

## UD02. Entornos de desarrollo Multiplataforma y Web.



Materiales actualizados por el profesorado de la Junta de Andalucía



[Aviso legal](#)



Materiales desarrollados inicialmente por el Ministerio de Educación, Cultura y Deporte

[Aviso Legal](#)

# Indice

---

- 1.- Introducción a los entornos de desarrollo.
  - 1.1.- Funciones de un entorno de desarrollo.
  - 1.2.- Estructura de entornos de desarrollo.
  - 1.3.- Entornos integrados libres y propietarios.
  - 1.4.- Clasificación de los entornos de desarrollo.
- 2.- Entornos integrados de desarrollo comunes.
  - 2.1.- NetBeans.
    - 2.1.1.- Instalación de Netbeans.
    - 2.1.2.- Configuración y personalización de NetBeans.
    - 2.1.2.1- Actualización y mantenimiento de NetBeans.
    - 2.1.3.- Uso básico de Netbeans.
      - 2.1.3.1.- Ejemplo de edición de código con Netbeans.
      - 2.1.3.2.- Edición de programas en NetBeans.
      - 2.1.3.3.- Generación de ejecutables en NetBeans.
    - 2.1.4.- Gestión de módulos en NetBeans.
      - 2.1.4.1.- Añadir módulos en Netbeans.
      - 2.1.4.2.- Eliminar módulos en Netbeans.
  - 2.2.- Eclipse.
    - 2.2.1.-Instalación y utilización de Eclipse.
    - 2.2.2.- Gestión de módulos en Eclipse. El marketplace.
  - 3.- Herramientas CASE para el desarrollo, prueba y documentación del código.
    - 3.1.-Herramientas de los navegadores web.
      - 3.1.1.- Safari.
      - 3.1.2.- Microsoft Edge.
      - 3.1.3.- Mozilla Firefox.
      - 3.1.4.- Firebug.
      - 3.1.5- Inspector de Chrome.
    - 3.2.- Plugin UML para netbeans
    - 3.3.- Software para generar documentación del código fuente.

# 1.- Introducción a los entornos de desarrollo.

---

Ya hemos tratado las fases en el proceso de desarrollo de software.

Una de ellas era la fase de **codificación**, en la cual se hacía uso de algún lenguaje de programación para pasar todas las acciones que debía llevar a cabo la aplicación a algún lenguaje que la máquina fuera capaz de entender y ejecutar.

También se ha hecho alusión a herramientas de apoyo al proceso de programación. A continuación vamos a analizar, instalar y ejecutar estas herramientas para entender su acción y efecto.

Muchas personas aprenden a programar utilizando un editor de texto simple, compilador y depurador. Pero la mayoría, finalmente, terminan haciendo uso de algún entorno de desarrollo integrado para crear aplicaciones.

Un entorno integrado de desarrollo (**IDE**), es un tipo de software compuesto por un conjunto de herramientas de programación. En concreto, el IDE se compone de:

- ✓ Editor de código de programación.
- ✓ Compilador.
- ✓ Intérprete.
- ✓ Depurador.
- ✓ Constructor de interfaz gráfico.

Un **compilador** traduce el programa escrito en un lenguaje de alto nivel (código fuente) a un lenguaje común, usualmente lenguaje de máquina, aunque también puede ser traducido a un código intermedio (bytecode).

Aunque no todos los lenguajes de programación requieren ser compilados y enlazados para ser ejecutados en la máquina. Los lenguajes interpretados utilizan un programa llamado intérprete.

Un **intérprete** analiza las instrucciones línea a línea y las traduce a código ejecutable conforme se va necesitando en el momento de la ejecución.

Un **depurador** sirve para detectar y corregir errores que se producen en tiempo de ejecución. Permite ejecutar el programa línea a línea, detenerse en un determinado lugar a través de los puntos de interrupción (breakpoints), inspeccionar los valores de las variables en cada paso de la ejecución o incluso modificarlos.

Los primeros entornos de desarrollo integrados nacieron a principios de los años 70, y se popularizaron en la década de los 90. Tienen el objetivo de ganar fiabilidad y tiempo en los proyectos de software. Proporcionan al programador una serie de componentes con la misma interfaz gráfica, con la consiguiente comodidad, aumento de eficiencia y reducción de tiempo de codificación.

Normalmente, un IDE está dedicado a un determinado lenguaje de programación. No obstante, las últimas versiones de los IDE tienden a ser compatibles con varios lenguajes (por ejemplo, Eclipse, NetBeans, Microsoft Visual Studio) mediante la instalación de plugins adicionales.

En este tema, nuestro interés se centra en conocer los entornos de desarrollo, los tipos, en función de su licencia y del lenguaje de programación hacia el cual están enfocados. Instalaremos NetBeans y veremos cómo se configura y cómo se generan ejecutables, haciendo uso de sus componentes y herramientas.

## 1.1.- Funciones de un entorno de desarrollo.

---

Como sabemos, los entornos de desarrollo están compuestos por una serie de herramientas software de programación, necesarias para la consecución de sus objetivos. Estas herramientas son:

- ✓ Un editor de código fuente.
- ✓ Un compilador y/o un intérprete.
- ✓ Automatización de generación de herramientas.
- ✓ Un depurador.

**Las funciones de los IDE son:**

- ✓ Editor de código: coloración de la  sintaxis.
- ✓ Auto-completado de código, atributos y métodos de clases.
- ✓ Identificación automática de código.
- ✓ Herramientas de concepción visual para crear y manipular componentes visuales.
- ✓ Asistentes y utilidades de gestión y generación de código.
- ✓ Archivos fuente en unas carpetas y compilados a otras.
- ✓ Compilación de proyectos complejos en un solo paso.
- ✓ Control de versiones: tener un único almacén de archivos compartido por todos los colaboradores de un proyecto. Ante un error, mecanismo de auto-recuperación a un estado anterior estable.
- ✓ Soporta cambios de varios usuarios de manera simultánea.
- ✓ Generador de documentación integrado.
- ✓ Detección de errores de sintaxis en tiempo real.

**Otras funciones importantes son:**

- ✓ Ofrece refactorización de código: cambios menores en el código que facilitan su legibilidad sin alterar su funcionalidad (por ejemplo cambiar el nombre a una variable).
- ✓ Permite introducir automáticamente tabulaciones y espaciados para aumentar la legibilidad.
- ✓ Depuración: seguimiento de variables, puntos de ruptura y mensajes de error del intérprete.
- ✓ Aumento de funcionalidades a través de la gestión de sus  módulos y  plugins.
- ✓ Administración de las interfaces de usuario (menús y barras de herramientas).
- ✓ Administración de las configuraciones del usuario.

## 1.2.- Estructura de entornos de desarrollo.

---

Los entornos de desarrollo, ya sean libres o propietarios, están formados por una serie de componentes software que determinan sus funciones.



Estos componentes son:

**Editor de textos:** Resalta y colorea la sintaxis, tiene la función de autocompletar código, ayuda y listado de parámetros de funciones y métodos de clase. Inserción automática de paréntesis, corchetes, tabulaciones y espaciados.

**Compilador/intérprete:** Detección de errores de sintaxis en tiempo real. Características de [refactorización](#).

**Depurador:** Botón de ejecución y traza, [puntos de ruptura](#) y seguimiento de variables. Opción de depurar en [servidores remotos](#).

**Generador automático de herramientas:** Para la visualización, creación y manipulación de componentes visuales y todo un arsenal de asistentes y utilidades de gestión y generación código.

**Interfaz gráfica:** Nos brinda la oportunidad de programar en varios lenguajes con un mismo IDE. Es una interfaz agradable que puede acceder a innumerables [bibliotecas](#) y plugins, aumentando las opciones de nuestros programas.

## 1.3.- Entornos integrados libres y propietarios.

### Entornos Integrados Libres

Son aquellos con licencia de uso público.

No hay que pagar por ellos, y aunque los más conocidos y utilizados son Eclipse y NetBeans, hay bastantes más.

#### Tipos de entornos de desarrollo libres más relevantes en la actualidad.

IDE	Lenguajes que soporta	Sistema Operativo
NetBeans. Eclipse. Gambas. Anjuta. GNAT Studio.	C/C++, Java, JavaScript, PHP, Python. Ada, C/C++, Java, JavaScript, PHP. Basic. C/C++, Python, Javascript. Fortran.	Windows, Linux, Mac OS X. Windows, Linux, Mac OS X. Linux. Linux. Windows, Linux, Mac OS X.

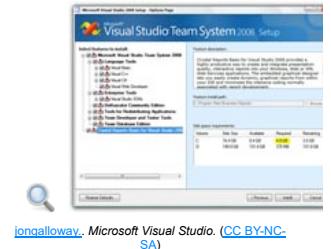
El aspecto de la licencia del IDE que se elija para el desarrollo de un proyecto es una cuestión de vital importancia. En su elección prevalecerá la decisión de los supervisores del proyecto y de la dirección de la empresa.

### Entornos Integrados Propietarios

Son aquellos entornos integrados de desarrollo que necesitan licencia. No son free software, hay que pagar por ellos.

El más conocido y utilizado es Microsoft Visual Studio, que usa el framework .NET y es desarrollado por Microsoft.

#### Tipos de entornos de desarrollo propietarios más relevantes en la actualidad.



[jongalloway](#), Microsoft Visual Studio. (CC BY-NC-SA)

IDE	Lenguajes que soporta	Sistema Operativo
Microsoft Visual Studio. FlashBuilder. C++ Builder. Turbo C++ profesional. JBuilder. JCreator. Xcode.	Basic, C/C++, C#. ActionScript. C/C++. C/C++. Java. Java. C/C++, Java.	Windows. Windows, Mac OS X. Windows. Windows. Windows, Linux, Mac OS X. Windows. Mac OS X.

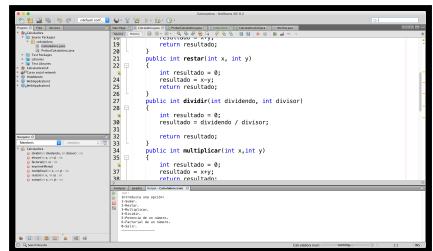
## 1.4.- Clasificación de los entornos de desarrollo.

Los entornos de desarrollo son un tipo de software que nos permite desarrollar aplicaciones facilitando la labor a través de herramientas y tareas predefinidas. Con el tiempo algunos han quedado como mas generales, pudiendo abarcar diversos lenguajes y tipos de aplicaciones, mientras que otros se han vuelto más específicos.

La utilización de un IDE nos permite desarrollar nuestro proyecto software de una forma mucho más ágil. Además, podemos utilizar las herramientas que incluya nuestro entorno de trabajo, como el depurador de código o el autoformateo del mismo.

Los entornos de desarrollo pueden clasificarse en base a numerosos factores. Por ejemplo podríamos clasificarlos en base al tipo de licencia como hemos visto antes, por ejemplo software libre o software privado, en función de la plataforma para la que esté disponible, si estamos ante un IDE multiplataforma o no o en función del lenguaje de programación que soporte, puesto que hay entornos de desarrollo para un lenguaje específico, como PHPStorm, o para múltiples lenguajes de programación, como NetBeans.

En esta unidad vamos a clasificar los distintos entornos de desarrollo en tres grupos. Primero veremos aquellos IDE que se utilicen indistintamente para el desarrollo de aplicaciones multiplataforma y aplicaciones web para posteriormente ver los entornos de desarrollo específicos en el desarrollo de aplicaciones multiplataforma así como los entornos de desarrollo específicos para el desarrollo de aplicaciones web.



En la siguiente tabla podemos ver los distintos entornos de desarrollo que vamos a estudiar a lo largo de la unidad, en base a la clasificación anteriormente citada.

### Entornos comunes DAM//DAW

Entornos de Desarrollo	Descripción
Netbeans	Es el entorno multilenguaje mas extendido y el mas integrado con Java. Lo puedes descargar desde la <a href="#">página oficial de Netbeans</a> .
Eclipse	Uno de los entornos mas populares multilenguaje y que además esta preparado para desarrollar entornos de desarrollo. Fue el primero en integrar la SDK de Android para desarrollo de aplicaciones móviles. Actualmente está perdiendo protagonismo, debido a que Oracle, propietario de Java, ha elegido a Netbeans como el IDE oficial de dicho lenguaje, y Android ha apostado por su propio IDE, Android Studio. <a href="#">Para descargar la última versión se puede acceder a la página oficial del proyecto</a> .
Visual Studio	Microsoft Visual Studio es un potente IDE multiplataforma que permite desarrollar aplicaciones en múltiples lenguajes y framework, tales como Node.js, javascript, C++, PHP y .NET entre otros. Podemos <a href="#">descargar Visual Studio de forma gratuita desde la página oficial del proyecto</a> .
Oracle SQL Developer	Oracle SQL Developer es un entorno de desarrollo integrado que simplifica el desarrollo y la gestión de bases de datos de Oracle. Con dicho IDE podemos desarrollar aplicaciones PL/SQL, realizar consultas a una base de datos y gestionar nuestras bases de datos a través del lenguaje SQL. <a href="#">El IDE lo podemos descargar desde la página oficial de Oracle</a> .
Monodevelop	Mono es un proyecto de código abierto en el que se desarrollan herramientas basadas en Linux que permiten desarrollar utilizando la tecnología .NET. Lo puedes descargar desde <a href="#">aquí</a> . Ademas desde <a href="#">aquí</a> puedes descargar el IDE Monodevelop para Mono.

### Entornos específicos para DAM

Entornos de Desarrollo	Descripción
Android Studio	El entorno de desarrollo implementado por Google para aplicaciones Android. Actualmente es la referencia para el desarrollo de este tipo de aplicaciones. Puedes descargarlo en su <a href="#">web oficial</a> .
XCode	XCode es el entorno de desarrollo para macOs y contiene las herramientas necesarias para desarrollar software para los distintos sistemas operativos de la marca Apple (macOS, iOS, watchOS y tvOS). Aunque es gratuito, solo puede instalarse en computadoras con sistema operativo macOS. Para más información <a href="#">podemos consultar la web oficial del proyecto</a> .
APPinventor	Permite el desarrollo de aplicaciones sencillas para Android con un entorno muy intuitivo. <a href="#">Podemos encontrar más información así como interesantes tutoriales en su página oficial</a> .
Unity	Unity es, además de un motor de videojuegos, una plataforma de desarrollo. Aunque la mayoría de versiones son de pago, podemos utilizar la "Personal" para poder desarrollar nuestros propios videojuegos. La mayoría de juegos que se desarrollan hoy en día para móviles y para escritorio, están desarrollados bajo Unity. <a href="#">En la página oficial podemos encontrar tutoriales, además de descargarnos el software necesario para comenzar a desarrollar en Unity</a> .
JMonkey	JMonkey es un motor de videojuegos libre orientado al desarrollo de juegos en tres dimensiones. Está escrito en Java y tiene su propio IDE de desarrollo. Lo podemos descargar desde <a href="#">aquí</a> .

## Entornos de Desarrollo de Aplicaciones Web

Entornos de Desarrollo	Descripción
Atom	Es un IDE de código abierto que fue lanzado a través de la plataforma GitHub. Al tratarse de una herramienta OpenSource (Código abierto) está provocando que muchos programadores lo estén mejorando y que obtenga el apoyo de una importante comunidad de desarrolladores. Esta disponible para la mayoría de las plataformas (Windows, macOs, GNU/Linux) es fácil de configurar y personalizar y facilita la integración con el framework Node.js. <a href="#">En la página oficial del proyecto podemos encontrar más información sobre el mismo, así como descargarlo.</a>
PhpStorm	PhpStorm es un IDE comercial orientado al desarrollo de aplicaciones web en php. Una de sus principales ventajas radica en la integración con framework de desarrollos de php como Symfony. Aunque sea de pago, podemos obtener una licencia gratuita para estudiantes y profesores. <a href="#">En la página oficial de la aplicación podemos encontrar más información y solicitar nuestra licencia libre como estudiantes/profesores.</a>



### ¿Es lo mismo un IDE que un framework?

Aunque la respuesta pueda parecer obvia, debido a la traducción que se hace al castellano de ambos términos queremos aclarar ambos conceptos.

Un **IDE** es un entorno de desarrollo integrado que consiste en una aplicación software que nos facilita el proceso de programación. Generalmente un IDE está formado por un editor de texto, herramientas de desarrollo automático, un depurador, y en función del tipo de lenguaje de programación, un compilador. También es muy común que el IDE incluya entornos de ejecución para validar el software desarrollado. Por ejemplo, en el caso de Android Studio, podemos ejecutar nuestra app en un terminal virtual de Android.

Un **framework** es un entorno de trabajo (veamos que por la traducción al castellano podría llevarnos a confusión) que nos proporciona un conjunto de herramientas, librerías y un patrón base que nos facilita el proceso de desarrollo. De forma genérica podríamos definirlo como una estructura de código base a partir de la cual podemos desarrollar un proyecto mucho más complejo, utilizando las herramientas que nos proporciona el framework utilizado. Generalmente un framework nos permite desarrollar software de una forma más sencilla, que en ocasiones significa desarrollar programas con un menor número de líneas de código. En el desarrollo web existen framework muy populares, como Angular.js, Symfony o Bootstrap.

Por ejemplo, PHPStorm es un IDE para desarrollar aplicaciones web con él podríamos hacer pequeñas aplicaciones en php. Symfony es un framework basado en php para desarrollar proyectos web, podríamos desarrollar una aplicación con dicho framework utilizando phpStorm o cualquier otro IDE de desarrollo que sea compatible con PHP.

## 2.- Entornos integrados de desarrollo comunes.

---

En el apartado anterior hemos visto que existen multitud de entornos de desarrollo integrados. A lo largo del siguiente epígrafe nos centraremos en los entornos comunes al desarrollo de aplicaciones web y multiplataforma.



**NetBeans**



**eclipse**



**Visual Studio**

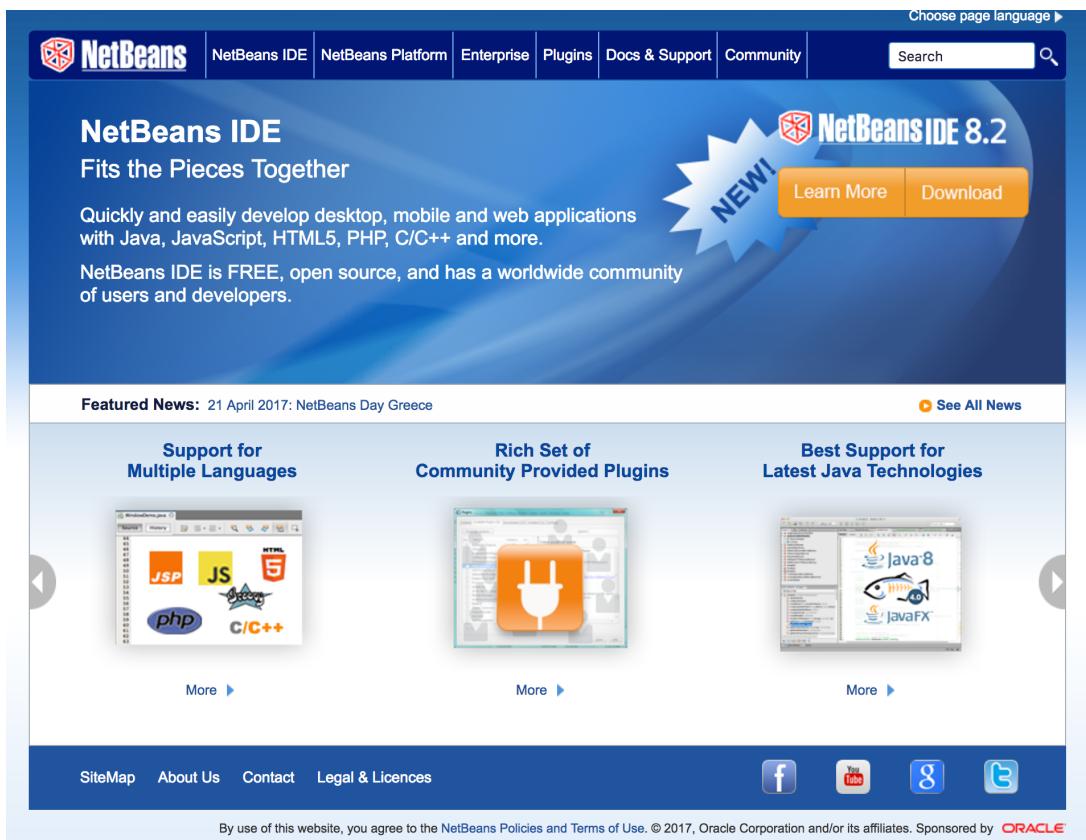
## 2.1.- NetBeans.

Netbeans es un entorno de desarrollo integrado libre desarrollado principalmente para el lenguaje de programación Java, aunque existen numerosos módulos que permiten desarrollar en otros lenguajes de programación como PHP.

NetBeans fue desarrollado por Sun MicroSystems y tras la adquisición de esta empresa por Oracle Corporation, pasó a ser propiedad de Oracle que es la empresa encargada de su mantenimiento.

El NetBeans IDE soporta el desarrollo de todos los tipos de aplicación Java (J2SE, web, EJB y aplicaciones móviles). Además permite aumentar su funcionalidad mediante la instalación de módulos. Entre dichos módulos podríamos destacar las herramientas para la generación de esquemas XML, el modelado UML o los plugins que dan soporte a otros lenguajes de programación como PHP o C++.

Podemos descargarnos Netbeans desde [la página oficial del proyecto.](#)



Javier Mejías Real. Aspecto de la web oficial del proyecto netbeans. ([CC0](#))

## 2.1.1.- Instalación de Netbeans.

Para instalar NetBeans bajo Windows, tendríamos que acceder a la página oficial de netbeans (<http://netbeans.org>) y una vez dentro acceder a la zona de descargas.

The screenshot shows the official NetBeans website. At the top, there's a navigation bar with links for NetBeans IDE, NetBeans Platform, Enterprise, Plugins, Docs & Support, Community, and a search bar. Below the navigation, the main content area features the NetBeans IDE logo and the tagline "Fits the Pieces Together". It highlights the IDE's ability to quickly develop desktop, mobile, and web applications using Java, JavaScript, HTML5, PHP, C/C++, and more. It also emphasizes that NetBeans IDE is FREE, open source, and has a worldwide community of users and developers. A prominent orange button labeled "Download" is visible, with a large black arrow pointing directly at it. At the bottom of the page, there's a "Featured News" section and a link to "See All News".

