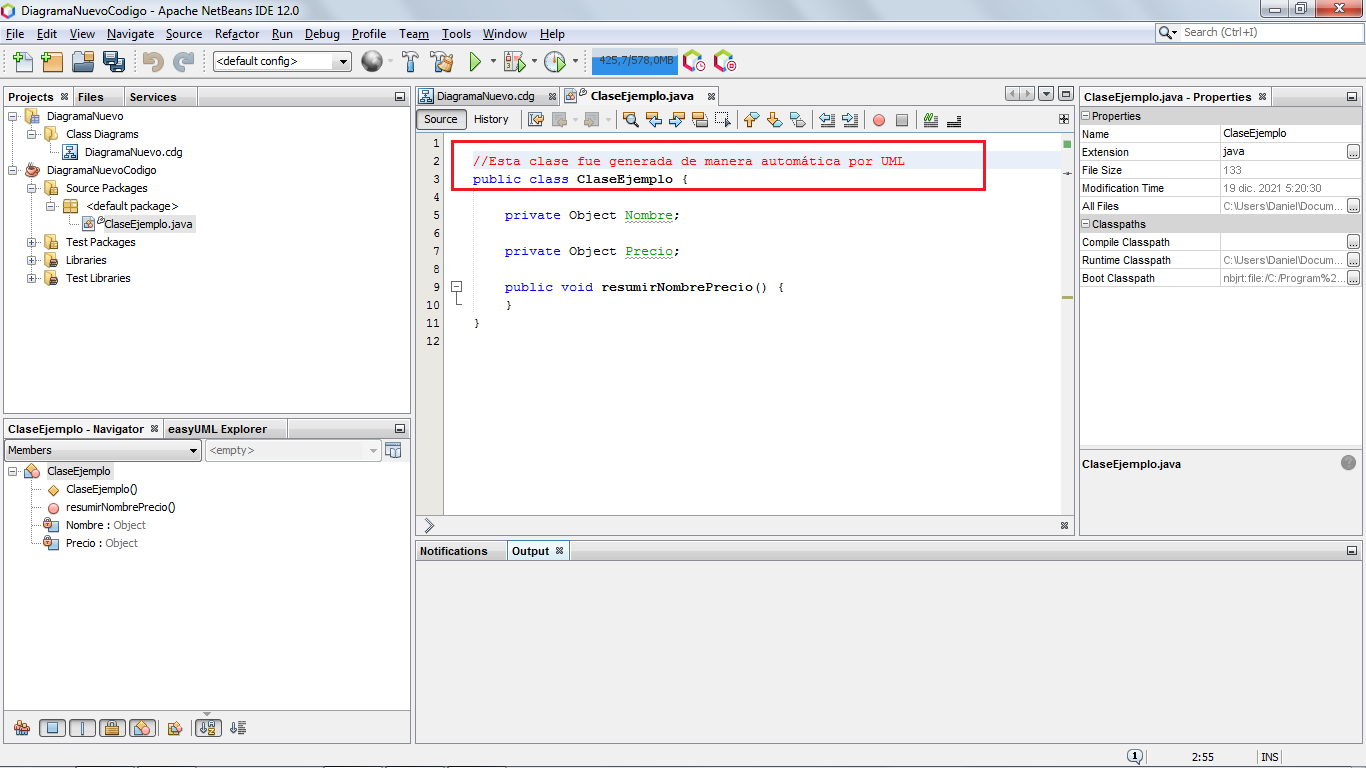


**Me dispongo a cambiar el color de los comentarios de gris a rojo, para mejor visibilidad**

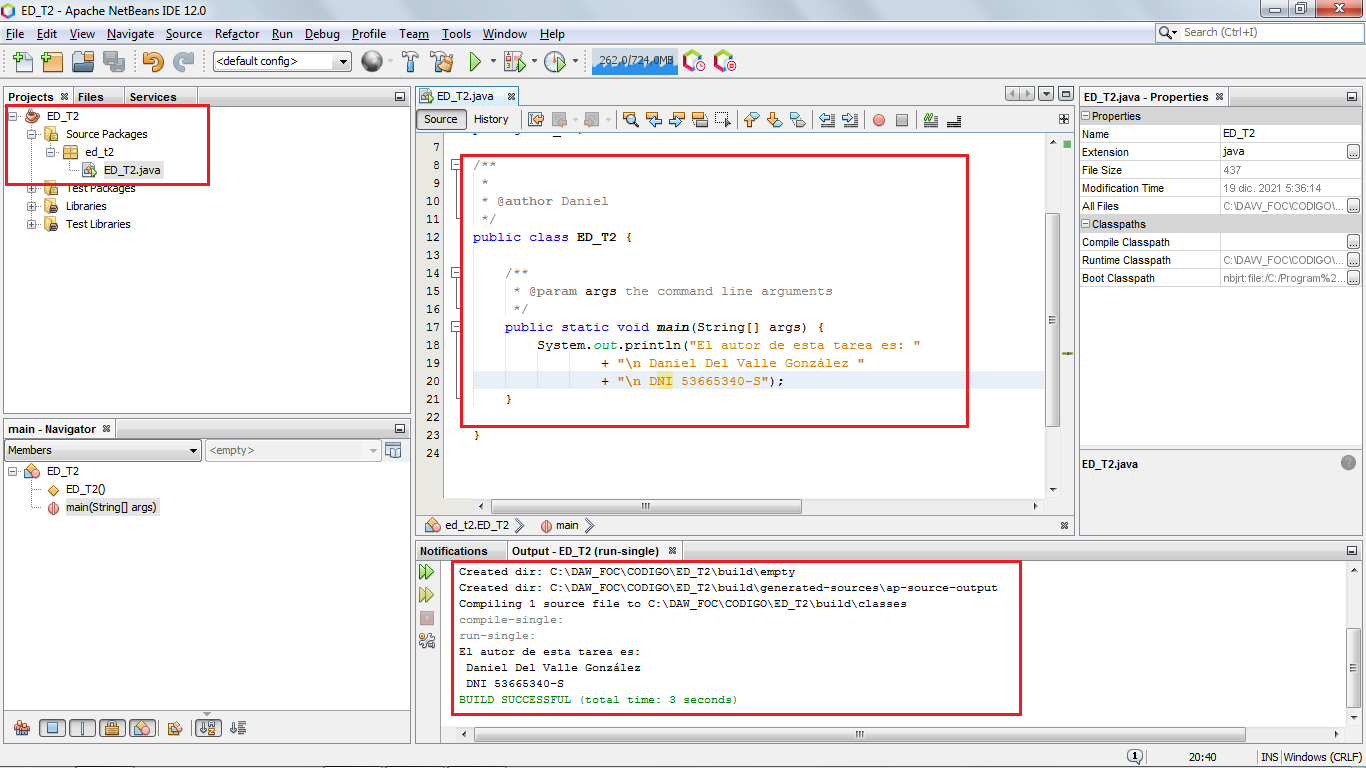


**Pulsando “Apply” cuando se habilite tras la