Una vez accedamos a la opción de descarga, se nos mostrarán las distintas configuraciones de NetBeans que podemos descargarnos para la plataforma desde la que estamos accediendo.

The screenshot shows the "NetBeans IDE 8.2 Download" page. The top navigation bar includes links for NetBeans IDE, NetBeans Platform, Plugins, Docs & Support, Community, Partners, and a search bar. Below the navigation, the page title is "NetBeans IDE 8.2 Download" with version options 8.1 | 8.2 | Development | Archive. There are fields for "Email address (optional)" and "Subscribe to newsletters:" with checkboxes for "Monthly" and "Weekly". The "IDE Language" is set to "Español" and the "Platform" is "Windows". A note states: "Note: Greyed out technologies are not supported for this platform." Below this, there's a table titled "NetBeans IDE Download Bundles in community contributed languages<sup>1</sup>". The table lists various technologies and their compatibility across different platforms. At the bottom of the table, there are several "Download" buttons for different file sizes: "Download", "Download", "Download x86", "Download x86", "Download x86", "Download x86", "Download x64", "Download x64", "Download x64", and "Download". Below the table, there's a note: "\* You can add or remove packs later using the IDE's Plugin Manager (Tools | Plugins)." Another note says: "HTML5JS, PHP and C/C++ NetBeans bundles include Java Runtime Environment and do not require a separate Java installation." A third note states: "JDK 7 and later versions are required for installing and running the Java SE, Java EE and All NetBeans Bundles. You can download standalone JDK or download the latest JDK with NetBeans IDE Java SE bundle." A fourth note provides instructions for starting development: "You can start developing applications based on the NetBeans Platform using the NetBeans IDE for Java SE. Learn more about the NetBeans Platform. NetBeans source code and binary builds without bundled runtimes are also available in zip file format. See also instructions on how to build the IDE from sources or installation instructions." A blue box contains "Important Legal Information": "NetBeans Community Distributions are available under a Dual License consisting of the Common Development and Distribution License (CDDL) v1.0 and GNU General Public License (GPL) v2. Such distributions include additional components under separate licenses identified in the License file. See the Third Party License file for external components included in NetBeans and their associated licenses." At the very bottom, there are links for SiteMap, About Us, Contact, Legal & Licences, and social media icons for Facebook, YouTube, Google+, and Twitter. A footer note says: "By use of this website, you agree to the NetBeans Policies and Terms of Use. © 2012, Oracle Corporation and/or its affiliates. Sponsored by ORACLE".

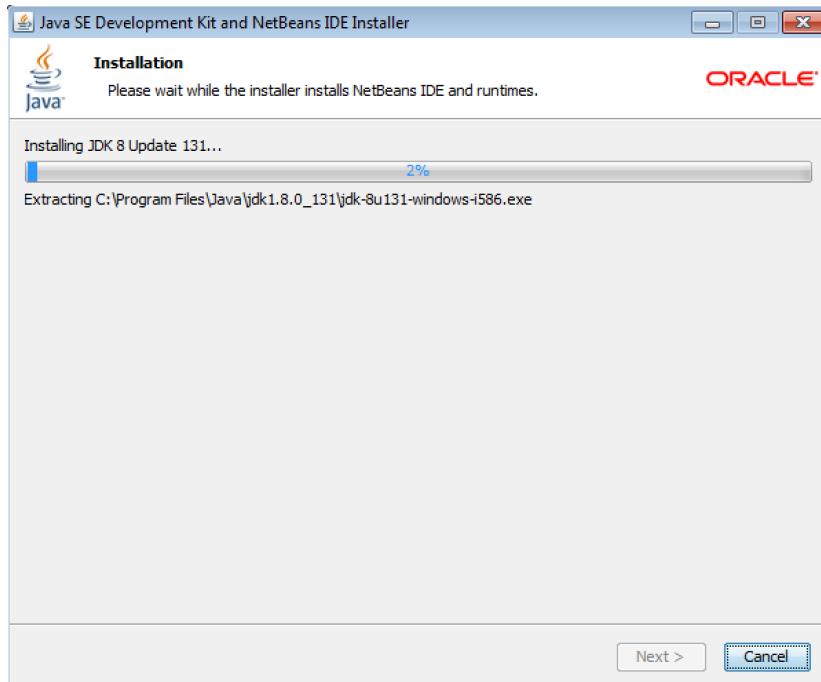
En la parte superior podemos elegir la versión de NetBeans que queremos descargarnos, así como la plataforma en la que lo vamos a instalar.



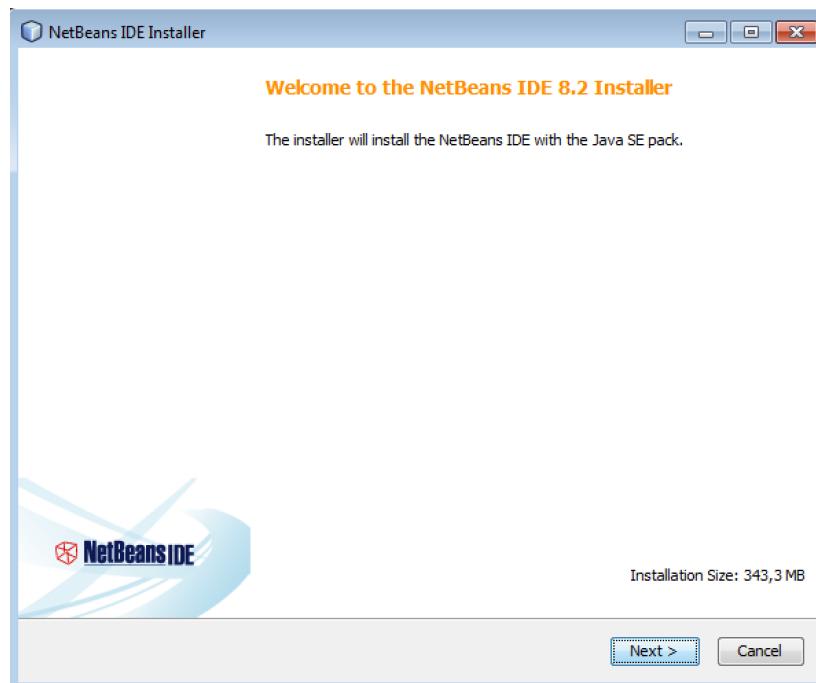
Si seleccionamos el Sistema Operativo Windows o Mac OS, el proceso de instalación consistirá en el proceso normal de instalación de un programa en dichos sistemas operativos.

## Instalación en Microsoft Windows

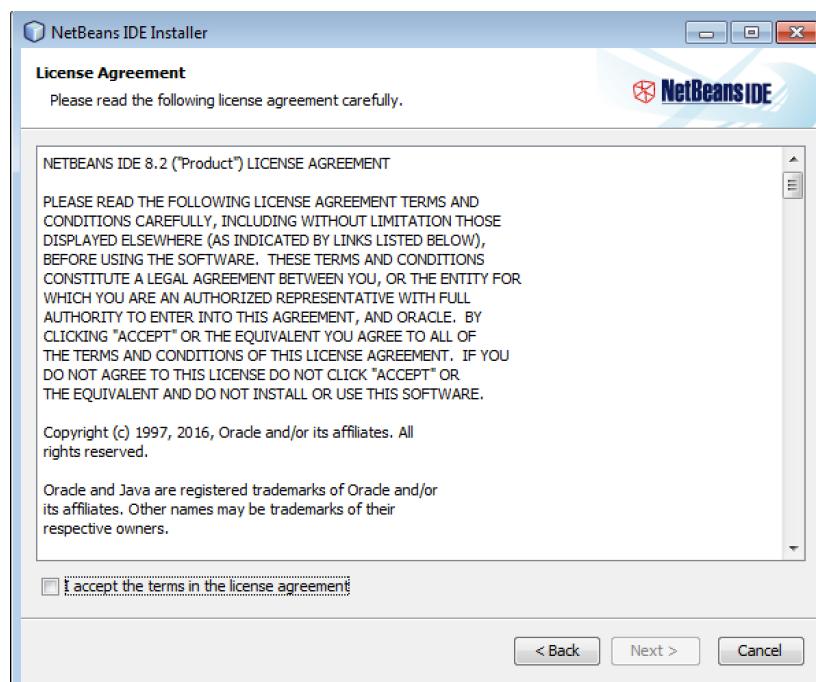
Los usuarios del sistema operativo Microsoft windows, estarán familiarizados con el proceso de instalación, puesto que se adapta al proceso estándar en el que basta con confirmar una serie de pasos. Si no tenemos las librerías de Java en nuestra computadora, podemos descargarlos la versión de NetBeans que viene ya con las librerías de Java.



Si ya tenemos instalada la JDK de Java, bastará con que nos descargamos la versión de Netbeans que mejor se adapte a nuestras necesidades e instalarla.

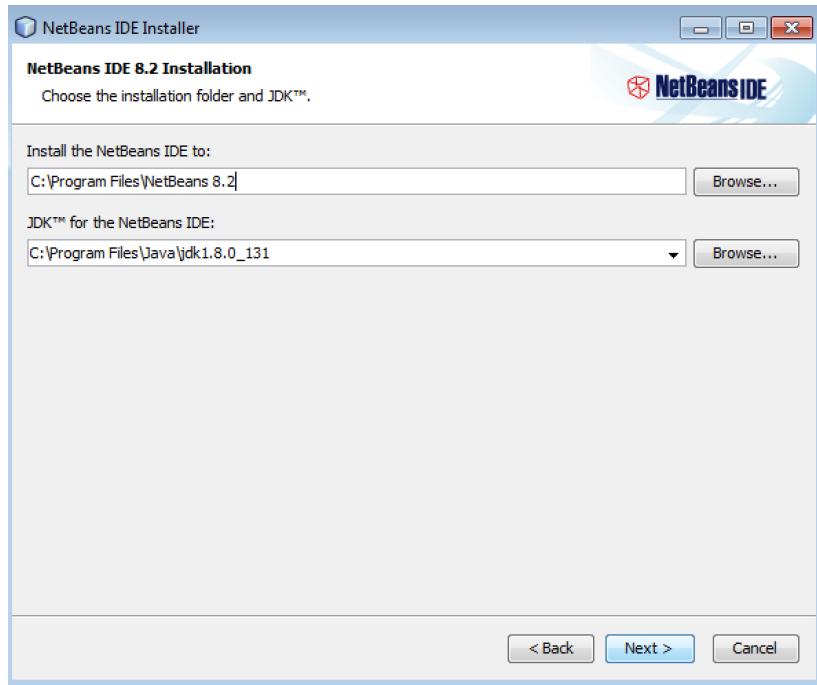


Javier Mejías. Proceso de instalación de Netbean ([CC0](#))

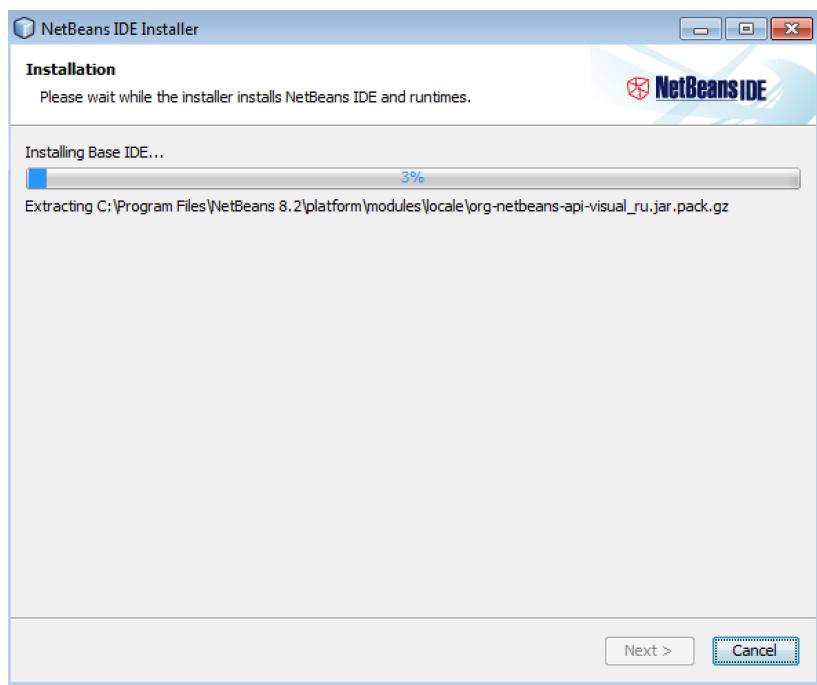


Javier Mejías. Proceso de instalación de Netbean ([CC0](#))

Una vez hemos aceptado la licencia de uso, nos preguntará donde tenemos instalado la JDK de Java. Si la tenemos instalada, nos pondrá automáticamente la ruta.



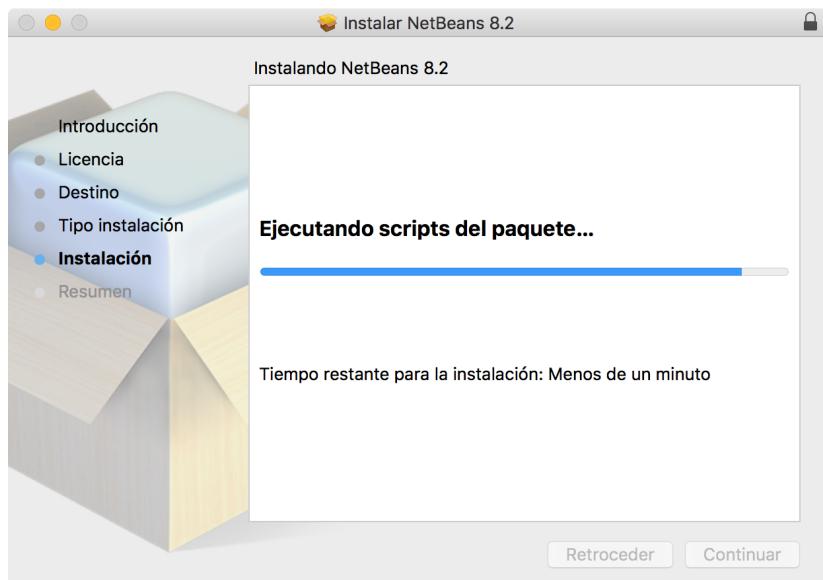
Javier Mejias. Proceso de instalación de Netbean ([CC0](#))



Javier Mejias. Proceso de instalación de Netbean ([CC0](#))

## Instalación en MacOs

Si lo instalamos en el sistema operativo Mac Os, el proceso será como el que se muestra a continuación:



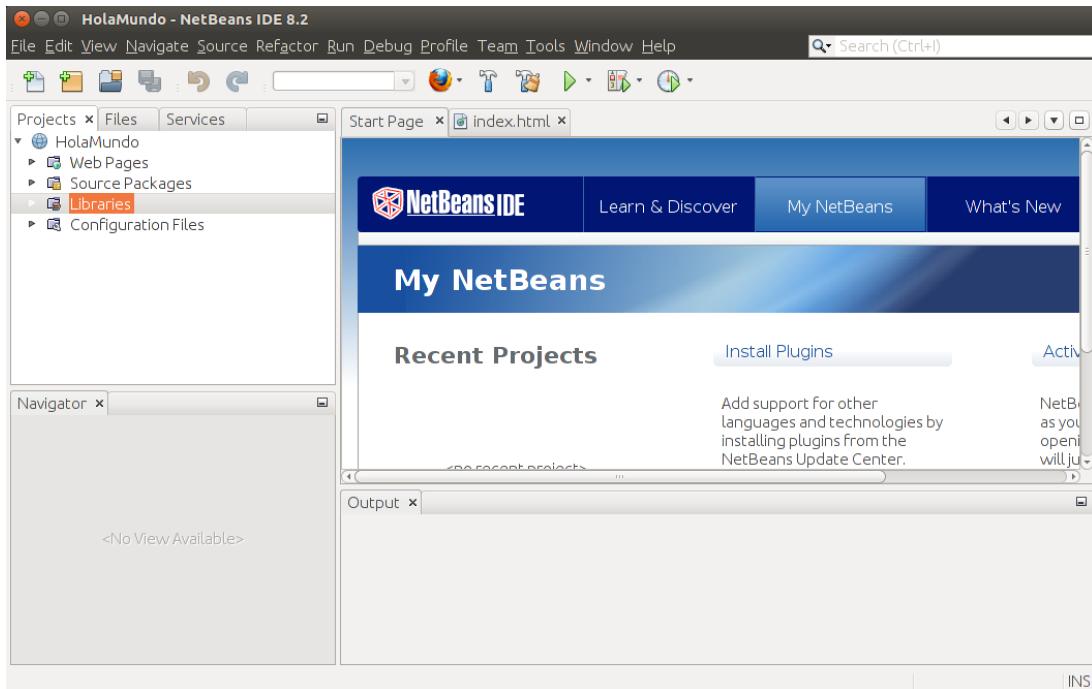
Javier Mejías. *Proceso de instalación de Netbean* ([CC0](#))

## Instalación en GNU/Linux

Si la instalación la vamos a realizar en GNU/Linux, el proceso de instalación será algo más complejo. Para ello, una vez hayamos descargado el paquete de NetBeans, tendremos que marcar el archivo como ejecutable desde la línea de comandos.

```
chmod +x netbeans-8.2-javase-linux.sh
sudo ./netbeans-8.2-javase-linux.sh
```

No es necesario modificar ningún parámetro durante el proceso de instalación. Una vez finalizado, puedes acceder a NetBeans mediante una nueva entrada que se ha generado en el menú o a través del icono que aparecerá en el escritorio.

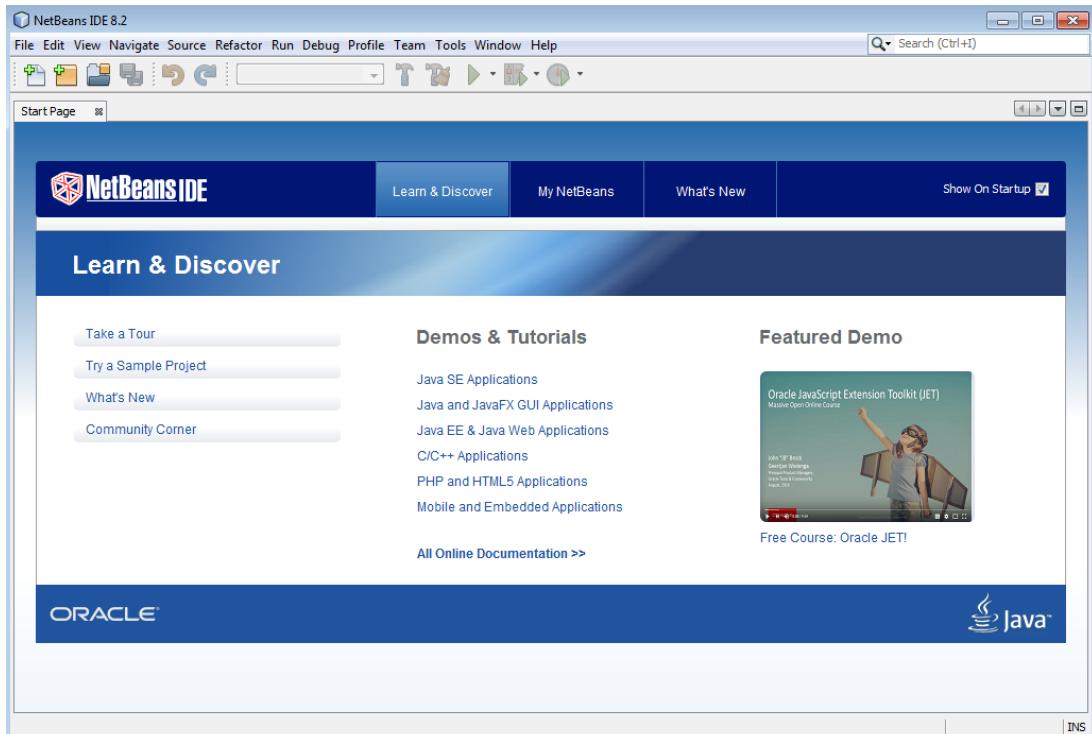


Javier Mejías. *NetBeans corriendo bajo GNU/Linux* ([CC0](#))

Recuerde que para poder instalar NetBeans en su computadora, es necesario que tenga instalada la Java SE Development Kit (JDK), la cual puede descargarse desde la página oficial de Java.

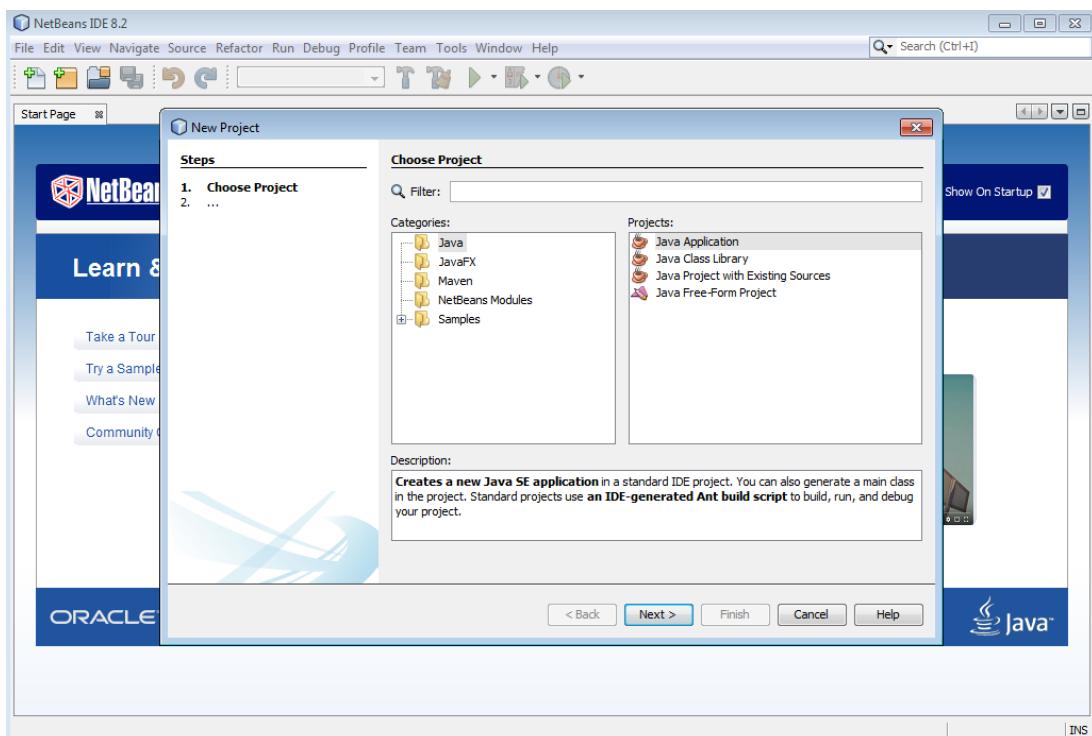
<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

Una vez tengamos instalado NetBeans ya podemos comenzar a utilizarlo.



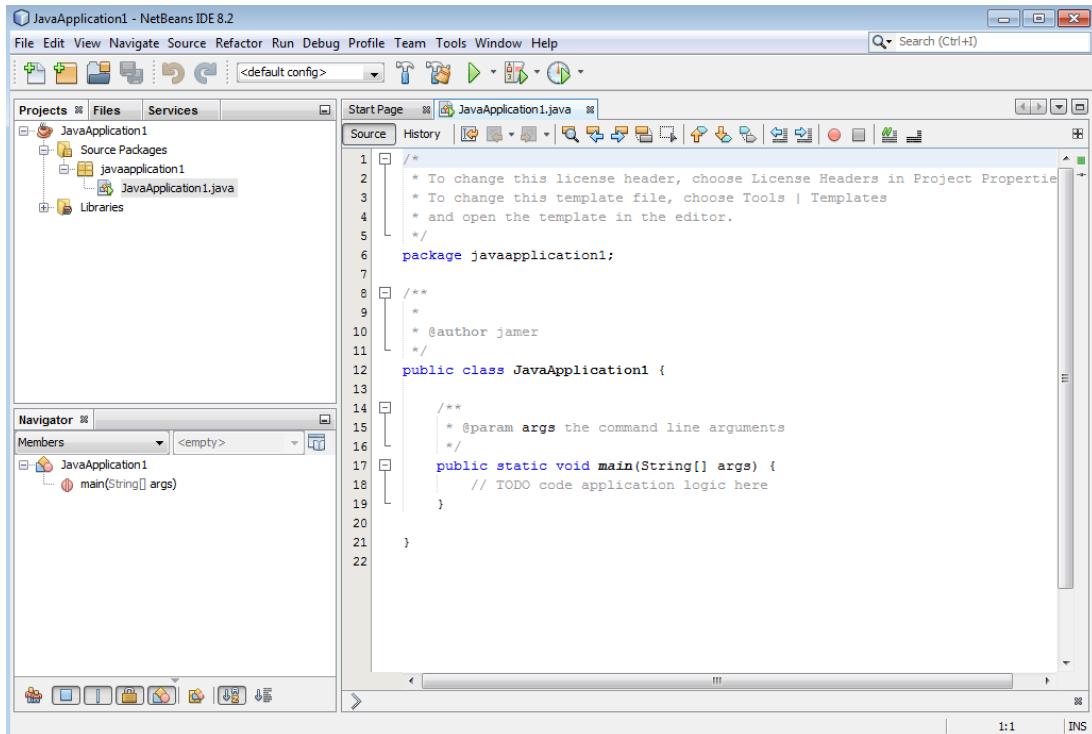
Javier Mejías. Ejecución de Netbeans (CC0)

Una vez hemos arrancado Netbeans por primera vez, podemos crear un proyecto accediendo al menú File->New Project y nos aparecerán los distintos tipos de proyectos que podemos crear.



Javier Mejías. Opciones de NetBeans a la hora de crear un proyecto. (CC0)

Al seleccionar el proyecto de tipo Java, NetBeans nos creará la estructura básica del mismo y ya estaremos preparados para comenzar a programar nuestro proyecto.



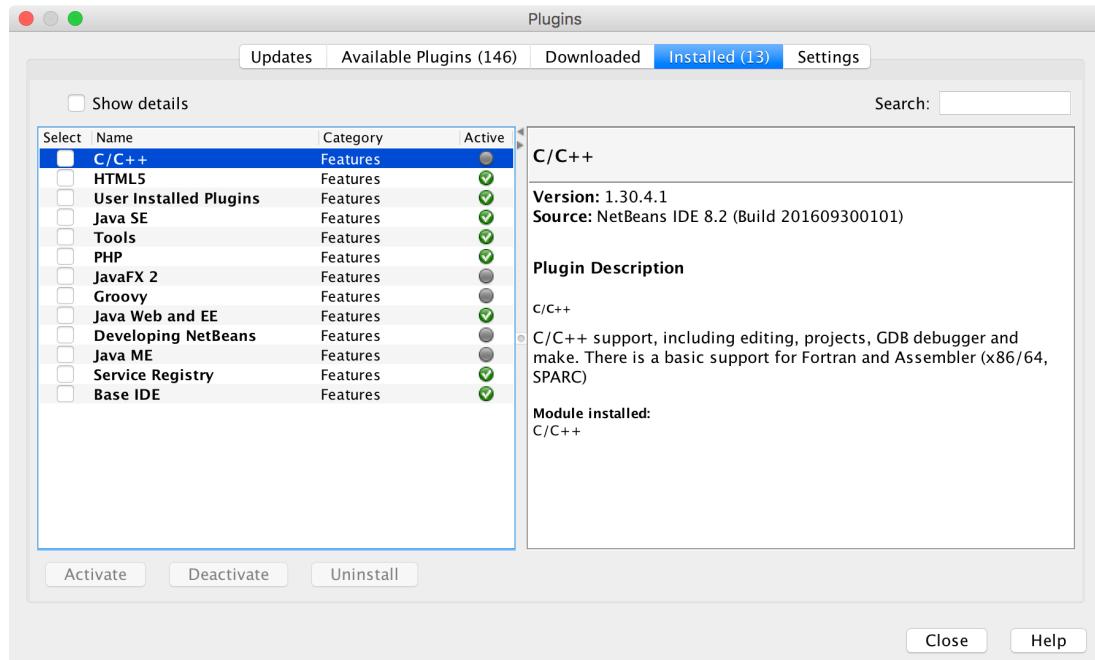
Javier Mejías. Aspecto del IDE NetBeans con un proyecto de Java (CC0)

Una vez tenemos instalado NetBeans, podemos instalarle algún plugin desde la sección "Plugins" colocada en la zona superior del menú principal de la página.

Plugin name	Category	NetBeans versions	Description	Owner	Last updated	Avg. Rating	Downloads	On UC
Sublime Theme	Web Applications , Version Control Systems, Utilities	8.0, 7.3	Sublime Text default theme for Netbeans IDE.	motammam	2014-09-24	★★★★★ [4.85/5]	414159	
JS CSS Minify Compress	Web Applications , Utilities, Building	8.1, 8.0, 7.4, 7.3	Plugin to minify JS, CSS and HTML for quicker page load times by reducing the size of javascript files (Other webapp optimization such as image BASE64 encode/decode , image compression)	ChrisLE	2017-03-22	★★★★★ [4.75/5]	40595	7.4, 8.0, 8.1
ResinTemplateModule	Web Applications , Utilities	7.0, 6.9	Resin Java & PHP Template	tdc.paul	2011-07-11	★★★★★ [0/5]	5156	
JSLint	Web Applications , Testing	8.0, 7.4, 7.3, 7.1, 7.0	The JavaScript Code Quality Tool	lomatek	2015-02-06	★★★★★ [4.63/5]	27947	7.0, 7.1, 7.3, 7.4, 8.0
TypoScript Support for NetBeans IDE	Web Applications , Programming Languages, Editing	8.1, 8.0, 7.4, 7.3, 7.2	This Project aims to provide support for the TypoScript Language used by TYPO3. Project wizard, Create theme(Underscores) action, Hyperlink navigation and code completion for filter and action function. Shortcut nodes for theme and output	dfranek	2017-03-16	★★★★★ [4.55/5]	10532	7.2, 7.3, 7.4, 8.0, 8.1
PHP WordPress Blog/CMS	Web Applications , Programming Languages	8.2, 8.1, 8.0, 7.4, 7.3	Support for WordPress. This plugin provides the following features. New Wordpress Project wizard, Create theme(Underscores) action, Hyperlink navigation and code completion for filter and action function. Shortcut nodes for theme and output	junichi11	2016-10-04	★★★★★ [4.31/5]	32233	7.4, 8.0, 8.1

Javier Mejías Real. Zona de plugins de la página de NetBeans. (CC0)

También podemos instalar los plugins desde el mismo IDE. Una vez lo tengamos instalado podemos acceder a la opción Tools->Plugins y se nos mostrará una ventana con los plugins que tenemos instalados, las posibles actualizaciones, así como la posibilidad de buscar e instalar plugins adicionales.



Javier Mejias Real. Pantalla para configurar los plugins de NetBeans (CC0)

## 2.1.2.- Configuración y personalización de NetBeans.

Una vez tenemos instalado nuestro entorno de desarrollo podemos acceder a personalizar su configuración.

Al abrir un proyecto existente, o bien crear un nuevo proyecto, seleccionaremos un desplegable con el nombre de "configuración" desde el que podremos personalizar distintas opciones del proyecto.

Podemos personalizar la configuración del entorno sólo para el proyecto actual, o bien para todos los proyectos, presentes y futuros.

Parámetros configurables del entorno:

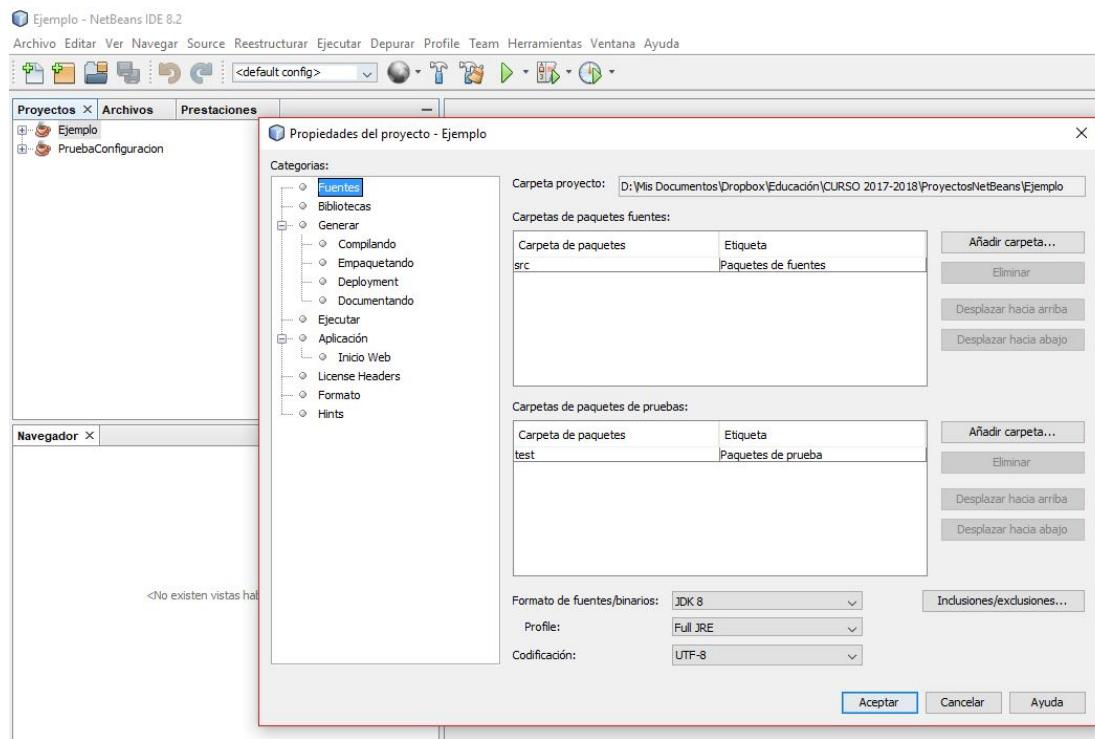
- ✓ Carpeta o carpetas donde se alojarán todos los archivos de los proyectos (es importante la determinación de este parámetro, para tener una estructura de archivos ordenada).
- ✓ Carpetas de almacenamiento de paquetes fuente y paquetes prueba.
- ✓ Administración de la plataforma del entorno de desarrollo.
- ✓ Opciones de la compilación de los programas: compilar al grabar, generar información de depuración.
- ✓ Opciones de **empaquetado** de la aplicación: nombre del archivo empaquetado (con extensión .jar, que es la extensión característica de este tipo de archivos empaquetados) y momento del empaquetado.
- ✓ Opciones de generación de documentación asociada al proyecto.
- ✓ Descripción de los proyectos, para una mejor localización de los mismos.
- ✓ Opciones globales de formato del editor: número de espaciados en las sangrías, color de errores de sintaxis, color de etiquetas, opción de autocompletado de código, propuestas de insertar automáticamente código.
- ✓ Opciones de combinación de teclas en teclado.
- ✓ Etc.

Accedemos a NetBeans y entramos en la página principal de la aplicación.

Para entrar a la aplicación podemos seleccionar "Nuevo Proyecto" y, una vez abierto, personalizar la configuración de NetBeans para ese proyecto.

En la barra de iconos de la aplicación, seleccionamos el desplegable de configuración

Seleccionamos "personalizar" y nos aparecerá la siguiente ventana (también podemos acceder desde el menú contextual del proyecto, en la opción "Establecer la configuración" y a continuación "personalizar"):



© Captura de pantalla de Netbeans, Edición de programas en NetBeans

Aquí vemos todo lo que podemos personalizar de la aplicación:

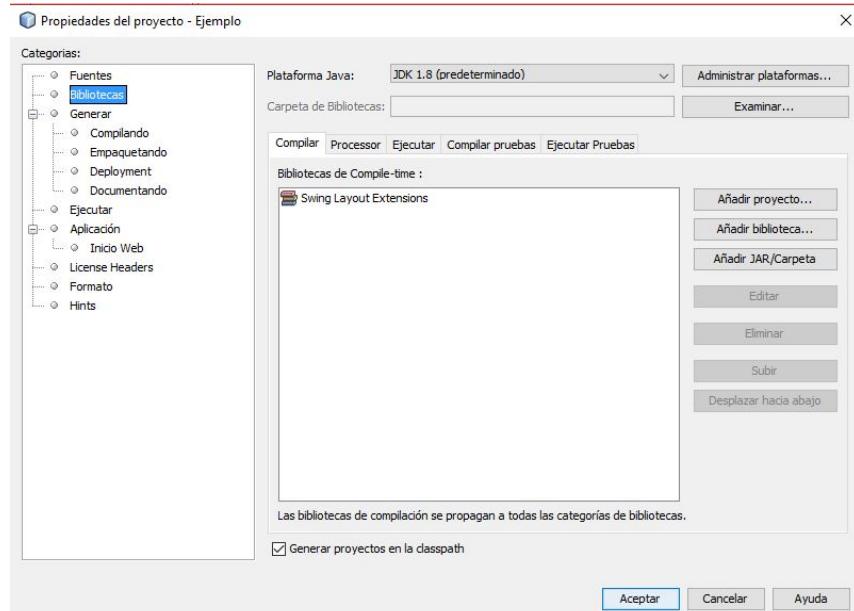
- ✓ Fuentes.
- ✓ Bibliotecas.
- ✓ Generación de código.
- ✓ Ejecución de código.
- ✓ Opciones de la aplicación.
- ✓ Formato del código en el editor de textos.

## Fuentes:

Podemos modificar:

- ✓ La carpeta que contendrá el proyecto
- ✓ La carpeta que almacenará los paquetes fuentes
- ✓ La carpeta que contendrá los paquetes prueba

## Bibliotecas:



© Captura de pantalla de Netbeans. Edición de programas en NetBeans

Desde esta ventana podemos elegir la plataforma de la aplicación.

Toma por defecto el JDK, pero se puede cambiar si se quiere, siempre y cuando sea compatible con la versión de NetBeans utilizada.

## Generar. Compilando el proyecto.

Las opciones que nos puede interesar modificar en cuanto a la compilación del programa entre otras son:

- ✓ Compilar al grabar: al guardar un archivo se compilará automáticamente.
- ✓ Generar información de depuración: para obtener la documentación asociada.
- ✓ Enable annotation processing: permitir anotaciones durante el proceso.

También podemos agregar anotaciones concretas para el proceso de compilación y añadir opciones de proceso que, según las características del proyecto, puedan ser de interés para nosotros.

## Generar. Empaquetando el proyecto.

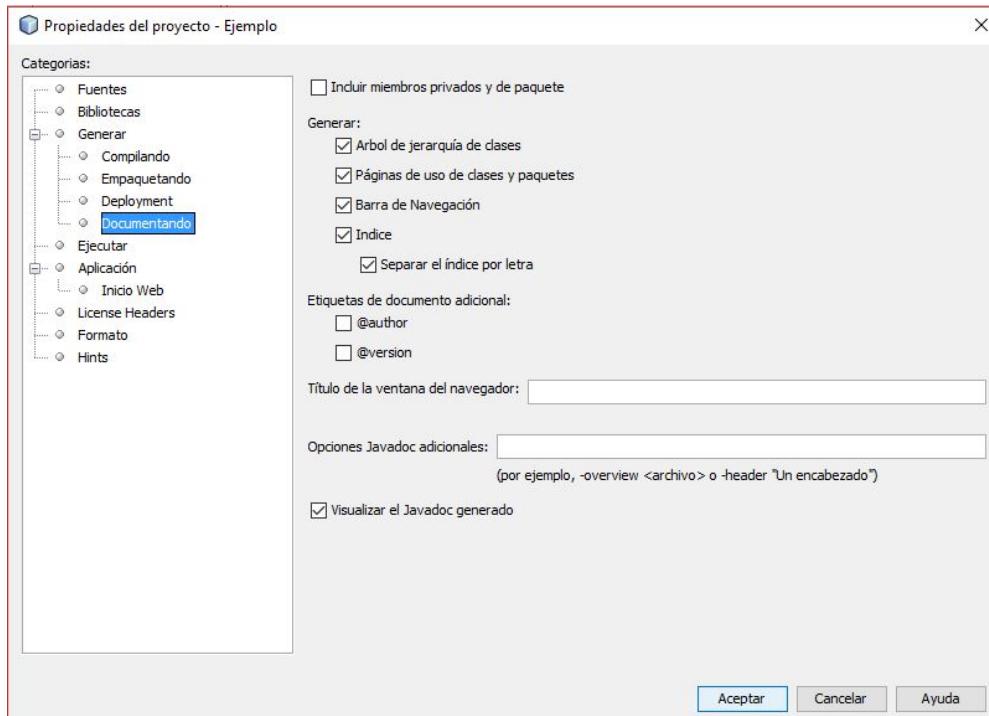
Las aplicaciones resultado de la compilación del código deben ser empaquetadas antes de su distribución, con objeto de tener un único archivo, generalmente comprimido, que contenga en su interior todos los archivos de instalación y configuración necesarios para que la aplicación pueda ser instalada y desarrollada con éxito por el usuario cliente.

En esta opción podemos indicar si queremos que el archivo resultante del empaquetado sea comprimido.

También podemos elegir que el archivo empaquetado se construya tras la compilación, que es lo habitual (por eso esta opción aparece como predeterminada) y seleccionar si queremos que incluya las librerías de las que depende el proyecto.

## Generar. Documentando el proyecto.

Como ya vimos en la unidad anterior, la documentación de aplicaciones es un aspecto clave que no debemos descuidar nunca. NetBeans nos ofrece una ventaja muy considerable al permitirnos obtener documentación de la fase de codificación de los programas de forma automática.

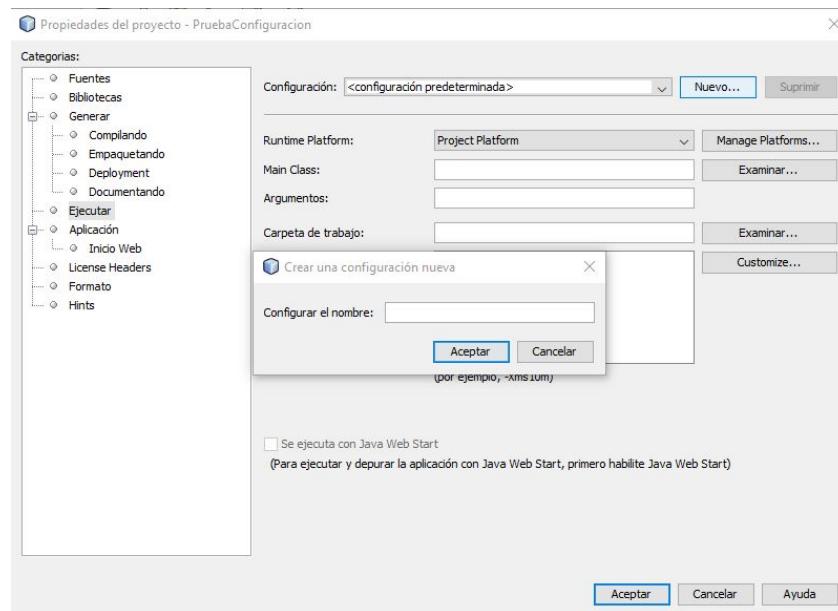


[© Captura de pantalla de Netbeans. Edición de programas en NetBeans](#)

Dentro del documento que se va a generar lo más recomendable es elegir que se incluyan todas las opciones que aparecen marcadas de forma predeterminada y lo mejor es dejarlo como está. Además puede ser interesante marcar también @author y @version para que se incluya el nombre del autor o autores que han desarrollado el proyecto y la versión de éste en la documentación generada.

## Ejecutar.

Esta opción nos permite definir una nueva configuración de ejecución de código, elegir la clase principal, las carpetas de trabajo del proyecto y opciones de la máquina virtual.