selección, se aplicarán los cambios**

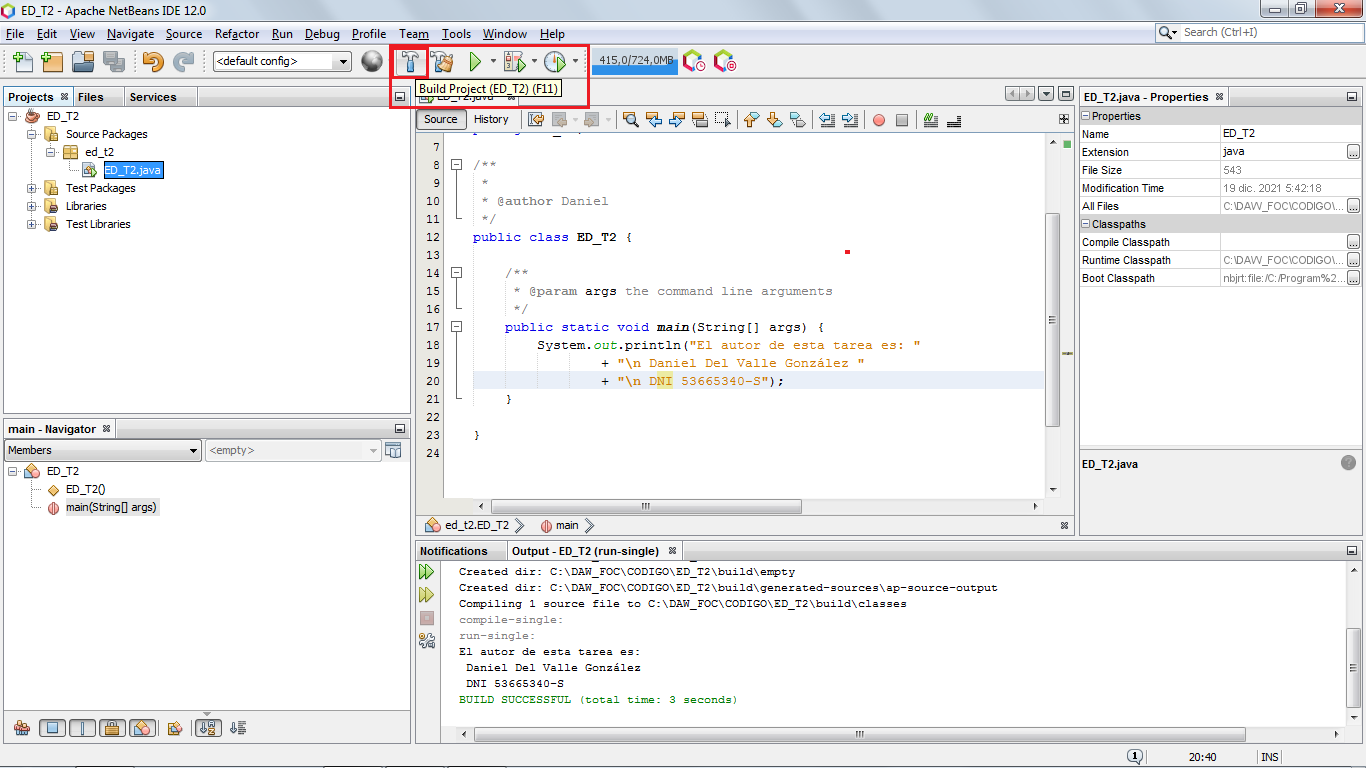




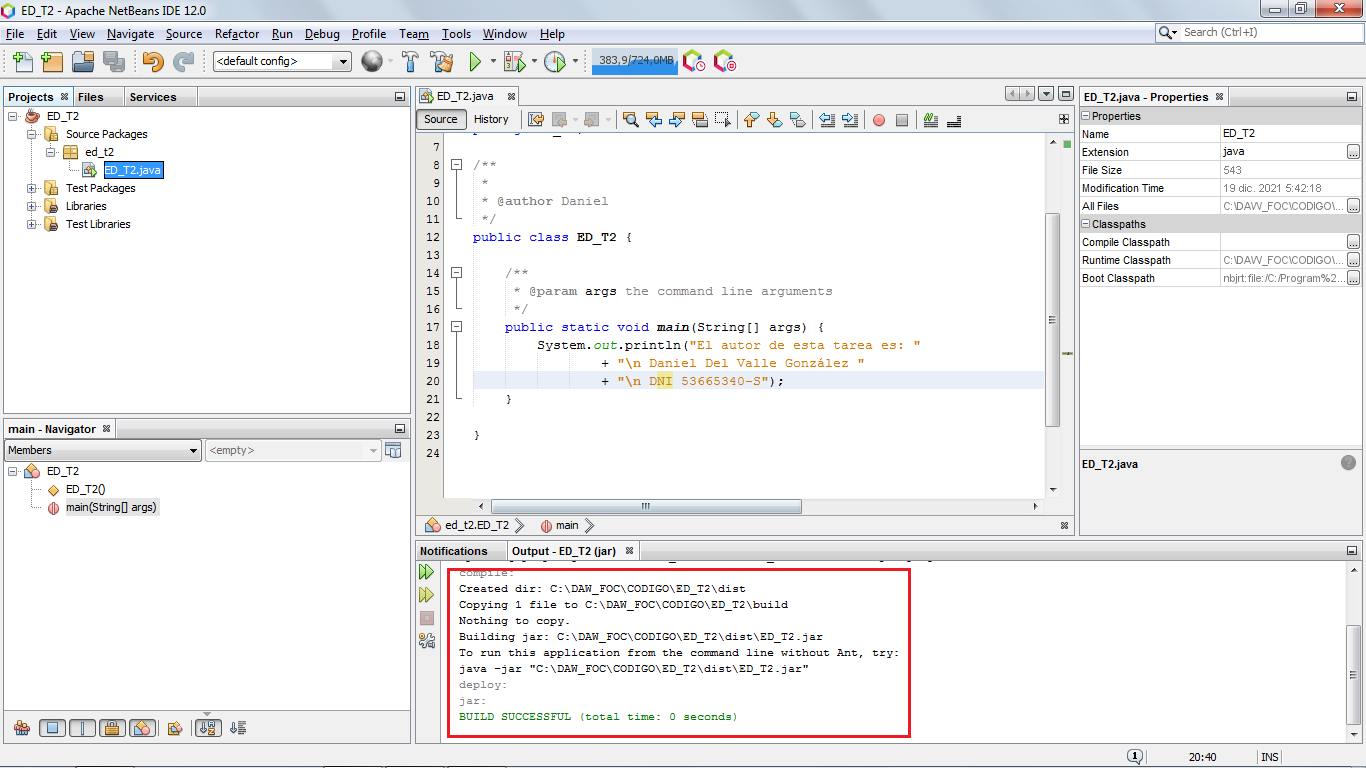
**Para generar un ejecutable a partir del IDE, primero hay que crear una clase, pulsando “File” + “New Project”. Tras elegir la ruta de almacenamiento y ponerle un título, en este caso “ED\_T2”, se crea un proyecto nuevo con una clase por defecto llamada como el proyecto. La clase, con el código requerido por la tarea, queda así:**