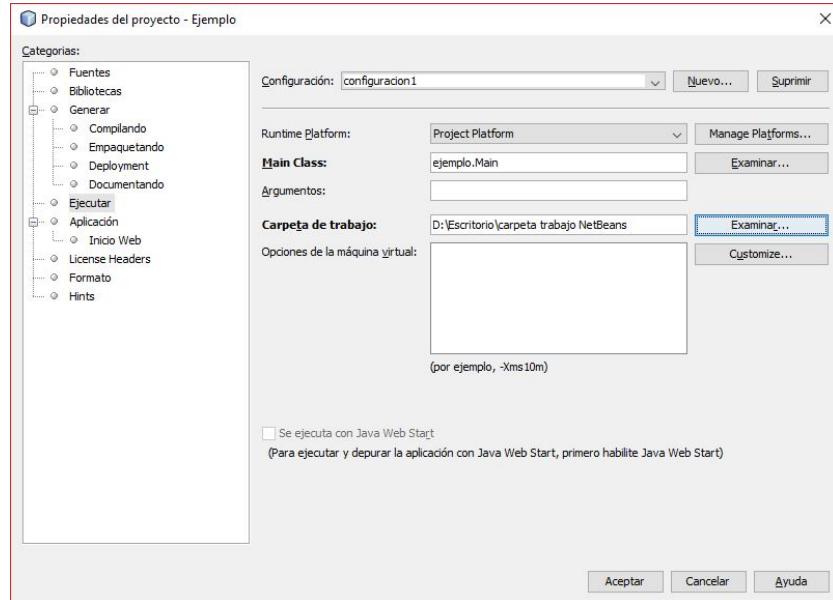
[© Captura de pantalla de Netbeans. Edición de programas en NetBeans](#)

En la ventana de “Configurar el nombre” escribimos el nombre que tendrá nuestra configuración personalizada.

En este caso, escribimos “configuración 1” y pulsamos “aceptar”

A partir de este momento, todas las opciones de configuración que seleccionemos que guardarán en “configuración 1”

También podemos elegir la carpeta de trabajo, para que por ej. los documentos de texto que genere la aplicación se creen en la ruta indicada en lugar de hacerlo en la ruta por defecto (que sería en la que se encuentra el proyecto).



© Captura de pantalla de Netbeans. Edición de programas en NetBeans

## Aplicación.

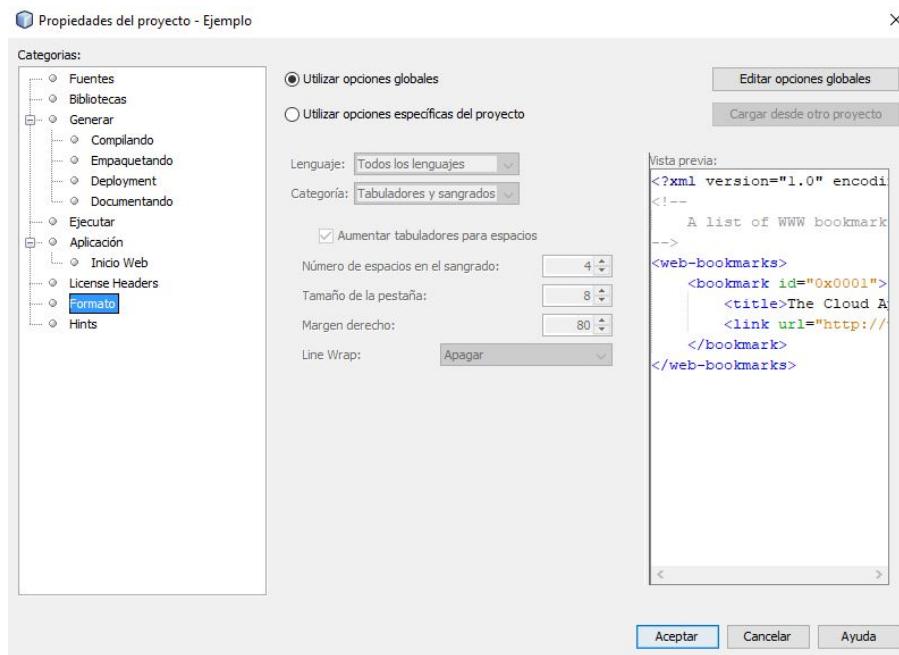
Podemos dar una descripción al proyecto, cambiarle el nombre, etc...

Es conveniente hacerlo, ya que el nombre de los nuevos proyectos se generan automáticamente por NetBeans al inicio de la sesión.

## Formato

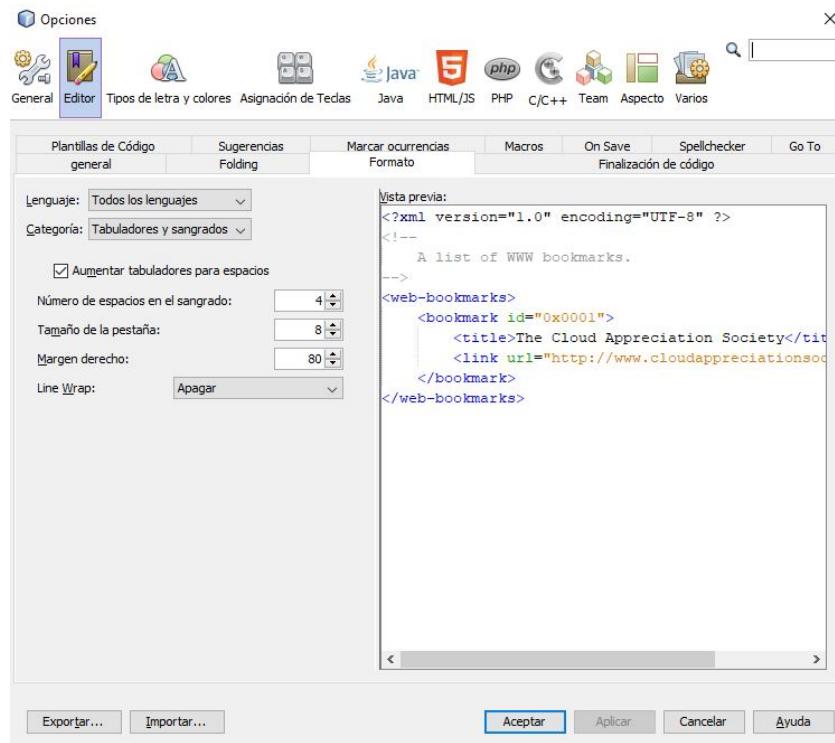
Aquí podemos personalizar aspectos globales del formato del código fuente en la aplicación.

Podemos personalizar las opciones sólo para el proyecto actual o bien para todos los proyectos que estén basados en NetBeans a partir de ahora (utilizar opciones globales)



© Captura de pantalla de Netbeans. Edición de programas en NetBeans

Si pulsamos en el botón Editar opciones globales podremos modificar los valores que tiene por defecto el editor en distintas secciones (General, Editor, Tipos de letras y colores, Asignación de letras, y otros aspectos en función de los complementos que se tengan instalados).

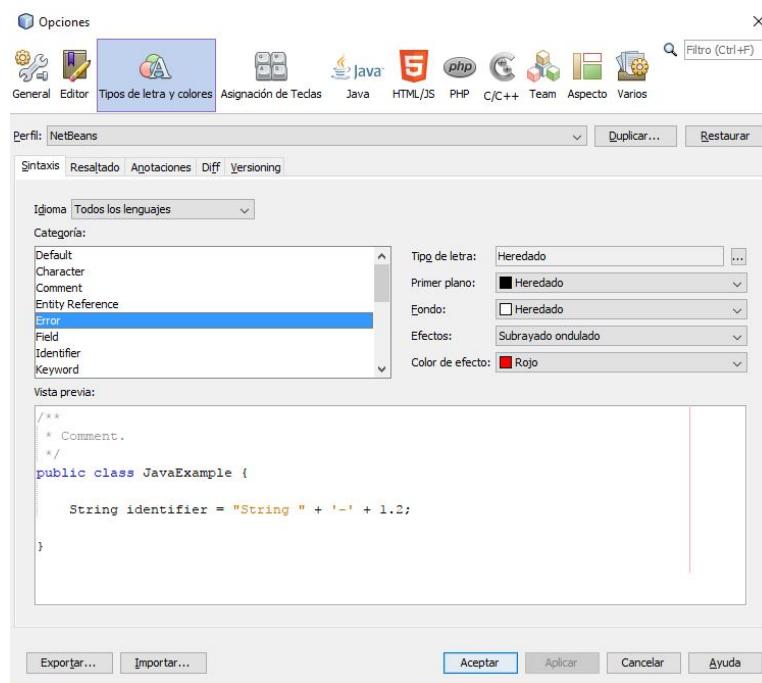


© Captura de pantalla de Netbeans, Edición de programas en NetBeans

En la pestaña Formato se pueden configurar los tamaños de los espaciados, pestañas, margen derecho...

En cuanto al ícono de Tipos de letra y colores, se puede elegir el tipo de letra y colores que prefiramos para el texto del código, así como efectos (si es que así lo deseamos).

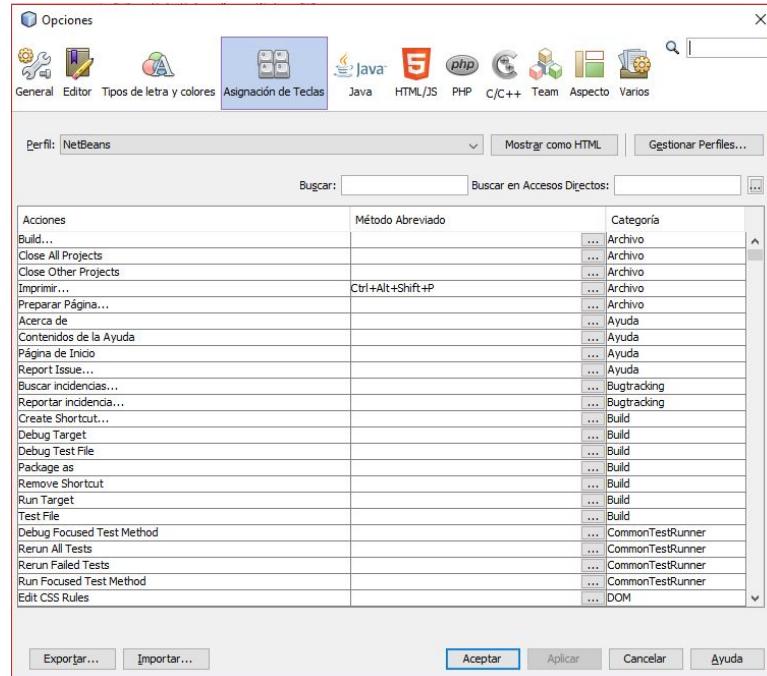
También podemos configurar el tipo de letra y color de los errores del código (por defecto, de color rojo) o cualquier otro aspecto que podemos seleccionar en el cuadro desplegable de Categoría.



© Captura de pantalla de Netbeans, Edición de programas en NetBeans

Y lo mismo con los números, espacios en blanco, etc...

En cuanto a los métodos abreviados de teclado (combinación de teclas equivalente a las acciones en NetBeans), podemos modificar aquellas acciones que hagamos con más frecuencia por aquella combinación de teclas que nos sea más fácil recordar.



© Captura de pantalla de Netbeans. Edición de programas en NetBeans

## **2.1.2.1- Actualización y mantenimiento de NetBeans.**

---

El mantenimiento del entorno de desarrollo es una tarea fundamental que requiere tener todos sus componentes periódicamente actualizados.

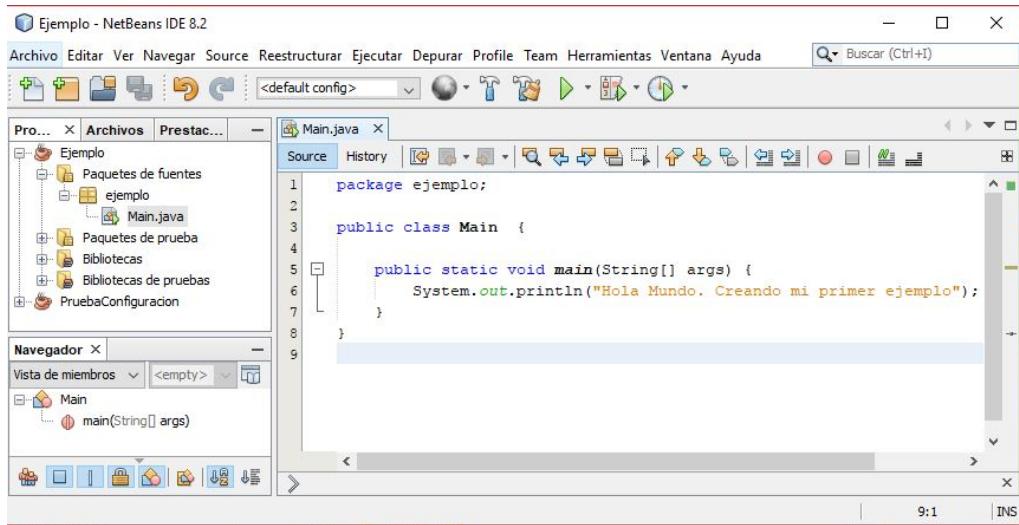
También es de vital importancia realizar copias de seguridad sobre las bases de datos de nuestros proyectos por si ocurriera algún error o proceso defectuoso poder restaurarlos.

El mantenimiento y las actualizaciones se hacen de forma on-line. En NetBeans contamos con el complemento llamado Auto Update Services, que ya viene incorporado y nos avisa cuando encuentra alguna actualización. Nos permite realizar continuas revisiones del entorno y actualizaciones de todos los plugins.

En Eclipse tenemos la opción "Check for Updates" disponible en el menú "Ayuda" que permite la actualización automática.

## 2.1.3.- Uso básico de Netbeans.

En el sitio principal del entorno de desarrollo de NetBeans nos encontramos con la siguiente ventana, que aparece cuando seleccionamos archivo, nuevo proyecto, java:



[© Captura de pantalla de Netbeans.. Netbeans.](#)

Vemos que el espacio se divide en tres secciones principales:

✓ **Ventana Izquierda: ventana de proyectos.**

Aquí irá apareciendo la relación de proyectos, archivos, módulos o clases que vayamos abriendo durante la sesión.

Cada proyecto comprende una serie de archivos y bibliotecas que lo componen.

El principal archivo del proyecto Java es el llamado **Main.java**.

✓ **Ventana derecha: espacio de escritura de los códigos de los proyectos.**

Aquí aparece el esqueleto propio de un programa escrito en lenguaje Java.

Se ha añadido el código:

```
System.out.println("Hola Mundo");
```

Y veremos su significado en las siguientes páginas. De momento, saber que para escribir cualquier código, hay que hacerlo en esta ventana.

✓ **Barra de herramientas:** Permite el acceso a todas las opciones del IDE.

## 2.1.3.1.- Ejemplo de edición de código con Netbeans.

En este documento vamos a introducirnos en la edición de programas en NetBeans a través de un ejemplo sencillo de una aplicación de Java.

Lo primero es iniciar la plataforma. Seleccionamos Archivo - Proyecto Nuevo.

Elegimos una aplicación de Java:



Lo vamos a llamar Ejemplo.

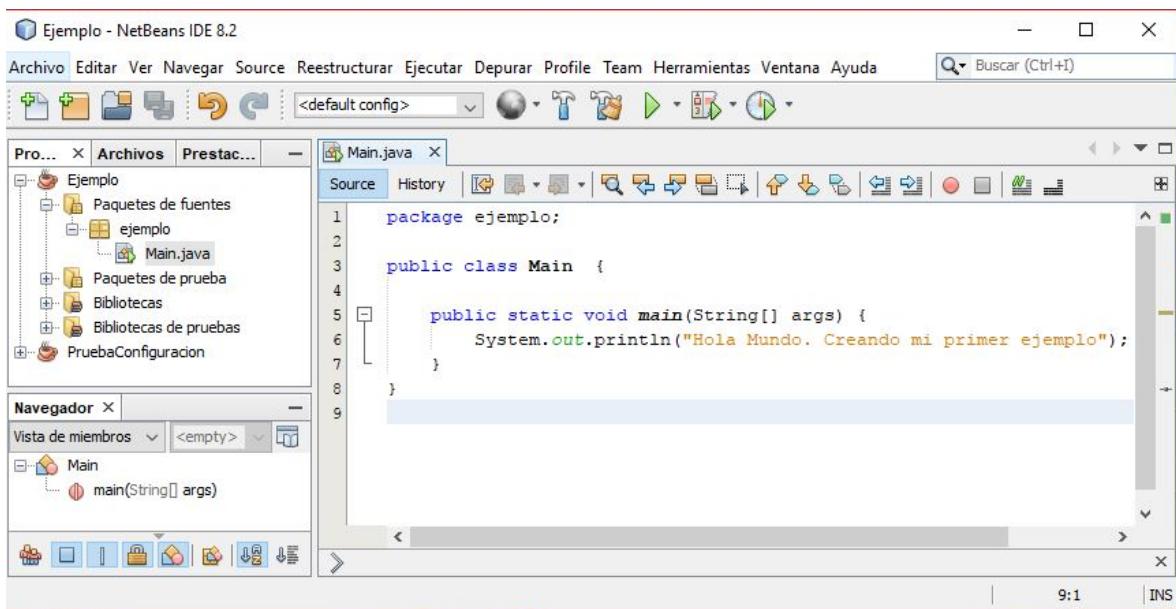
Una vez iniciado el proyecto, en la ventana de proyectos (izquierda) vemos cómo se ha cargado el proyecto ejemplo. Lo seleccionamos con el ratón y se despliega, mostrando todos sus archivos componentes. Seleccionamos Main.java (que es el archivo principal del proyecto, el cual vamos a editar).

En la ventana de edición (a la derecha) nos aparece el esqueleto de la estructura básica de una aplicación en Java.

Lo que vamos a hacer a lo largo del ejemplo es añadir código.

La primera línea de código que vamos a agregar es una orden sencilla en Java, cuya ejecución posterior dará lugar a la aparición de un mensaje por pantalla.

La apariencia del IDE será la siguiente:



Añadimos otra línea más con otro mensaje "Creando mi primer ejemplo"

```

package ejemplo;
/*
 * @author usuario
 */
public class Main {
    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        System.out.println("Hola Mundo");
        System.out.println("Creando mi primer ejemplo");
    }
}

```

Salida - Ejemplo (run)

- run: Hola Mundo
- Creando mi primer ejemplo

GENERACIÓN CORRECTA (total time: 0 seconds)

© Captura de pantalla de Netbeans. Edición de programas en NetBeans

Ahora vamos a modificar la parte de arriba del programa. Añadimos la siguiente línea:

```

package ejemplo;

public class Main {

    public static void main(String[] args) {
        System.out.println("Hola Mundo. Creando mi primer ejemplo");
        JOptionPane.showMessageDialog(null, "Ahora mostramos un mensaje emergente.");
    }
}

```

© Captura de pantalla de Netbeans. Edición de programas en NetBeans

Esta línea nos va a servir para adentrarnos en una de las utilidades más importantes de NetBeans.

NetBeans entiende esta orden como un error (aparece subrayada en una línea roja ondulada y con un pequeño ícono al lado izquierdo)

Si pulsamos sobre ese ícono con el ratón, NetBeans nos aporta sugerencias para deshacer el error:

© Captura de pantalla de Netbeans. Edición de programas en NetBeans

En este caso, elegimos importar javax.swing.JOptionPane.

Según el tipo de error, el IDE no nos propone sugerencias para solucionarlo, pero si situamos el ratón encima del ícono que muestra el error nos da información sobre qué ocurre (lo que nos ayuda a poder solucionarlo).

© Captura de pantalla de Netbeans. Edición de programas en NetBeans

En este caso, aunque no nos ofrece las sugerencias para solucionarlo nos está indicando que falta un ; (punto y coma).

Llegados a este punto, ya hemos comprobado que el editor no nos da ningún problema más. En el siguiente punto del tema, veremos cómo ejecutar esto.

El código completo del ejemplo es el siguiente:

```
package ejemplo;

import javax.swing.JOptionPane;

public class Main {

    public static void main(String[] args){
        System.out.println("Hola Mundo. Creando mi primer ejemplo");
        JOptionPane.showMessageDialog(null,"Ahora mostramos un mensaje emergente.");
    }
}
```

## 2.1.3.2.- Edición de programas en NetBeans.

En este sencillo ejemplo se ve una modificación de las líneas de código en la ventana de codificación del archivo **Main.java** del proyecto **ejemplo** que acabamos de crear.

Las dos líneas que aparecen resaltadas se han escrito sobre la ventana y, tal y como significan en lenguaje Java, su ejecución implicará que sendos mensajes encerrados entre comillas y entre paréntesis saldrán impresos.

No hay que decir que la programación en Java no es objeto del presente módulo, pero puedes probar con algunos ejemplos en Java que tengas de otros módulos.

Mientras escribimos en el editor de textos nos percatamos de varias características de NetBeans que ya hemos señalado en páginas anteriores:

- ✓ Autocompletado de código.
- ✓ Coloración de comandos.
- ✓ Subrayado en rojo cuando hay algún error y posibilidad de depuración y corrección de forma visual, mediante un pequeño ícono que aparece a la izquierda de la línea defectuosa.

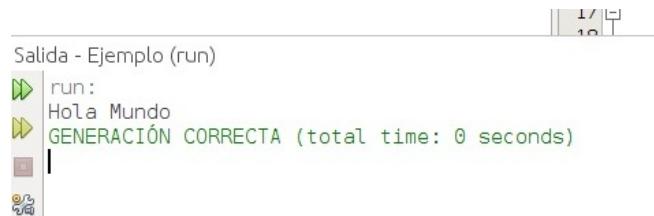
```
Página de Inicio x Main.java x
1
2
3
4
5
6
7
8 package ejemplo;
9
10 /**
11  * @author usuario
12 */
13 public class Main {
14
15 /**
16  * @param args the command line arguments
17 */
18 public static void main(String[] args) {
19     System.out.println("Hola Mundo");
20     System.out.println("Creando mi primer ejemplo");
21 }
```

© Netbeans Captura de Pantalla de Netbeans

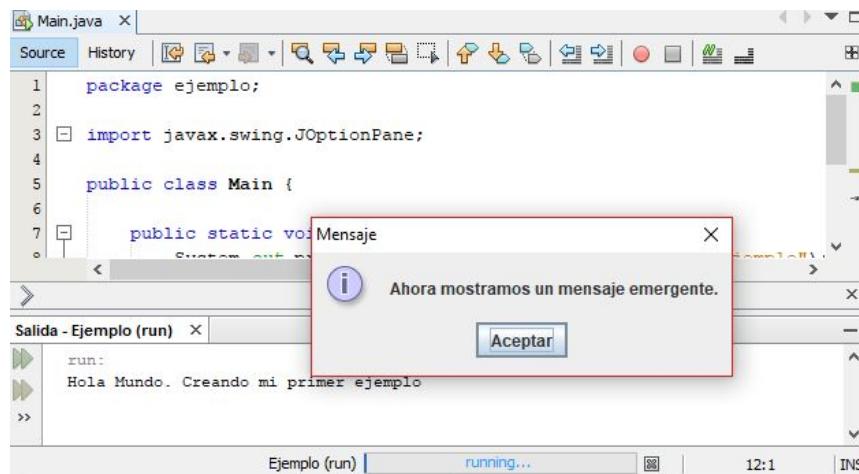
## 2.1.3.3.- Generación de ejecutables en NetBeans.

Una vez tenemos el código plasmado en la ventana de comandos y libre de errores de sintaxis, los siguientes pasos son: compilación, depuración, ejecución.

Al ejecutar el ejemplo anterior, el resultado es:



Si a este ejemplo le añadimos la funcionalidad de JOptionPane, el resultado de la ejecución es:



Continuando con el ejemplo anterior, recuerda que habíamos llegado a este punto, tenemos el programa escrito en el editor libre de errores sintácticos.

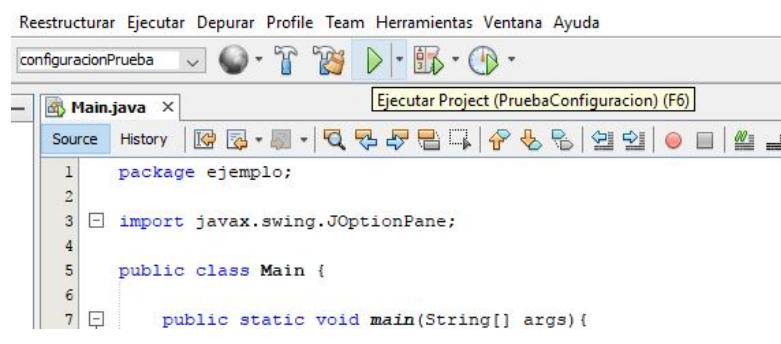
¿Cómo convertir ese programa en ejecutable?

Cabe destacar que, por la sencillez y pequeñez del programa, la ejecución del mismo podría ser directa sin ningún problema.

Sin embargo, debemos acostumbrarnos a seguir los pasos adecuados, que son:

- ✓ Editor libre de errores → Compilación → Depuración → Ejecución
- ✓ Debido a la característica Compilar al guardar del IDE, no hay que compilar el proyecto manualmente para ejecutarlo en el IDE. Cuando guardamos un archivo fuente Java, el IDE lo compila automáticamente. No obstante la opción se encuentra en la barra de menús: Ejecutar → Compile File
- ✓ Depurar → Barra de menús: Depurar → Debug File (o Debug project).
- ✓ Ejecutar → En la barra de menús: Ejecutar → Ejecutar proyecto.

También se puede acceder a las distintas opciones desde el menú contextual del proyecto (en la ventana de la izquierda) o desde la barra de herramientas.



## 2.1.4.- Gestión de módulos en NetBeans.

Con la plataforma dada por un entorno de desarrollo como NetBeans podemos hacer uso de módulos y plugins para desarrollar aplicaciones.

En la página oficial de NetBeans ([www.netbeans.org](http://www.netbeans.org) pinchando en la pestaña Plugins) encontramos una relación de módulos y plugins, divididos en categorías.

Seleccionando la categoría Lenguajes de Programación (Programming Languages), encontraremos aquellos módulos y plugins que nos permitan añadir nuevos lenguajes soportados por nuestro IDE.

Plugins catalogue								
Live Filters - select or type to filter the catalogue, filters are applied all together. There is also multi column sorting activated by 'shift' clicking column headers.								
Name:	Category:	NetBeans version:	Description:	Owner:	Last updated	Avg. Rating	Downloaded	On UC
TypeScript Support for NetBeans IDE	Web Applications, Programming Languages, Editing	8.1, 8.0, 7.4, 7.3	This Project aims to provide support for the TypeScript Language used by TYPO3.	dfranek	2017-09-17	★★★★★ [4.55/5]	11853	7.3, 7.4, 8.0, 8.1
PHP WordPress Blog/CMS	Web Applications, Programming Languages	8.2, 8.1, 8.0, 7.4, 7.3	Support for WordPress.This plugin provides the following features. New Wordpress Project wizard. Create theme(Underscores) action. Hyperlink navigation and code completion for filter and action function. Shortcut nodes for theme and plugin directory.	junichi11	2016-10-04	★★★★★ [4.31/5]	35169	7.4, 8.0, 8.1, 8.2
GWT4NB (for NB8.1+)	Web Applications, Programming Languages	8.2, 8.1	Google Web Toolkit (GWT) Support	markiewb	2017-01-20	★★★★★ [4.49/5]	1121	8.1, 8.2

Un **módulo** es un componente software que contiene clases de Java que pueden interactuar con las **API** del entorno de desarrollo y el manifest file, que es un archivo especial que lo identifica como módulo.

Los módulos se pueden construir y desarrollar de forma independiente. Esto posibilita su reutilización y que las aplicaciones puedan ser construidas a través de la inserción de módulos con finalidades concretas. Por esta misma razón, una aplicación puede ser extendida mediante la adición de módulos nuevos que aumenten su funcionalidad.

Existen en la actualidad multitud de módulos y plugins disponibles para todas las versiones de los entornos de desarrollo más utilizados. En las secciones siguientes veremos dónde encontrar plugins y módulos para NetBeans que sean de algún interés para nosotros y las distintas formas de instalarlos en nuestro entorno.

También aprenderemos a desinstalar o desactivar módulos y plugins cuando preveamos que no los vamos a utilizar más y cómo podemos estar totalmente actualizados sin salir del espacio de nuestro entorno.

Veremos las categorías de plugins disponibles, su funcionalidad, sus actualizaciones...

## 2.1.4.1.- Añadir módulos en Netbeans.

Añadir un módulo va a dotar de mayor funcionalidad a nuestros proyectos desarrollados en NetBeans.

Para añadir un nuevo módulo tenemos varias opciones:

1. Añadir algún módulo de los que NetBeans instala por defecto.
2. Descargar un módulo desde algún sitio web permitido y añadirlo.
3. Instalarlo on-line en el entorno.

Por supuesto, una cuarta posibilidad es crear el módulo nosotros mismos (aunque eso no lo veremos aquí).

Sin embargo, lo más usual es añadir los módulos o plugins que realmente nos interesan desde la web oficial de NetBeans. El plugin se descarga en formato **.nbp** que es el propio de los módulos en NetBeans. Posteriormente, desde nuestro IDE, cargaremos e instalaremos esos plugins. A esta manera de añadir módulos se le conoce como **adición off-line**.

También es habitual instalarlos on-line, sin salir del IDE.

La **adición on-line** consiste en instalar complementos desde nuestro mismo IDE, sin tener que descargarlos previamente.

A modo de ejemplo, en el siguiente enlace tienes un documento donde se explican los pasos para añadir un módulo o plugin, de forma off-line (descargando el archivo e instalándolo posteriormente) y de forma on-line.

Hay dos formas de añadir módulos y plugins en NetBeans:

### 1. Off-line: Buscar y descargar plugins desde la página web oficial de la plataforma:

[Descarga de plugins para NetBeans.](#)

#### Ejemplo:

Vamos a buscar un plugin para jugar al sudoku desde nuestro IDE. No es muy educativo, pero sirva como ejemplo la manera en que se va a realizar el proceso (será igual en todos los casos):

Entramos en la zona de descargas de plugins para NetBeans y en la zona del catálogo, escribiremos la palabra sudoku en el cuadro de texto Nombre:

#### Plugins catalogue

Live Filters - select or type to filter the catalogue, filters are applied all together. There is also multi column sorting activated by 'shift' clicking column headers.								
Name: <input type="text" value="sudoku"/>	Category: <input type="text" value="Any"/>	NetBeans version: <input type="text" value="Any"/>	Description: <input type="text"/>	Owner: <input type="text"/>	Verified for: <input type="text" value="Any"/>	Last updated	Avg. Rating	Downloaded
 nbSudoku	Utilities	6.0	Sudoku Game	rahulgarg	2007-04-01	 ★ [4.0/4.5]	2889	
 Sudoku Solver	Fun Stuff	6.0	Simple Sudoku Solver plugin	wijprakash	2012-03-21	 ★ [0/5]	1876	
Showing 1 to 2 of 2 entries (filtered from 1,070 total entries)								

[Netbeans. Módulos de Netbeans.](#)

[@ Captura de pantalla de](#)

Se nos abre una ventana con las características del plugin y la opción de descargarlo. Elegimos la carpeta donde queramos que se guarde.

Entramos en NetBeans: Creamos nuevo proyecto y seleccionamos el tipo de proyecto que queramos (por ejemplo, aplicación de Java).

Nos vamos a Herramientas -> Complementos: En la pestaña "Descargado" pinchamos el botón "Aregar Plugins"

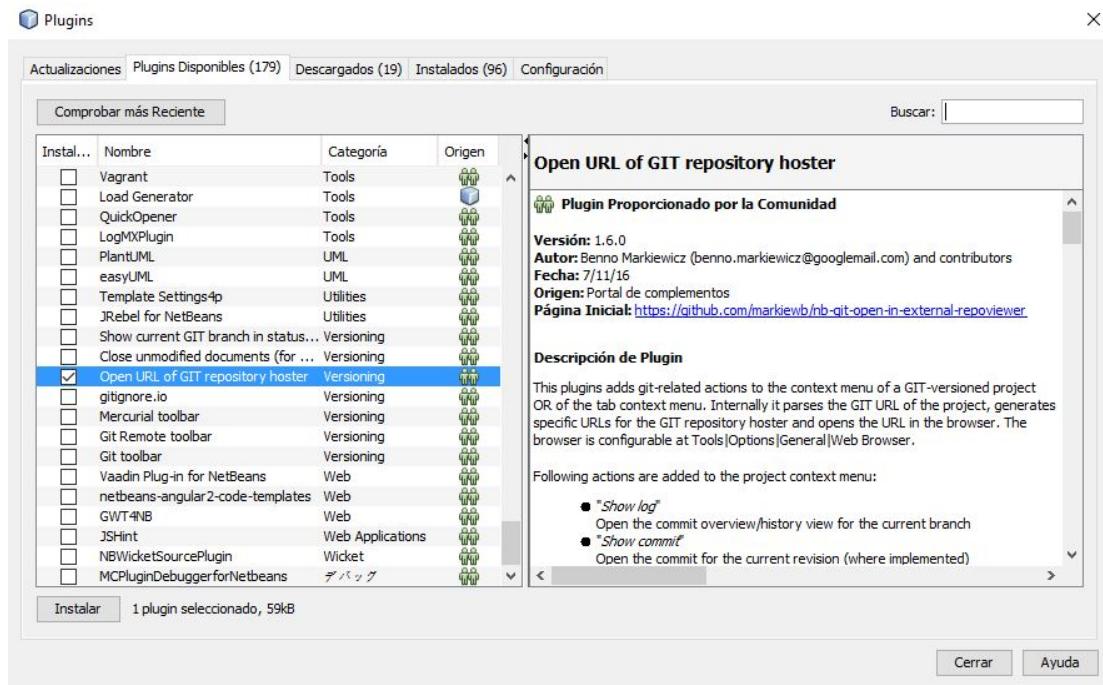
Seleccionamos la carpeta donde habíamos guardado el plugin del sudoku y le damos a "aceptar"

Estando el plugin seleccionado, pulsamos "instalar", empieza la instalación, pulsamos siguiente. Después, aceptamos la licencia para instalar.

### 2. On-Line: Instalarlos desde el propio entorno de desarrollo:

Ahora vamos a instalar otro plugin con mayor utilidad que el anterior:

Estando en NetBeans, seleccionamos Herramientas -> Complementos: En la pestaña de plugins disponibles: seleccionamos Open URL of Git repository hoster -> Instalar



[© Captura de pantalla de Netbeans. Módulos de Netbeans.](#)

Se abre la ventana de instalación donde tenemos que aceptar los términos de licencia y pulsamos sobre *Instalar*.

Puede que nos salte un mensaje de advertencia para verificar el certificado si Netbeans no lo considera de confianza, si confiamos en su procedencia pulsamos en *Continuar* y el proceso de instalación seguirá.

Finalmente pulsamos sobre *Terminar* y ya tenemos el plugin instalado.

## 2.1.4.2.- Eliminar módulos en Netbeans.

Cuando consideramos que algún módulo o plugin de los instalados no nos aporta ninguna utilidad, o bien que el objetivo para el cual se añadió ya ha finalizado, el módulo deja de tener sentido en nuestro entorno. Es entonces cuando nos planteamos eliminarlo.

Eliminar un módulo es una tarea trivial que requiere seguir los siguientes pasos:

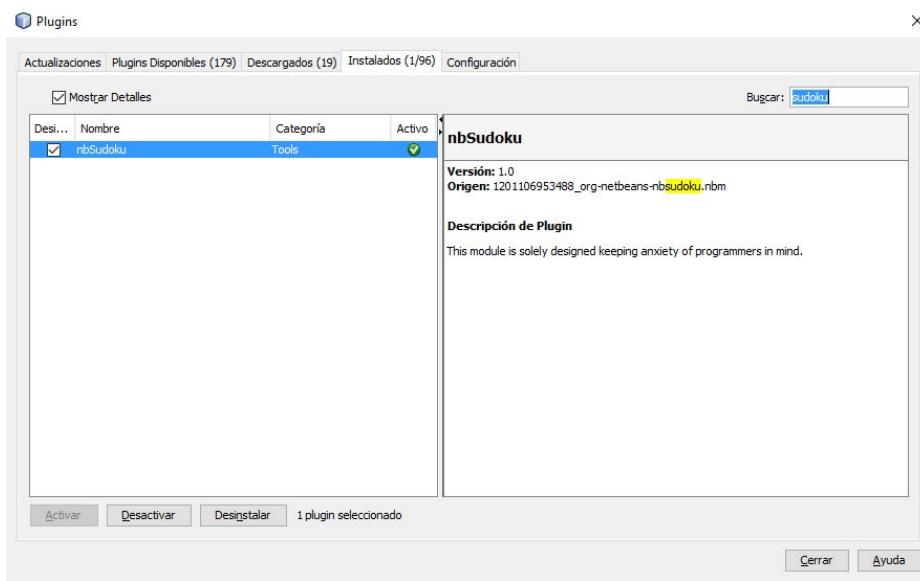
1. Encontrar el módulo o plugin dentro de la lista de complementos instalados en el entorno.
2. A la hora de eliminarlo, tenemos dos opciones:
  - o **Desactivarlo**: El módulo o plugin sigue instalado, pero en estado inactivo (no aparece en el entorno).
  - o **Desinstalarlo**: El módulo o plugin se elimina físicamente del entorno de forma permanente.

En la ventana de Complementos (Plugins) que se abre desde Herramientas->Plugins, nos iremos a la pestaña Instalados, seleccionaremos el complemento y tendremos que elegir entre esas dos opciones: desactivar o desinstalar.

Vamos a ver la secuencia de pasos a seguir para eliminar el plugin del juego del sudoku del entorno.

El proceso es muy sencillo: basta con conseguir la lista de complementos instalados (Herramientas -> Complementos). Localizamos el complemento que queremos eliminar escribiendo su nombre en el lugar destinado para ello y seleccionamos una de entre las dos opciones posibles: desinstalarlo o desactivarlo

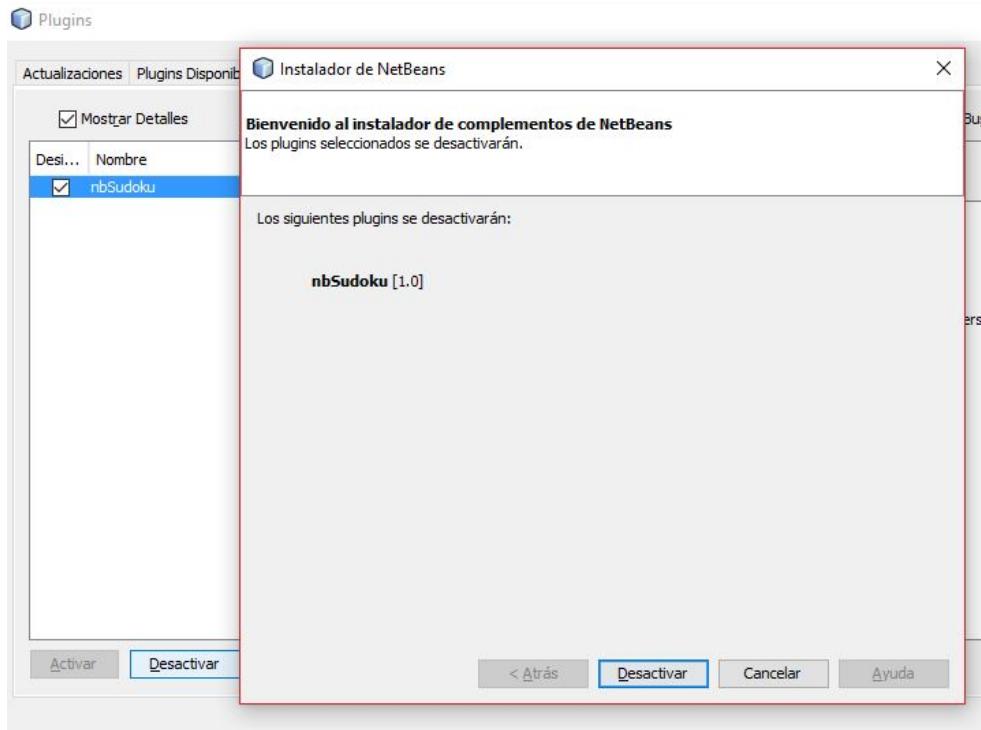
En la pestaña de complementos instalados, escribimos el nombre del plugin (sudoku) en la barra de búsqueda:



[© Captura de pantalla de Netbeans. Módulos de Netbeans.](#)

Cuando lo encuentra, en la ventana aparecen las dos posibilidades de eliminación.

En este caso, hemos optado por desactivarlo.



[© Captura de pantalla de Netbeans. Módulos de Netbeans.](#)

## 2.2.- Eclipse.

---

Eclipse es mucho más que un entorno de desarrollo integrado. Eclipse podría definirse como una plataforma de desarrollo software que dispone de Entornos de desarrollo integrados para algunos de los lenguajes de programación más utilizados, como Java, PHP y C++, así como una increíble comunidad de desarrolladores que respalda al proyecto.

Los distintos entornos de desarrollo que ofrece el proyecto Eclipse son de código abierto, gratuitos y multiplataforma. A pesar de su gratuidad, podemos colaborar en la financiación del proyecto eclipse.

Desde [la página oficial del proyecto](#) podemos descargar los distintos IDEs que proporciona la comunidad, además podemos formar parte de la comunidad Eclipse y participar, o simplemente conocer, los distintos proyectos software de eclipse.

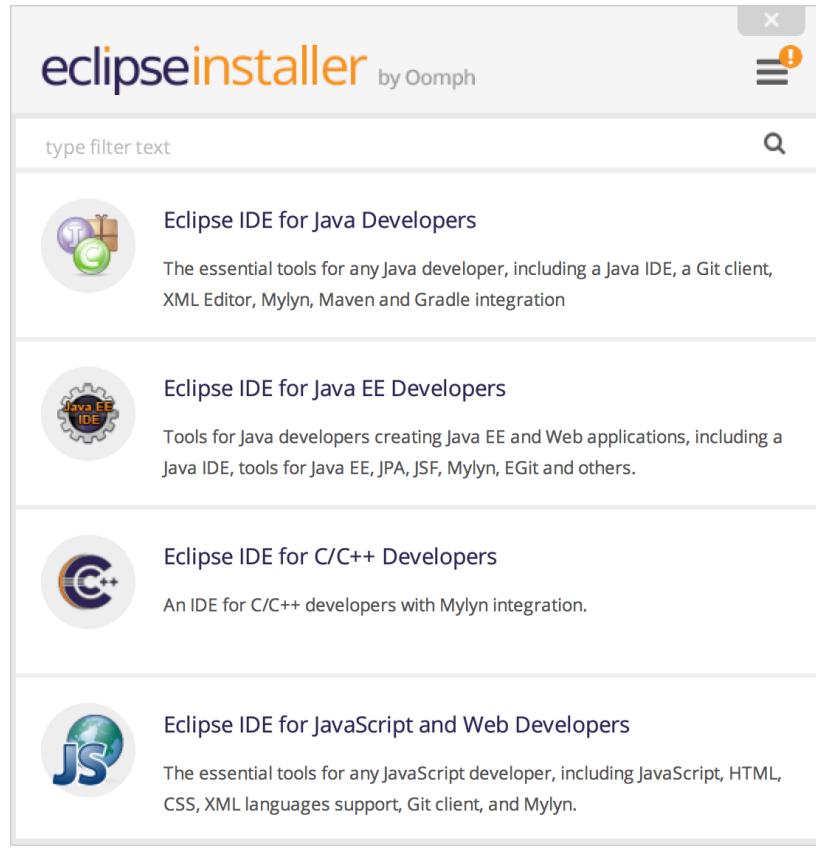
Una vez hemos accedido a la página, si nos vamos a la sección de Descargas (Download) podemos descargarnos las distintos entornos de desarrollo integrados que ofrece el proyecto eclipse.

Dentro de las funciones básicas que ofrece Eclipse podemos encontrar:

- ✓ Creación de nuevos proyectos. Requieren de un workspace para poder desarrollar. En netbeans se estructuran en proyectos que al final son directorios.
- ✓ Admite etiquetado resaltado y autocompletado.
- ✓ Permite la importación/exportación de proyectos.
- ✓ Funciona con proyectos Android, aunque se haya en desuso.
- ✓ Su potencialidad viene dada por los plugins. Además actualmente se instala de forma segmentada por lenguajes lo que lo hace mas ligero.
- ✓ Admite acciones de puntos de ruptura y de refactorización.
- ✓ Contiene un sistema de depuración de proyectos.
- ✓ Aunque Netbeans tambien permite el desarrollo de aplicaciones Web, Eclipse es mas usado para este tipo de proyectos.
- ✓ Posee detección de errores en tiempo real.
- ✓ Dispone de la opción de actualización automática desde el menú "Ayuda".
- ✓ A partir del proyecto Babel, permite la traducción a 47 idiomas diferentes.
- ✓ Admite la generación de código Javadoc.
- ✓ Soporta JUnit.