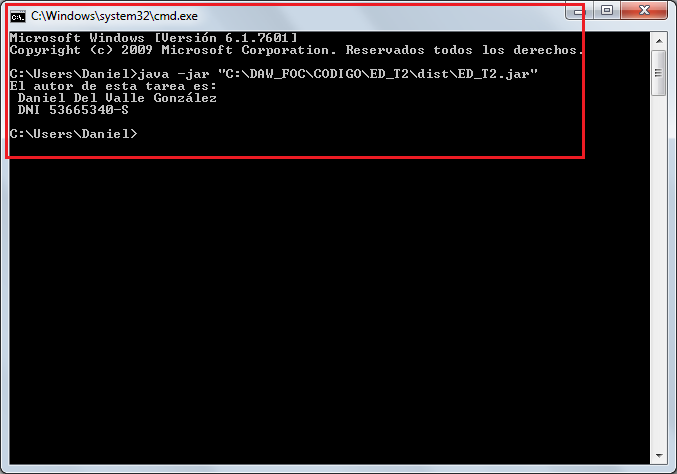
**Para generar el ejecutable, pulsamos “Build Proyect”**



**Tras esto, se notifica la creación de la carpeta “dist” dentro del directorio del proyecto, y el IDE nos muestra las instrucciones para poder ejecutar el archivo ejecutable .jar, ubicado en la carpeta dist, desde consola.**

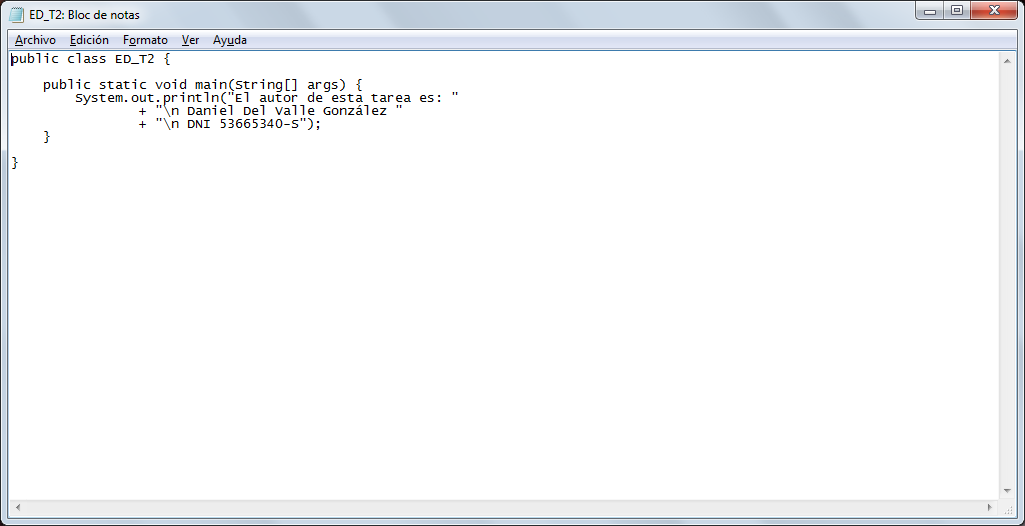


**Ahora, siguiendo las instrucciones, ejecuto el .jar Previamente, habré añadido la ruta C:\Program Files\Java\jdk-11.0.12\bin al path de las variables de entorno del sistema.**