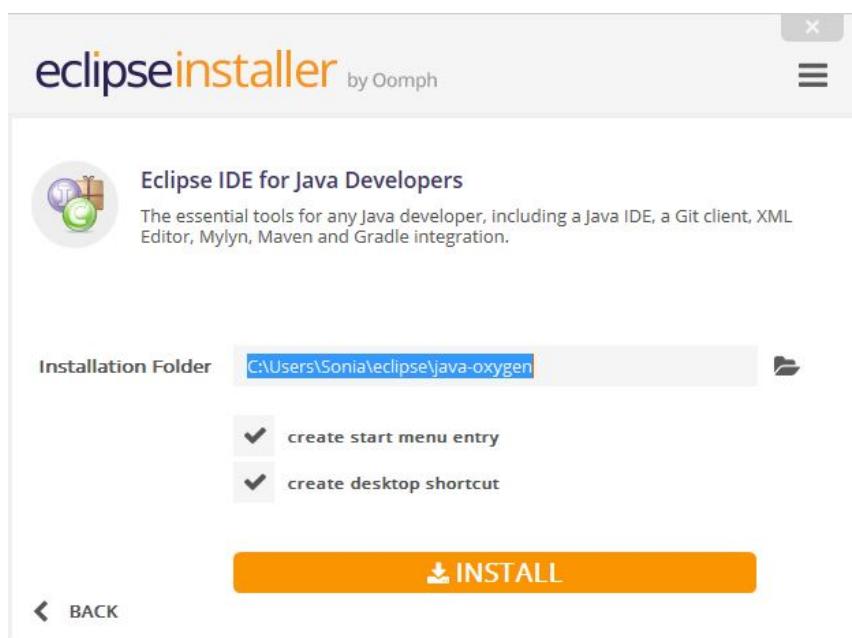
## 2.2.1.-Instalación y utilización de Eclipse.

Una vez hayamos descargado el entorno de desarrollo que mejor se adapte a nuestras necesidades, procedemos a su instalación, que según el sistema operativo que utilicemos será de una forma u otra. Si seleccionamos la opción Eclipse Oxygen (*última versión disponible en enero de 2018*) al comenzar el proceso de instalación nos preguntará el lenguaje de programación con el que vamos a utilizar eclipse. De esta forma podemos instalar el IDE con la configuración que mejor se adapte a nuestras necesidades. Por otra parte podíamos habernos descargado la opción que más nos interesa para nuestro proyecto software.

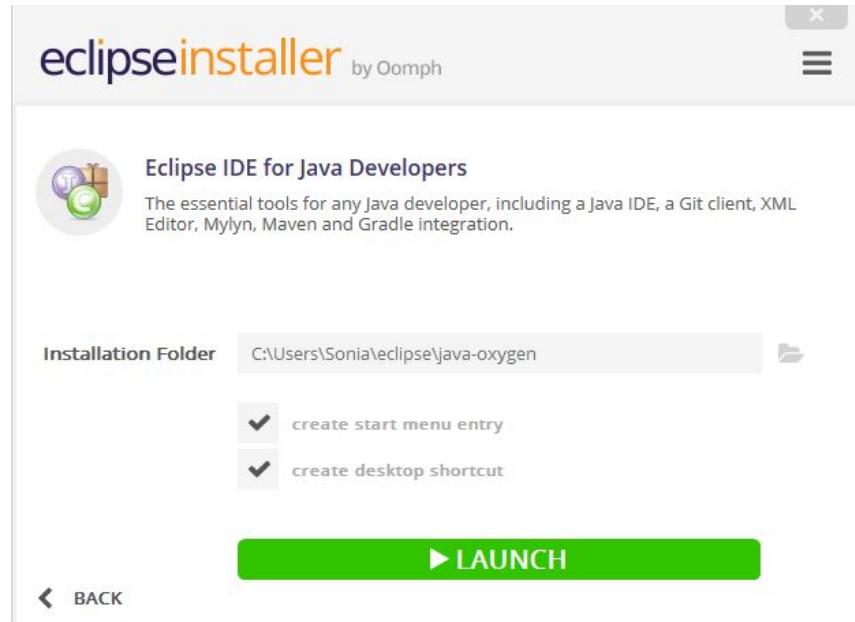


## Instalación en Microsoft Windows

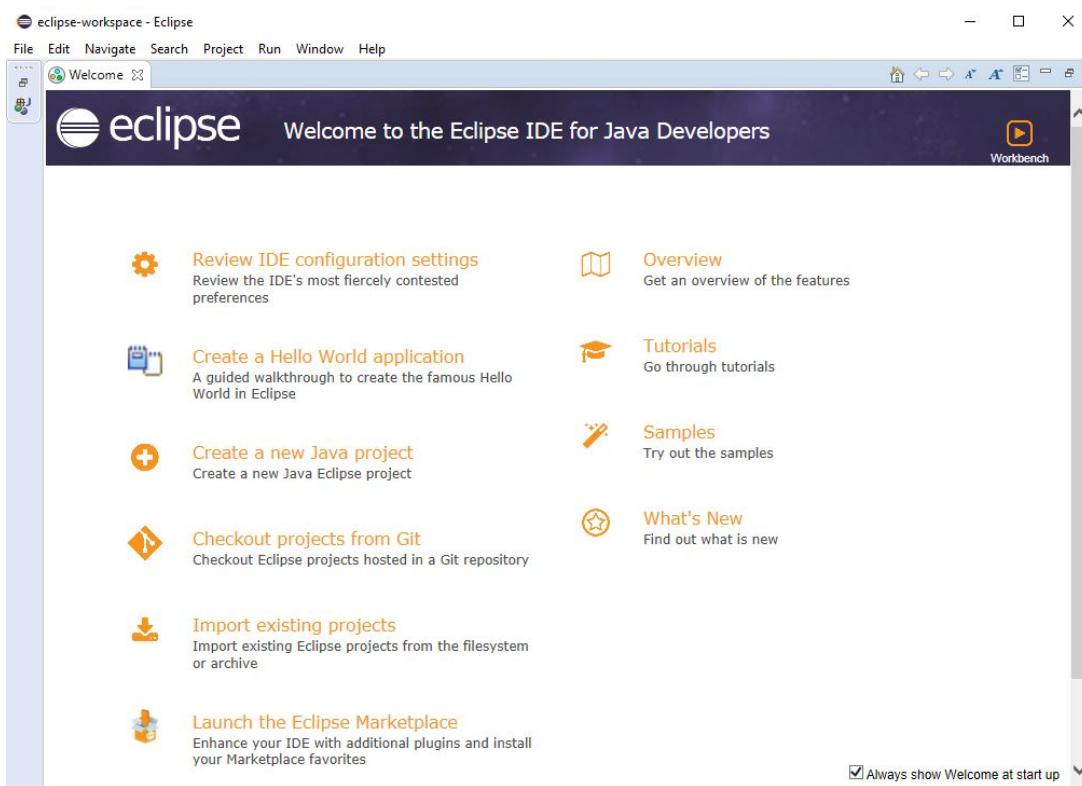
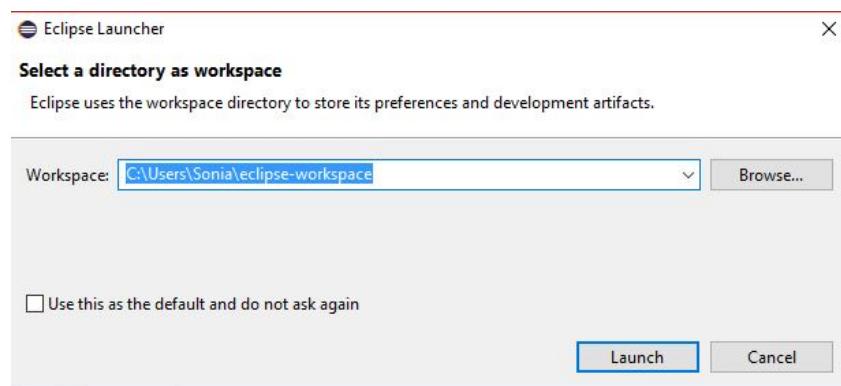
Para llevar a cabo la instalación en Microsoft Windows, nos descargamos el instalador de Eclipse y desde él elegiremos la opción Eclipse IDE for Java Developers.



Indicaremos el directorio en el que queremos que realice la instalación y pulsaremos el botón Install.



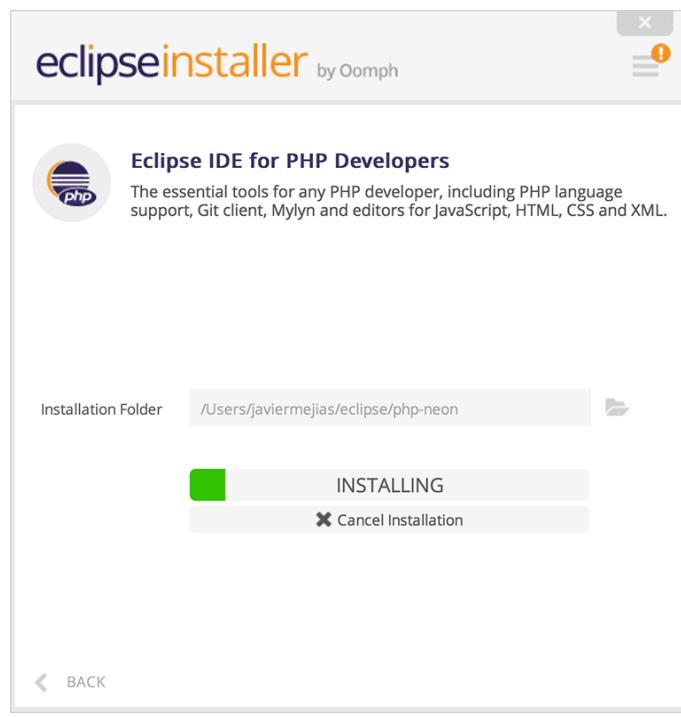
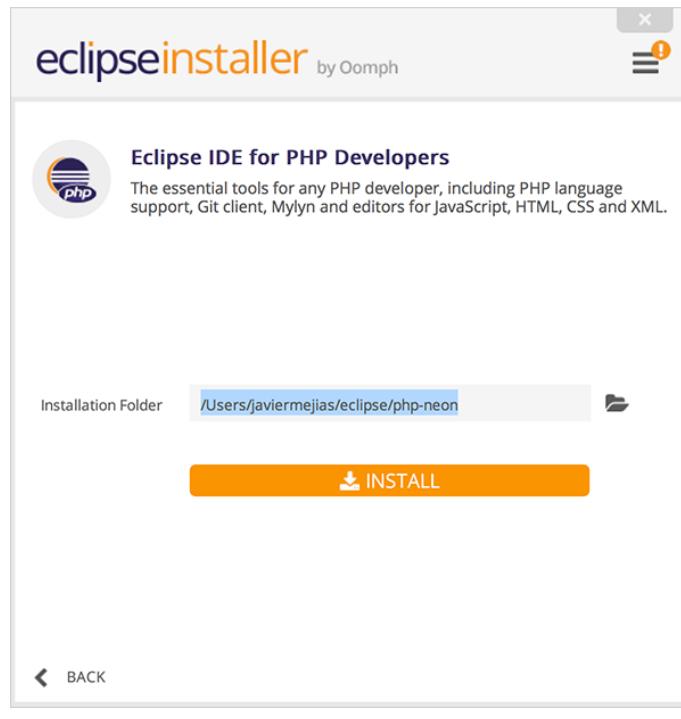
Y por último nos pedirá establecer el directorio para el workspace.



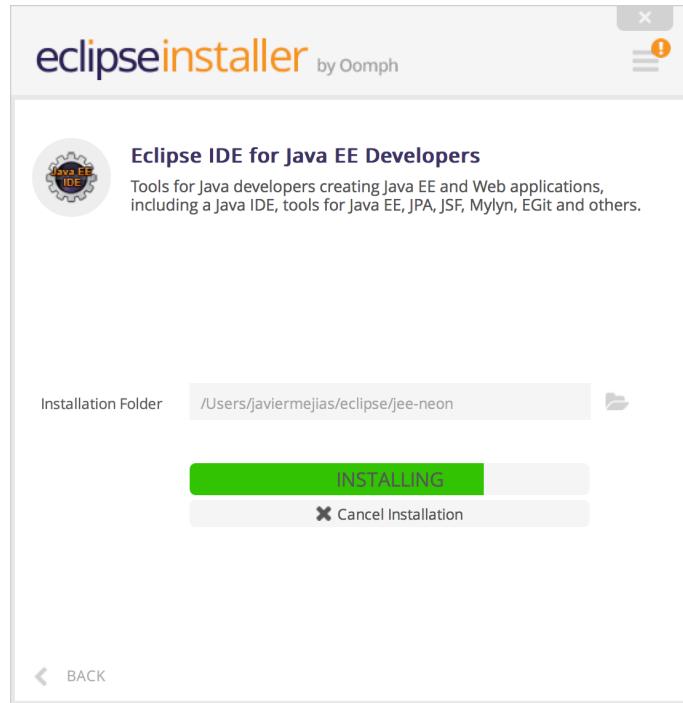
# Instalación en MacOs

Los usuarios del sistema operativo de Apple están acostumbrados a sencillos procesos de instalación. Basta con descargar la versión que queremos del entorno de desarrollo, hacer doble clic en el paquete descargado y comenzará un sencillo proceso de instalación, en la que el usuario tendrá que aceptar la licencia de utilización de dicho IDE, así como seleccionar la carpeta en la que quiere que se guarden sus proyectos por defecto. De una forma sencilla y rápida quedará el IDE instalado en nuestra computadora.

Comenzamos la instalación de la versión Eclipse Neon genérica y elegimos la opción para PHP.



Si la opción elegida hubiera sido otra, por ejemplo Eclipse para Java, el proceso de instalación hubiera sido el mismo, mostrando las distintas pantallas del proceso de instalación adaptadas al IDE elegido.



Javier Mejías Real. Instalación de Eclipse para Java (CC0)

Recordar también que si desde la página de descarga hubiéramos elegido la configuración de eclipse acorde a nuestras necesidades, por ejemplo Eclipse para PHP o para C++, en el proceso de instalación del IDE genérico, no nos hubiera preguntado qué configuración de eclipse queremos.

## Instalación en GNU/Linux

Para instalar eclipse en nuestro sistema GNU/Linux podemos hacerlo de dos formas. La primera sería descargar el paquete desde la página oficial de Eclipse, descomprimirlo y ejecutarlo

La segunda es a través de nuestro terminal. Si estamos utilizando una distribución GNU/Linux basada en Ubuntu o Debian, ejecutaremos el siguiente comando:

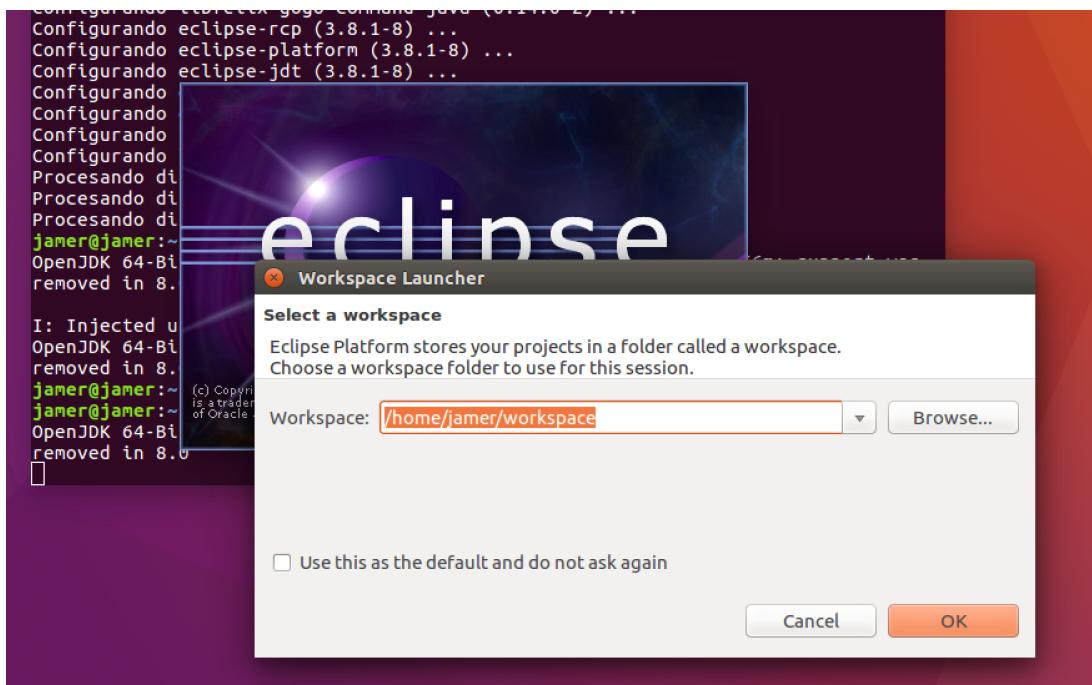
```
sudo apt-get install eclipse
```

```
jamer@jamer:~$ sudo apt-get install eclipse
[sudo] password for jamer:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  ant ant-optimal aspectj binfmt-support eclipse-jdt eclipse-pde
```

Javier Mejías. Terminal de GNU/Linux (CC0)

Nos pedirá nuestra contraseña de root y luego nos preguntará si queremos descargar la distribución de Eclipse. Una vez hayamos afirmado, eclipse se instalará sin ningún tipo de problema en nuestra computadora.

Una vez lo tengamos instalado, podremos ejecutarlo escribiendo en el terminal "eclipse".

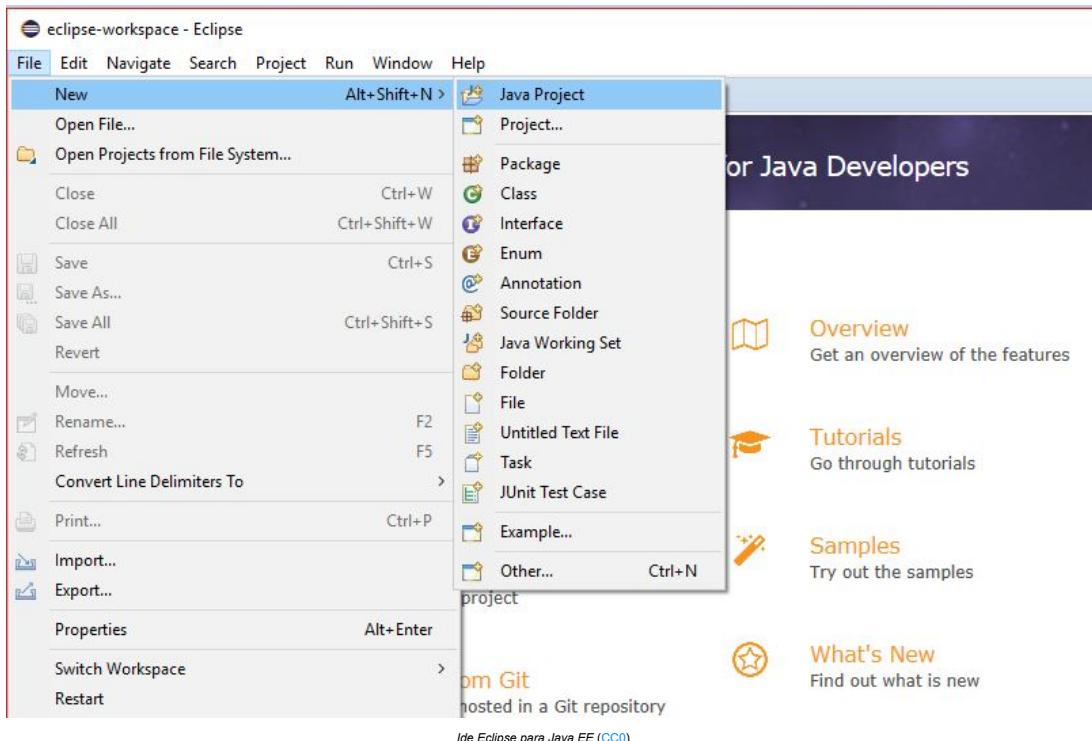


Javier Mejías. Ejecución de Eclipse en GNU/Linux ([CC0](#))

Recuerda que los distintos IDE del proyecto eclipse son proyectos código abierto y gratuitos, los cuales pueden descargarse desde la página oficial del proyecto. La página web del proyecto eclipse es <https://www.eclipse.org/> y en ella encontraremos todos los posibles entornos de desarrollo soportados por la comunidad Eclipse. Recomendamos realizar las descargas siempre desde las páginas oficiales del proyecto, de esta forma evitaremos descargar software de dudosa procedencia, lo cual siempre puede ser un riesgo para la integridad y seguridad de nuestro sistema.

Una vez hemos instalado el entorno de desarrollo ya podemos trabajar con él. Cada vez que lo iniciemos nos preguntará por el directorio donde queremos guardar los proyectos que vayamos desarrollando. Podemos marcar la opción para que por defecto se abra siempre el mismo directorio y no nos pregunte cada vez que iniciemos eclipse.

Una vez hemos arrancado Eclipse, podremos trabajar en los proyectos que tengamos o crear uno nuevo accediendo a través del menú File.