**A continuación realizaré la misma tarea pero sin IDE.**

**Lo primero, copio el código (eliminando comentarios) en un editor de texto plano, y lo guardo en el escritorio con el nombre ED\_T2.java. Ha de llamarse tal y como se indica en el código de la clase, para que haya coherencia. Este será el archivo de código fuente que convertiré en código objeto (.class) y más tarde en código ejecutable (.jar).**

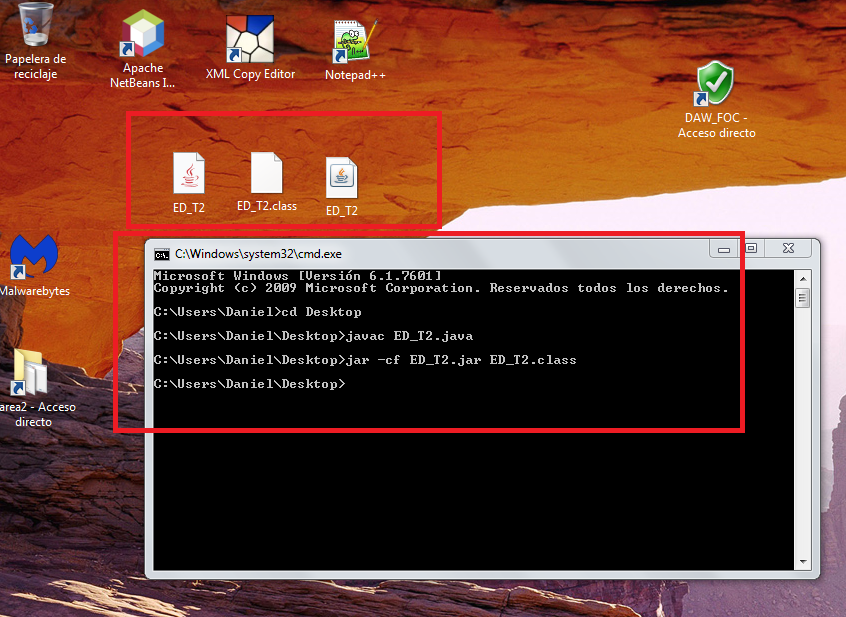


**A continuación, en consola, me sitúo en la ubicación del archivo con el comando**

**cd Desktop y ejecuto los comandos:**

**javac ED\_T2.java (para crear el archivo .class)**

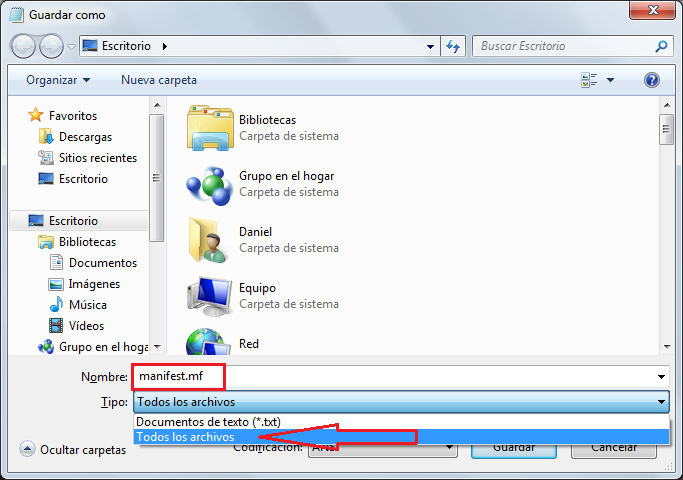
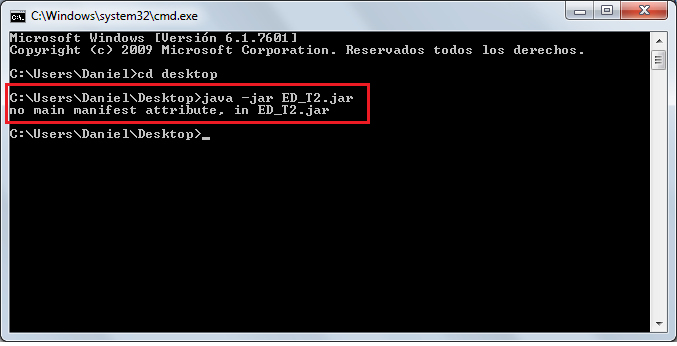
**jar –cf ED\_T2.jar ED\_T2.class (para crear el archivo .jar)**