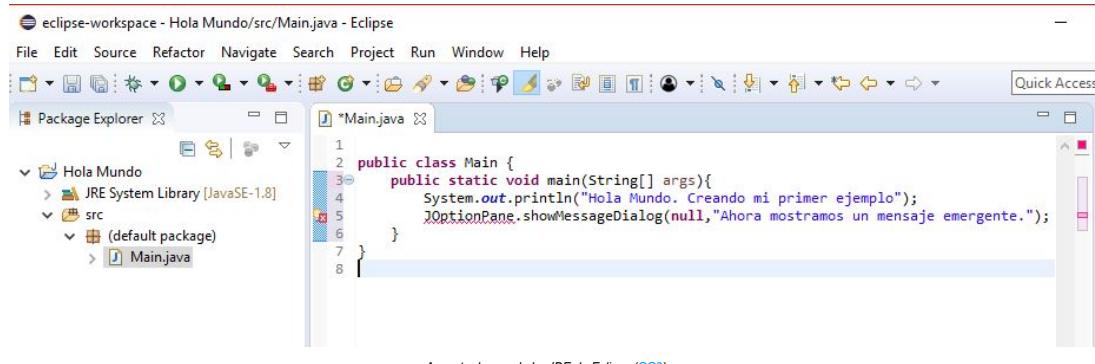
Ide Eclipse para Java EE ([CC0](#))

Creamos un nuevo proyecto, por ejemplo Hola Mundo. Una vez creado añadiremos una nueva clase a la que llamaremos Main y en la que copiaremos el siguiente código:

```
public static void main(String[] args){  
    System.out.println("Hola Mundo. Creando mi primer ejemplo");  
}
```

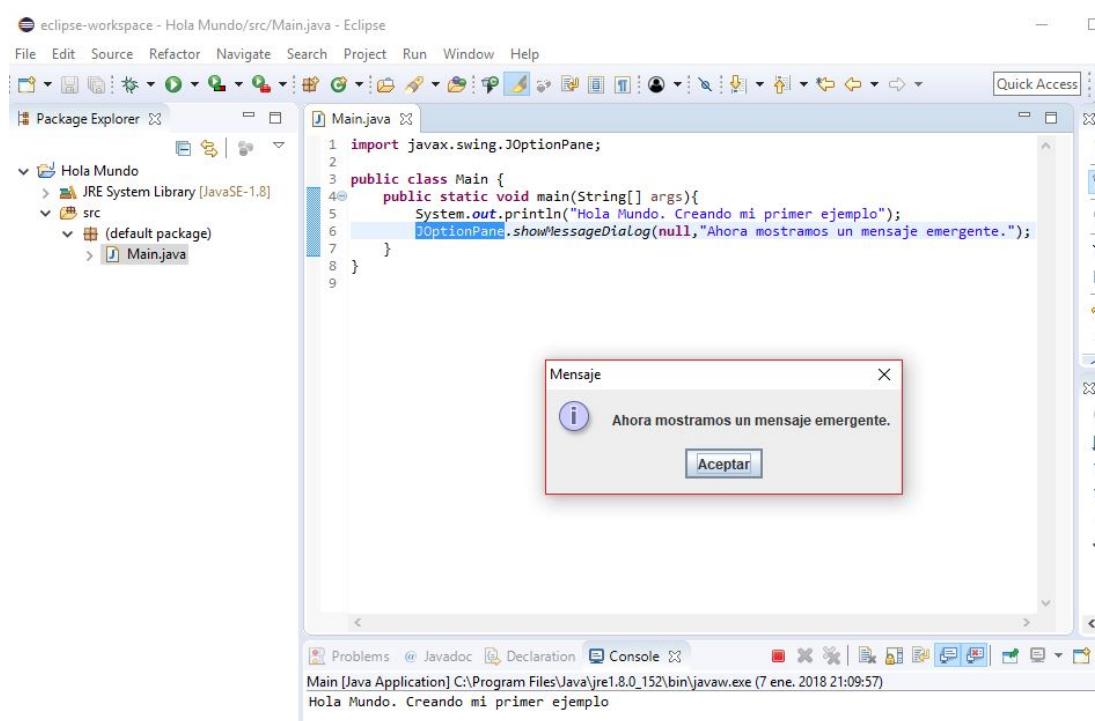
```
        JOptionPane.showMessageDialog(null,"Ahora mostramos un mensaje emergente.");
    }
```

Quedando de la siguiente manera:



Al igual que NetBeans, Eclipse también nos proporciona sugerencias para corregir errores. Así que añadimos la librería que falta del mismo modo que hicimos con NetBeans.

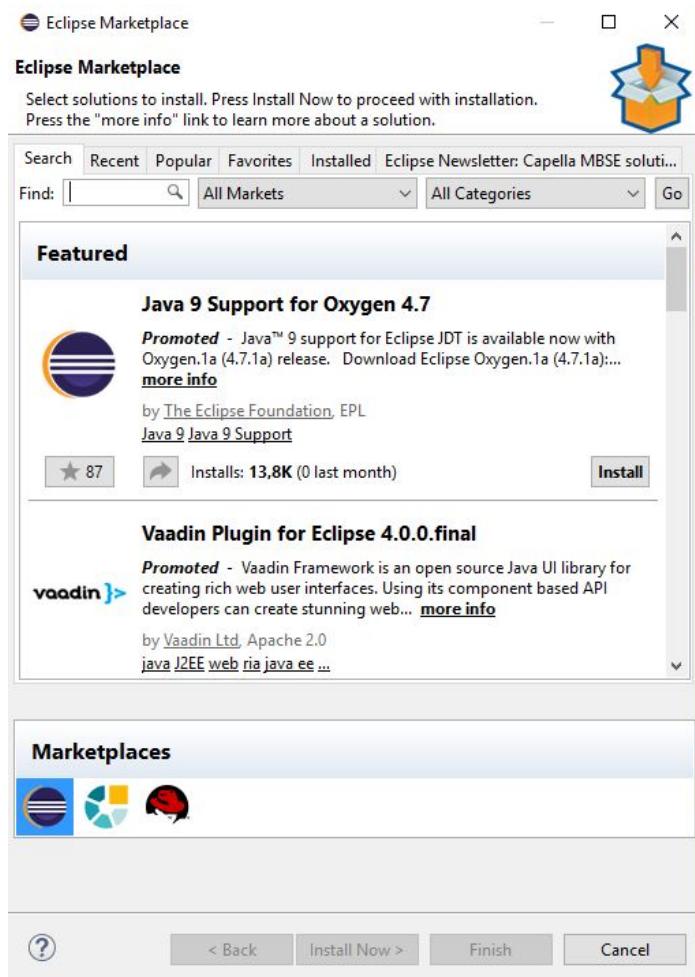
Su ejecución será similar también, podemos darle directamente al botón Ejecutar de la barra de herramientas.



## 2.2.2.- Gestión de módulos en Eclipse. El marketplace.

El marketplace de Eclipse permite instalar módulos o plugins para aumentar las capacidades o ampliar las características del entorno.

Para acceder a él debes acceder al menú de Ayuda o Help donde podrás seleccionar la opción "Eclipse marketplace" que permite además buscar plugins también en otros marketplaces como RedHat y Obeo.



Entre otros plugins son muy recomendables los siguientes.

- ✓ UML Designer. Sirve para el modelado de objetos en PQQ y el árbol XML.
- ✓ Ant Visualization: Proporciona una presentación gráfica de las dependencias entre objetos Ant.
- ✓ Eclipse Visual Editor: Plataforma para la creación de Clases visuales y gráficas sin Eclipse. Soporta WYSIWYG.

### **3.- Herramientas CASE para el desarrollo, prueba y documentación del código.**

---

Las herramientas CASE (Computer Aided Software Engineering) son aplicaciones software que nos facilitan el desarrollo del software, reduciendo el tiempo y el coste del mismo.

En este epígrafe vamos a identificar las funciones más usuales de un conjunto de herramientas CASE para el desarrollo, prueba y documentación del código. Veremos algunas herramientas incluidas en los propios entornos de desarrollo, bien de forma nativa o como plugin.

## 3.1.-Herramientas de los navegadores web.

---

Actualmente, las últimas versiones de los navegadores web más populares: Mozilla Firefox, Google Chrome, Safari y Microsoft Edge, contienen herramientas para los desarrolladores web. A través de dichas herramientas podemos trabajar con nuestros desarrollos web de forma dinámica, permitiéndonos habilitar y deshabilitar distintos lenguajes utilizados, como JavaScript y CSS, modificar la hoja de estilos en tiempo real e incluso modificar líneas de código HTML de forma dinámica, para ver como quedaría nuestra web con dichas modificaciones y con la ventaja de que los cambios serán deshechos al recargar la página.

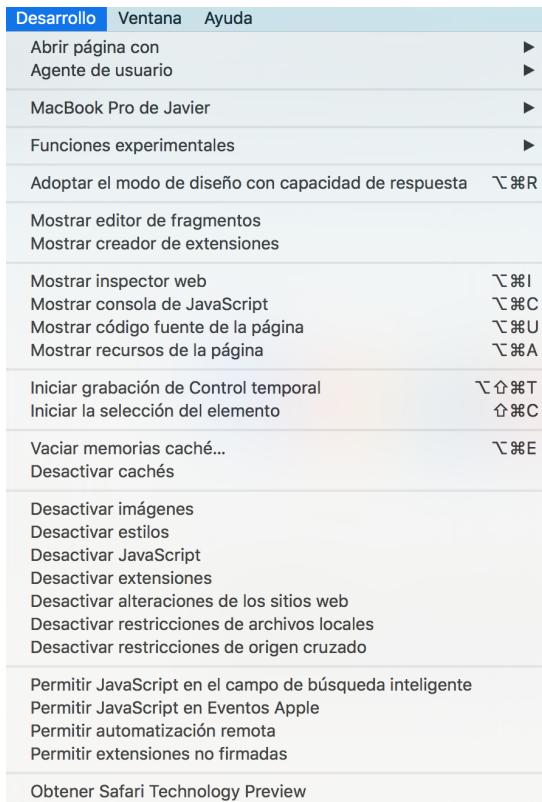
Veremos las herramientas de ayuda a los desarrolladores web que contienen los navegadores web: Safari, Microsoft Edge, Mozilla Firefox y Google Chrome.



Javier Mejías. Logos de distintos navegadores web. ([CC0](#))

### 3.1.1.- Safari.

Safari es el navegador web desarrollado por Apple. Dispone de un amplio conjunto de herramientas para desarrolladores web. Desde la pestaña Desarrolladores, se puede acceder a un amplio abanico de herramientas para el desarrollo web.

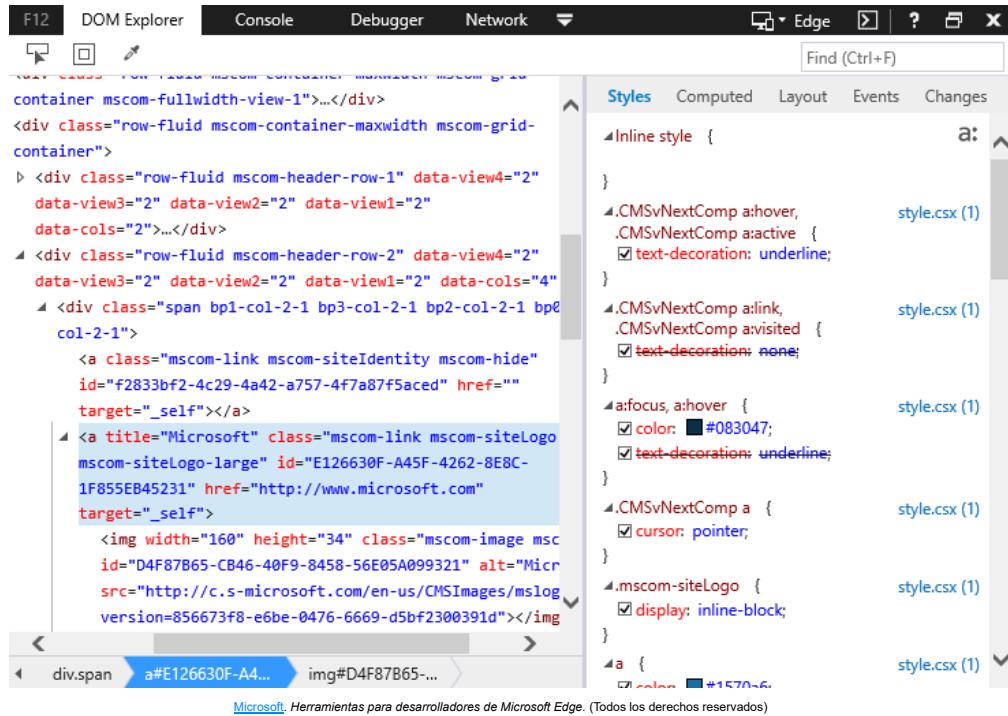


Entre las distintas herramientas nos encontramos el inspector web, que puede ejecutarse haciendo clic en la opción "Mostrar inspector web". Podremos analizar todo el código HTML y CSS de cualquier página web, además podremos realizar modificaciones sobre el mismo en tiempo real, modificaciones que solo estarán visibles en nuestro navegador y de forma temporal, hasta que recarguemos la página.

A screenshot of a Google search results page for 'Google' in Spanish ('Google España'). The browser window includes standard Mac OS X controls (red, yellow, green buttons) and a toolbar with icons for file operations, search, and tabs. The address bar shows 'Buscar o introducir sitio web' and 'Google'. Below the address bar, there are links for 'Gmail', 'Imágenes', and 'Iniciar sesión'. The main content area displays the Google logo and a search bar with the placeholder 'Buscar con Google' and 'Voy a tener suerte'. At the bottom of the page, there's a link 'Google.es también en: català àleoo euskara'. The bottom half of the screen shows the Safari developer tools. The 'Elementos' tab is active, displaying the DOM tree for the current page. A specific element in the tree is highlighted with a blue border. The right side of the developer tools interface shows the 'Inspector' panel with detailed information about the selected element, including its properties and styles. The status bar at the bottom of the developer tools panel indicates 'Javier Mejías. Inspector web en Safari (CC0)'.

### 3.1.2.- Microsoft Edge.

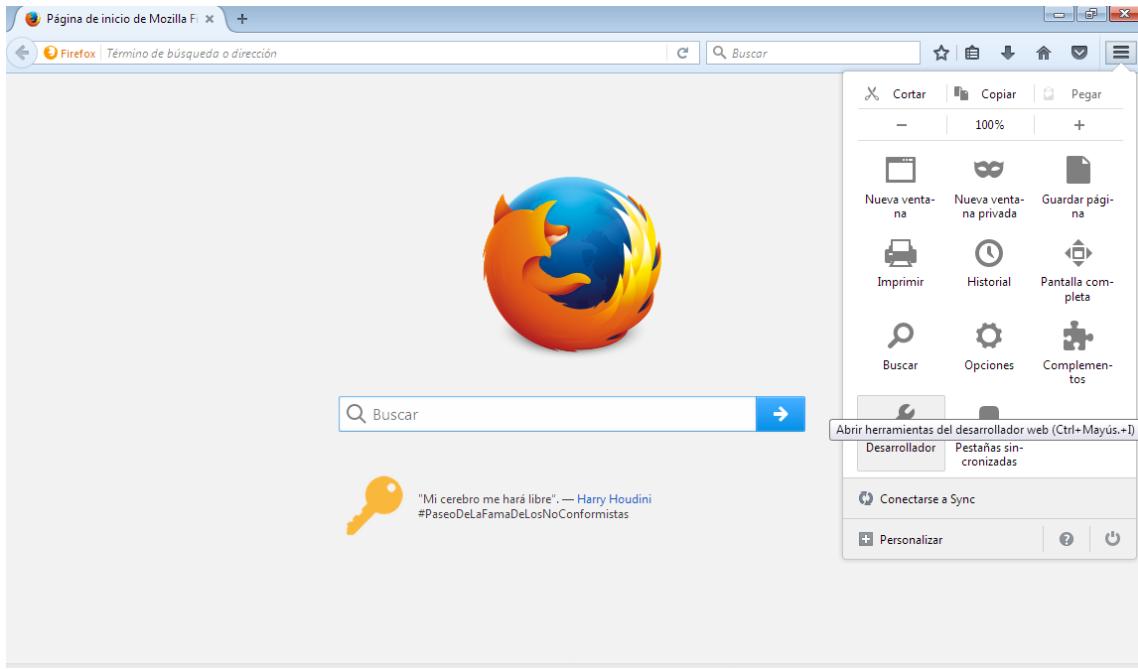
Microsoft Edge es el nuevo navegador web de Microsoft Windows. Disponible a partir de la versión 10 de Windows. Para acceder a las herramientas de desarrollo basta con pulsar la tecla F12 y se abrirá el menú de herramientas para desarrolladores.

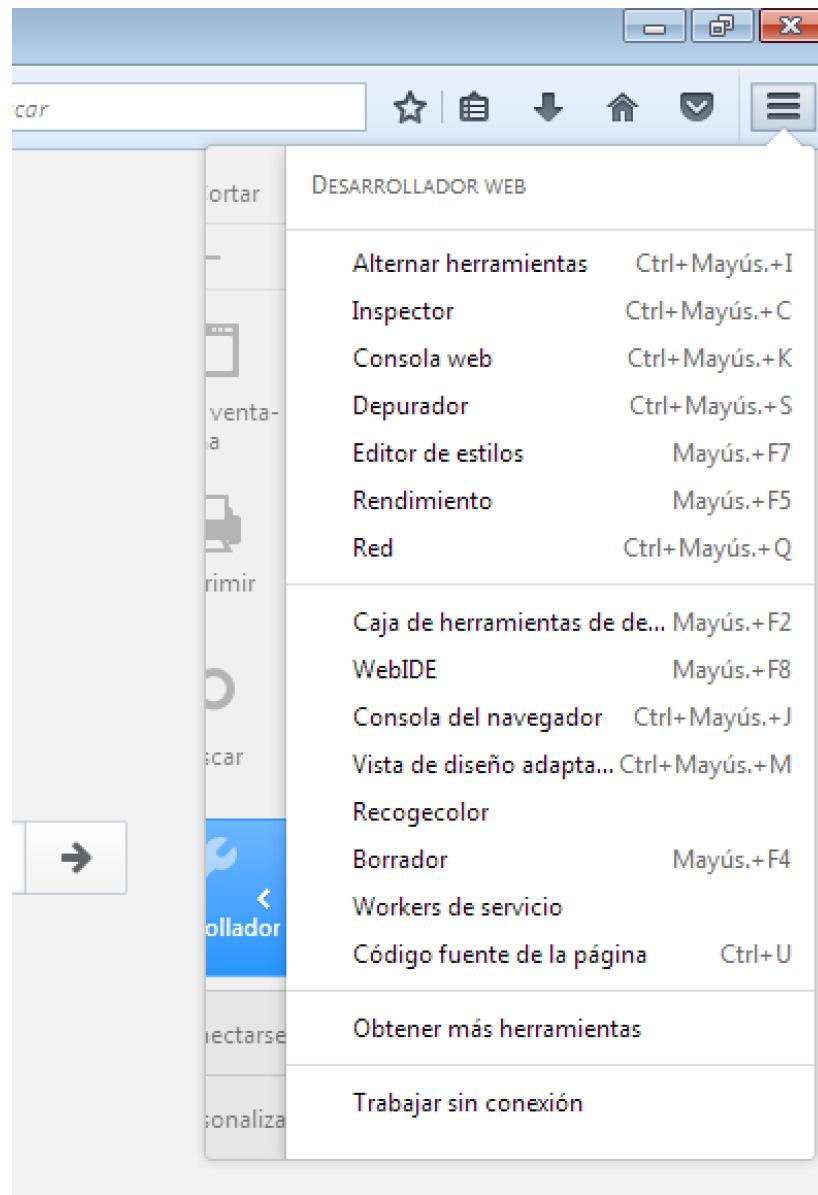


### 3.1.3.- Mozilla Firefox.

Originalmente se utilizaba como herramienta para inspeccionar el código fuente en HTML y CSS de la web la extensión Firebug. En las últimas versiones, Mozilla Firefox ya incluye como utilidad nativa herramientas para el desarrollo de la web, entre las que se encuentra un inspector de código con la misma funcionalidad que la extensión Firebug.

Se puede acceder al menú con las opciones de desarrollo accediendo a la pestaña Herramientas->Desarrollador web.





Javier Mejías. opciones de desarrollador en Mozilla Firefox (CC0)

Entre las herramientas de desarrollo nos encontramos el inspector web que nos permite la edición, depuración y monitorización de los lenguajes CSS, HTML y JavaScript en cualquier página web y en tiempo real.

Publicidad Empresa Acerca de

Publicidad Empresa Acerca de

+ Inspector Consola Depurador Editor de estilos Rendimiento Memoria Red

Reglas Calculado Animaciones Tipografías

Privacidad Condiciones Configuración

Google

Buscar con Google Voy a tener suerte

Google.es también en: català gallego euskara

html > body > #hp-wsq

Javier Mejías. Inspector web en Mozilla Firefox. (CC0)

### **3.1.4.- Firebug.**

Firebug es un complemento para el navegador web Firefox que integra diversas herramientas de desarrollo web. Permite la edición, depuración y monitorización de los lenguajes CSS, HTML y JavaScript en cualquier página web y en tiempo real.

Para instalar la extensión es tan sencillo como acceder [aquí](#) y hacer clic en el botón de "Aregar a Firefox". Puede no ser compatible con las últimas versiones.

Características fundamentales:

- ✓ Muestra de forma jerárquica el arbol de etiquetas (lo que lo hace mas accesible).
- ✓ Pestaña DOM: Permite recorrer el Documento de Modelado de Objetos de la web en ejecución así como modificar parámetros.
- ✓ Pestaña Consola: Permite la ejecución de comandos en tiempo real.
- ✓ Pestaña CSS. Para la edición y modificación de los archivos [CSS](#).
- ✓ Pestaña Script: Para visualizar y modificar los documentos o métodos JavaScript cargados en la ejecución de la web.
- ✓ Pestaña Net: Permite la visualización de todos y cada uno de los recursos de la página.

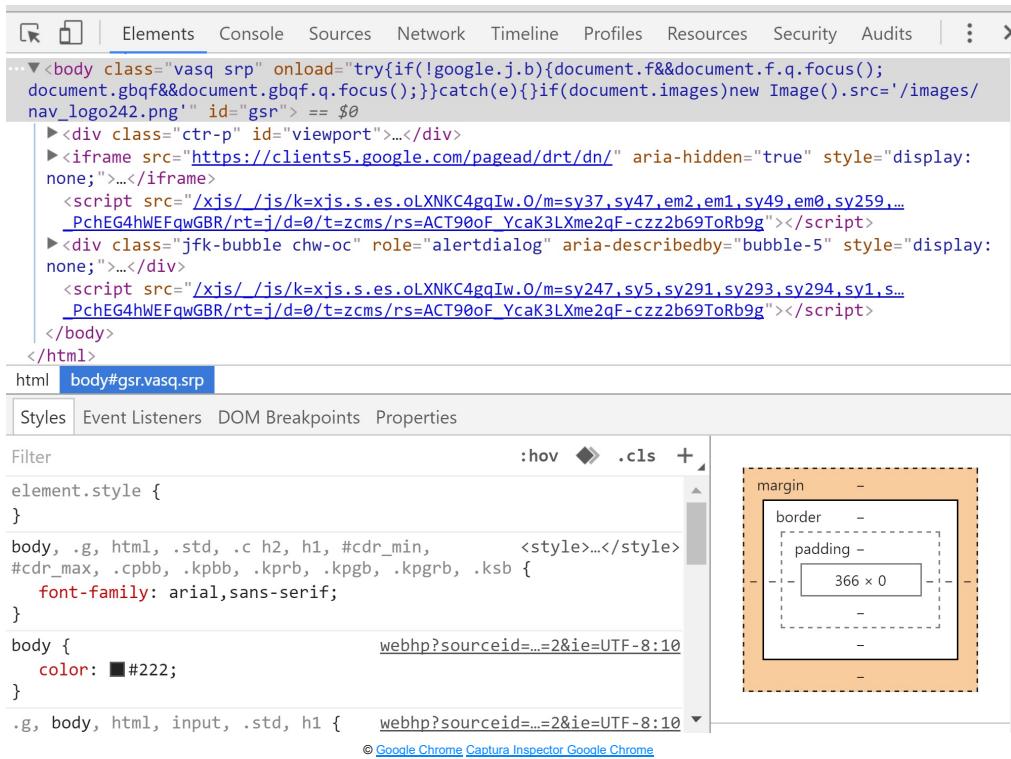


#### **Para saber más**

En la [web oficial de firebug](#) tienes acceso tanto a la documentación como a la comunidad de Firebug.