****

**Llegado a este punto, si trato de ejecutar el archivo .jar, la consola me dirá que falta el archivo manifest. He de crearlo con extensión .mf, y tiene que incluir 2 líneas de texto:**

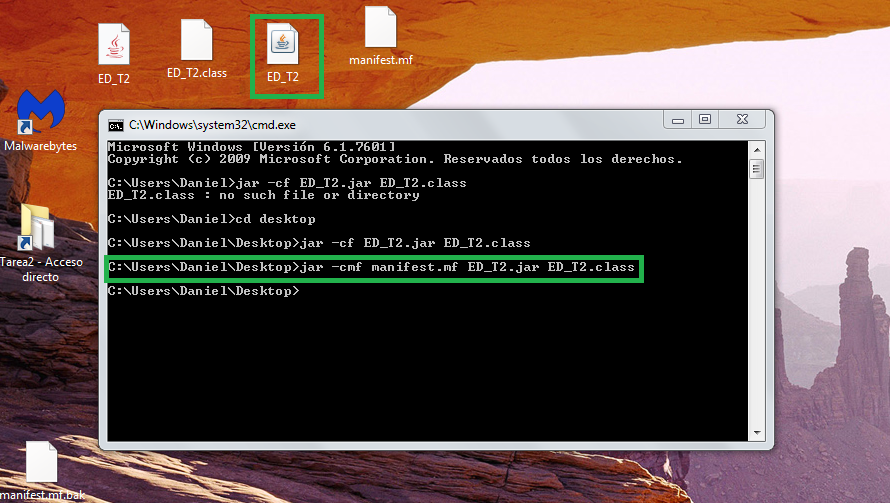
**“Class-Main: ED\_T2.jar” (para indicar cuál es la clase Main)**

**“Sealed: True”**

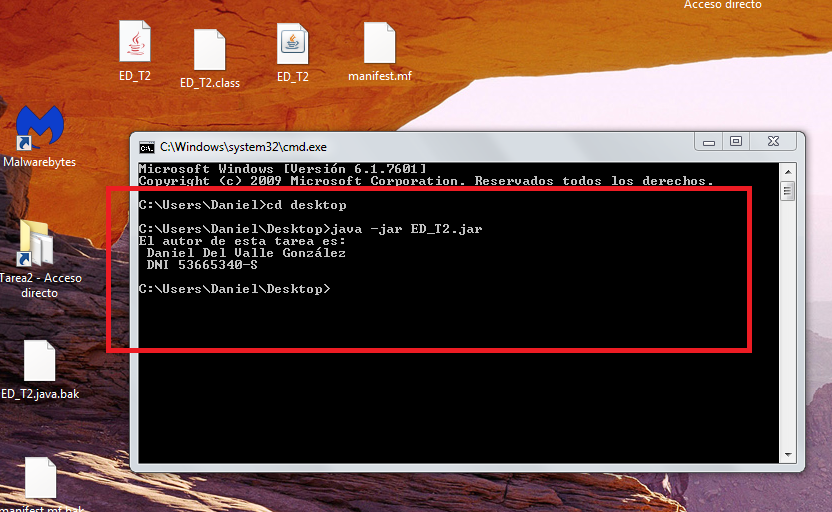
**Lo hago usando también la libreta de apuntes, pero cambiando la extensión a .mf y seleccionando “todos los archivos”. Se llamará manifest.mf. Es necesario para la compilación (contiene instrucciones para el compilador).** 

A continuación, ejecuto el código que añade el manifest:

**Jar –cmf ED\_T2.jar ED\_T2.class,** y el archivo .jar es creado



Por último, ejecuto el archivo con el comando **java –jar ED\_T2.jar**

****