### 3.1.5- Inspector de Chrome.

Chrome es un navegador que provee de diversas herramientas para la edición y depuración de la web en tiempo real. Esto quiere decir que podemos incluir modificaciones y visualizarlas o ejecutarlas en tiempo real.



Para realizar estas modificaciones es tan sencillo como hacer doble clic en el ítem que se desea modificar. Los cambios serán a nivel local, por lo que no quedarán guardados en el servidor.

Por otro lado uno de los problemas actuales relacionados con las hojas de estilo CSS es como se verá en los dispositivos actuales. Muchos desarrolladores indican que la web se verá bien en todos los dispositivos porque es responsive, pero esto no siempre es así. Para poder comprobarlo puedes hacerlo con [Responsive Web Desing Tester](#)



#### Ejercicio Resuelto

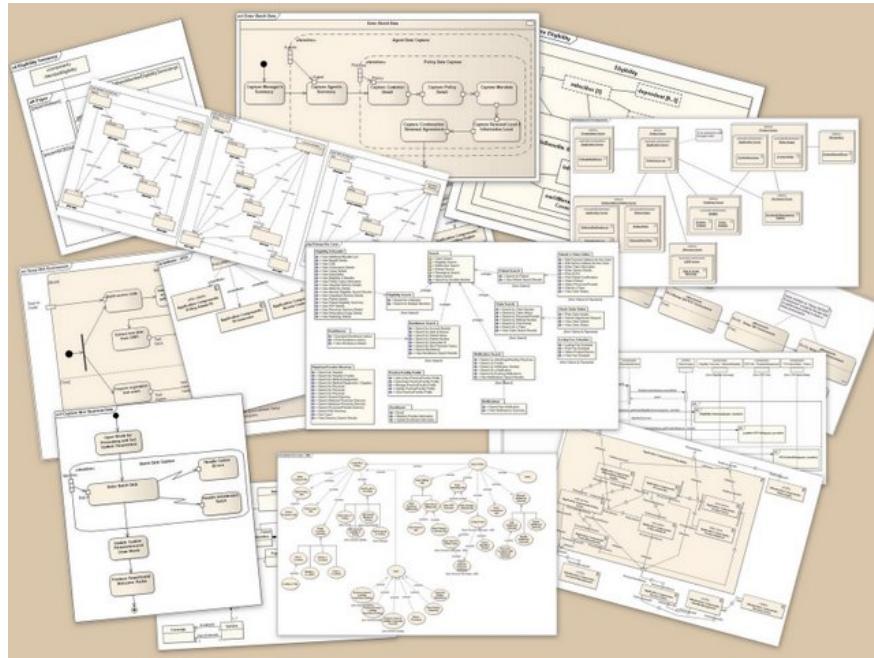
Vamos a modificar de forma dinámica el texto de una web. Para ello accede a la pestaña Elements y busca el texto que se esté visualizando en ese momento. Comprueba que no sea una imagen.

Cuando lo encuentres (te puedes ayudar del buscador usando la combinación de teclas Ctrl+F), cambialo y fíjate como cambia en la web.

[Mostrar retroalimentación](#)

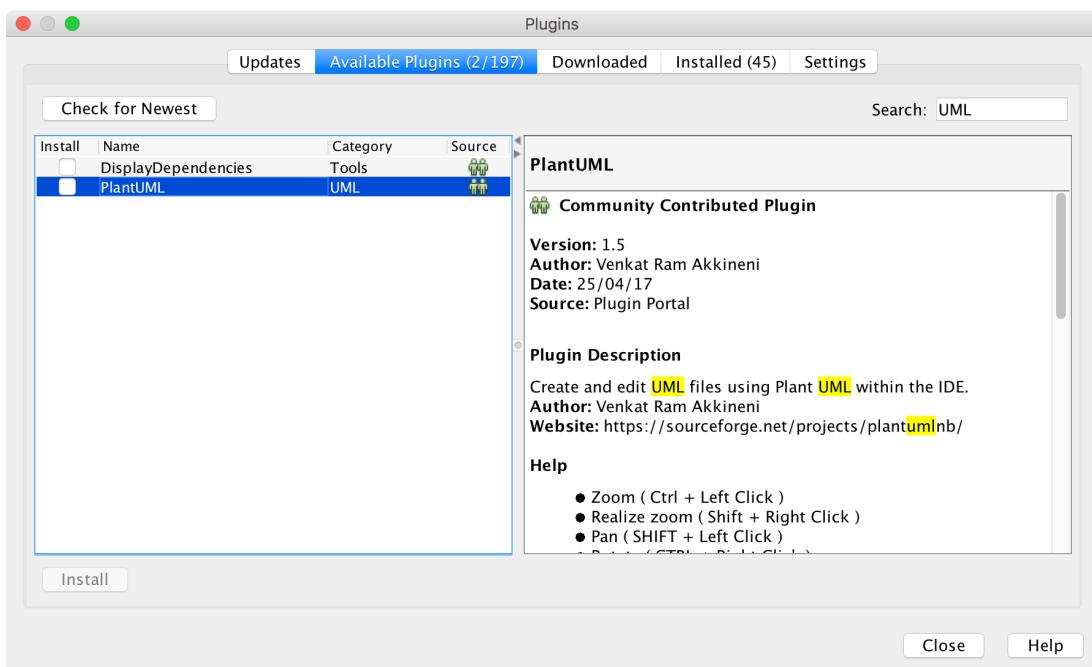
## 3.2.- Plugin UML para netbeans

UML es un lenguaje de modelado Unificado. Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema software.



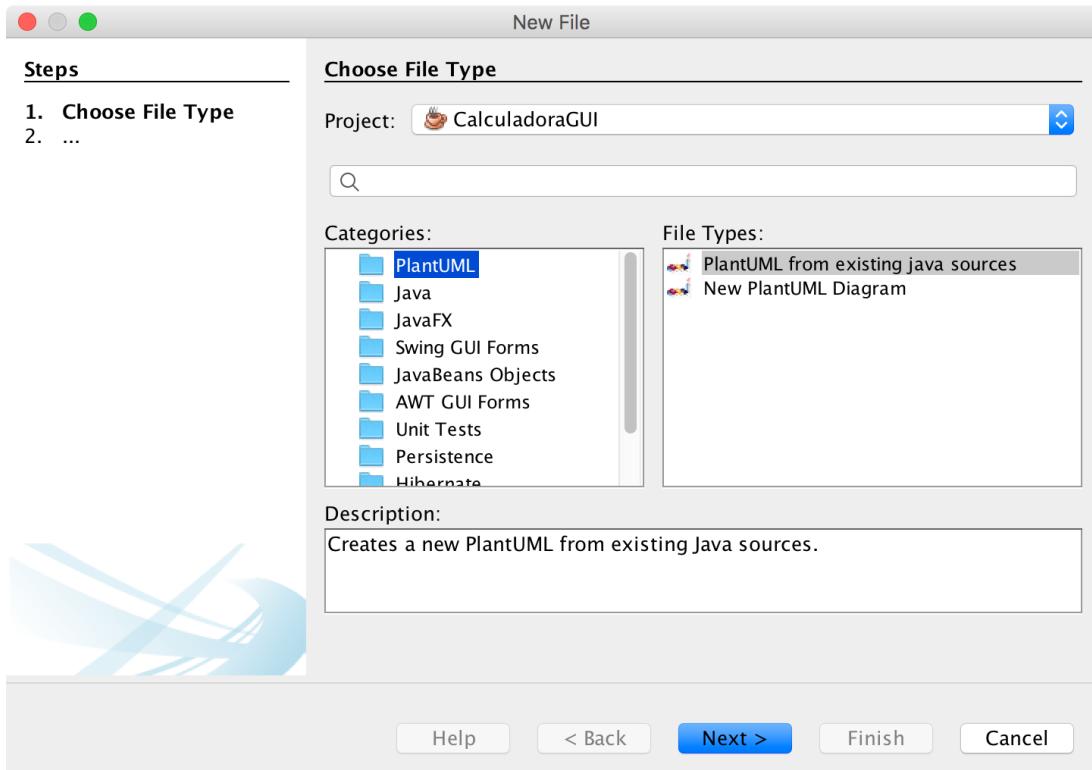
Por Kishorekumar 62 (Trabajo propio). Collage de diagramas UML. (CC0)

Podemos acceder a través de la opción Tools->Plugins al gestor de Plugins de Netbeans.



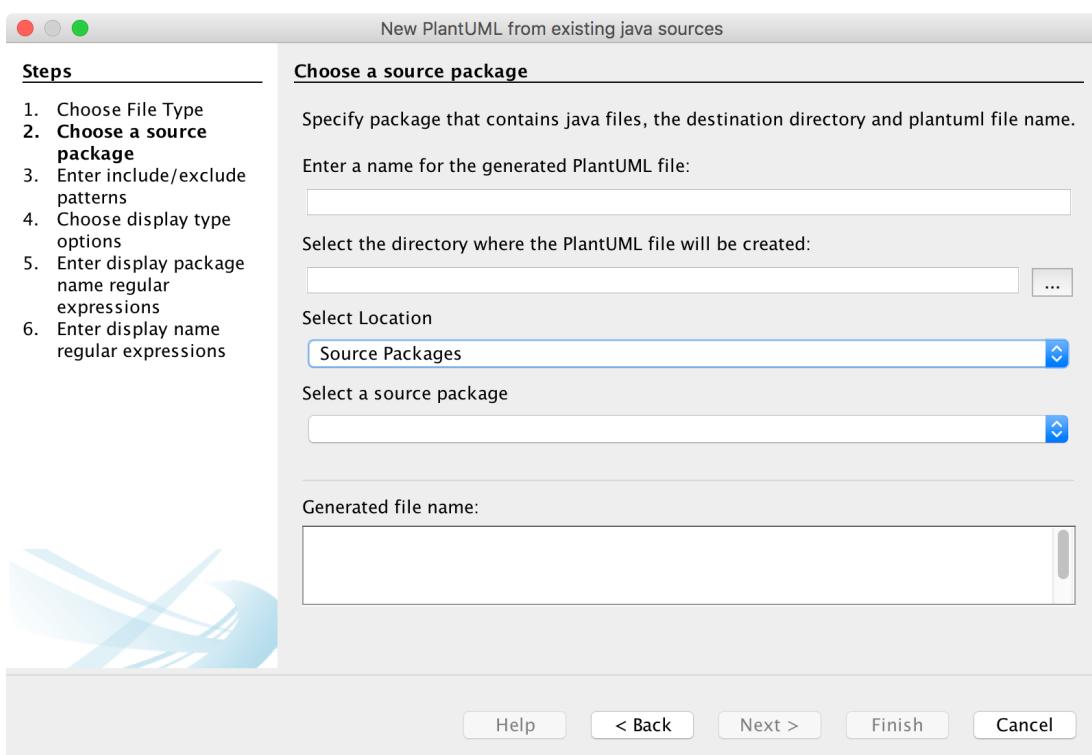
Javier Mejias. Ventana para gestionar los plugins de Netbeans (CC0)

Buscamos los plugins relacionados con UML y llevamos a cabo su instalación. Una vez instalado, como si de un fichero nuevo se tratara podemos crear los distintos diagramas de UML. Accedemos a File->New File y seleccionamos la opción PlantUML por ejemplo.



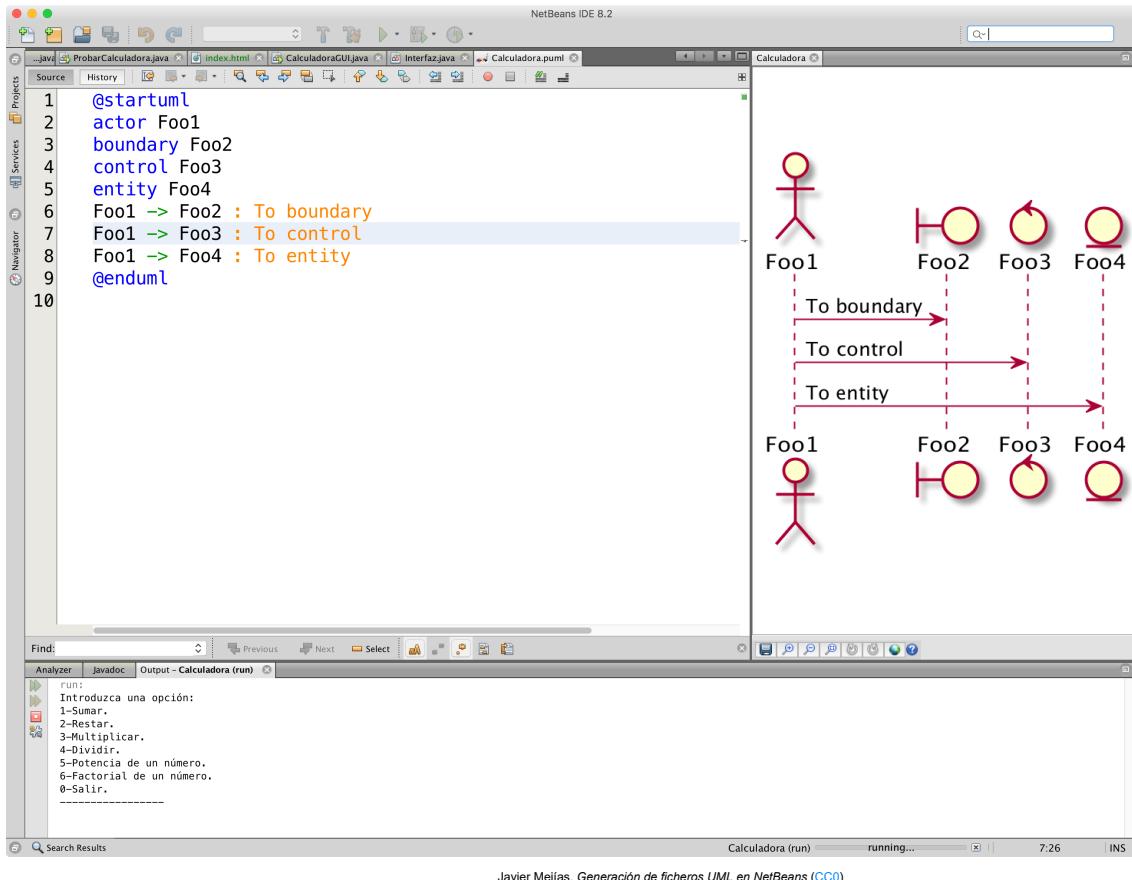
Javier Mejías. Pantalla de NetBeans donde se permite crear un nuevo diagrama UML. ([CC0](#))

Una vez seleccionemos PlantUML, nos aparecerá un asistente que nos ayudará en el proceso de creación del diagrama UML.



Javier Mejías. Asistente para la creación del diagrama UML en NetBeans. ([CC0](#))

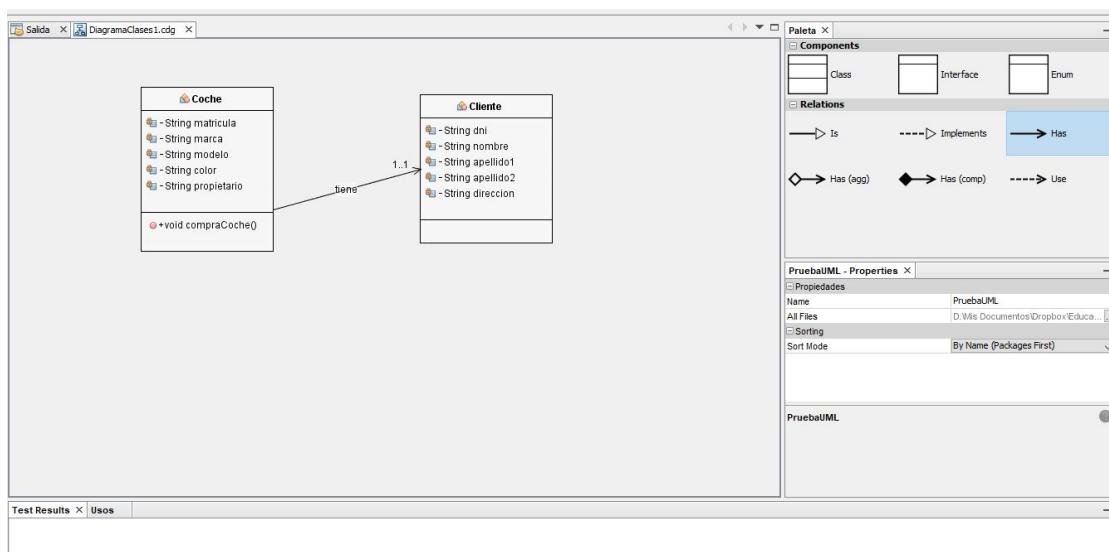
Una vez tenemos creado nuestro fichero .puml dentro del proyecto, podemos utilizando la sintaxis correspondiente, generar nuestros propios diagramas de UML.



Para que se genere el diagrama UML, habrá que compilar el proyecto sin errores.

El lector que esté interesado en conocer más información sobre planUML puede consultar [la página oficial del proyecto](#).

Existen otros plugins como easyUML, es solo para crear diagramas de clases, pero su uso es mucho más intuitivo, ya que hace uso de Drag & Drop.



### 3.3.- Software para generar documentación del código fuente.

Uno de los principales objetivos que debe de satisfacer el software desarrollado es su reutilización. Uno de los axiomas principales radica en no reinventar la rueda. Para ello es fundamental reutilizar el software que ya hemos desarrollado, o bien otros programadores, siempre y cuando su licencia lo permita.

Por este motivo es muy importante desarrollar código fuente bien documentado. Además de la documentación propia del código fuente, es importante que exista una documentación de las clases y de ficheros fuentes desarrollados.

Existen herramientas software que facilitan el proceso de generar la documentación a partir de nuestro código fuente. En este epígrafe vamos a centrarnos en JavaDoc, que permite generar la documentación del código fuente escrito en Java y está disponible en NetBeans. Decir que existen programas orientados a otros lenguajes de programación como PHPDoc.

#### JavaDoc

JavaDoc nos permite obtener la documentación a partir de nuestro código fuente. Para ello en el código tenemos que insertar una serie de etiquetas, las cuales vamos a ver a continuación:

- **@author** - El nombre del autor del proyecto por ejemplo Javier Mejías
- **@version** - La versión del proyecto, por ejemplo 2.3
- **@see** - Añade una referencia a una clase, método o enlace web
- **@param** - Nombre de parámetro utilizado en una función indicando para qué sirve.
- **@return** - El resultado devuelto por la función.
- **@exception** - Excepciones que lanza nuestra función o clase.
- **@throws** - Nombre de la excepción junto con su descripción.
- **@deprecated** - Añade una alerta al usuario de que el método que sigue a continuación ya no debe usarse y que será eliminado en versiones posteriores.

Introduciremos como comentario al comienzo de cada función y/o clase las etiquetas anteriores que nos interesen. Veamos el siguiente ejemplo.

```
* and open the template in the editor.  
*/  
package javaejemplo;  
  
/**  
 * @author javiermejias  
 * @version - 1.0 mayo de 2017  
 */  
public class Usuario {  
    private String login;  
    private String password;  
  
    /**  
     * @return el login del usuario. Variable de tipo String.  
     */  
    public String getLogin() {  
        return login;  
    }  
}
```

Javier Mejías. Código fuente documentado. [\(CC0\)](#)

Una vez tengamos generado todo nuestro proyecto, tendremos que ir a la opción Ejecutar (Execute)->Generar JavaDoc (Generate JavaDoc) y de forma automática se nos generará toda la documentación de nuestro proyecto.

The screenshot shows a JavaDoc interface for the class `Usuario`. At the top, there are tabs for "All Classes", "PACKAGE", "CLASS" (which is highlighted in orange), "USE", "TREE", "DEPRECATED", "INDEX", and "HELP". Below the tabs, there are links for "PREV CLASS", "NEXT CLASS", "FRAMES", and "NO FRAMES". A summary bar includes links for "SUMMARY", "NESTED", "FIELD", "CONSTR", "METHOD", "DETAIL", "FIELD", "CONSTR", and "METHOD". The main content area starts with the package name `javaejemplo` and the class name `Class Usuario`. It then lists the class's inheritance: `java.lang.Object` and `javaejemplo.Usuario`. The code snippet for the class definition is shown as:

```
public class Usuario  
extends java.lang.Object
```

Below the code, there is a "Constructor Summary" section with a "Constructors" tab (highlighted in orange) and a "Constructor and Description" tab. The "Constructor and Description" tab contains the constructor `Usuario()`.

Further down is a "Method Summary" section with three tabs: "All Methods" (highlighted in orange), "Instance Methods", and "Concrete Methods". The "All Methods" tab displays methods categorized by modifier and type. The listed methods are:

Modifier and Type	Method and Description
java.lang.String	<code>getLogin()</code>
java.lang.String	<code>getPassword()</code>
void	<code>setLogin(java.lang.String login)</code>
void	<code>setPassword(java.lang.String password)</code>

Below the method summary is a "Methods inherited from class `java.lang.Object`" section, which lists methods like `clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, and `wait`.

At the bottom of the page, there is a footer note: "Javier Mejías. Documentación generada por JavaDoc. (CC0)".

Esta herramienta está disponible en múltiples IDEs, como por ejemplo Eclipse.

None