

campusformacion.imf.com © EDICIONES ROBLE, S.L.
Manuel Vázquez Enriquez

Instalación y uso de entornos de desarrollo © EDICIONES ROBLE, S.L.

campusformacion.imf.com © EDICIONES ROBLE, S.L.
Manuel Vázquez Enriquez

campusformacion.imf.com © EDICIONES ROBLE, S.L.
Manuel Vázquez Enriquez

Indice

Instalación y uso de entornos de desarrollo	3
I. Introducción	3
II. Objetivos	3
III. Contenidos	3
3.1. Funciones de un entorno de desarrollo	3
3.2. Tipos de entornos de desarrollo	6
3.3. Características de un buen entorno de desarrollo	7
3.4. Instalación de un entorno de desarrollo	8
3.4.1. Instalación del JDK	8
3.4.2. Instalación de Eclipse	11
3.4.3. Instalación de NetBeans	14
3.5. Uso básico de un entorno de desarrollo	15
3.5.1. Vídeos sobre las funciones básicas de Eclipse y NetBeans	16
IV. Actividades interactivas	21
Actividad 1: Encuentra los tipos de entorno de desarrollo	21
Actividad 2: Relaciona los siguientes elementos	21
V. Resumen	21
VI. Lecturas obligatorias	22
Ejercicios	23
Ejercicio 1	23
Se pide	23
Recursos	24
Enlaces de Interés	24
Bibliografía	24
Glosario.	24

Instalación y uso de entornos de desarrollo

I. Introducción

Los entornos de desarrollo integrado son indispensables en la explicación de aplicaciones software, ya que comprenden un conjunto completo de herramientas para ayudar a los programadores e ingenieros a realizar un software de calidad.

Dada la complejidad de las aplicaciones que se demandan actualmente, sin la ayuda de estas herramientas, resultaría una tarea muy difícil de abordar. Además, teniendo en cuenta que la figura del programador solitario ha desaparecido, los entornos de desarrollo son fundamentales para poder trabajar en equipo junto a otros desarrolladores.

II. Objetivos

Conocer qué es y para qué se usa un entorno de desarrollo integrado (IDE).

Identificar las funcionalidades que un IDE puede contener para la ayuda de creación de software.

Conocer los IDE de mercado más conocidos y actuales.

Manejar la instalación de entornos de desarrollo como NetBeans y Eclipse.

Utilizar las funciones básicas de los entornos de desarrollo NetBeans y Eclipse.

III. Contenidos

3.1. Funciones de un entorno de desarrollo



Un IDE es una aplicación que nos ayuda en la tarea de creación de proyectos de software. Está pensada para facilitar la escritura del mismo por medio de asistentes y de realizar comprobaciones para que todo esté correcto. Es decir, es un **editor de código, compilador, depurador y constructor de interfaz gráfica (GUI)**.

Los IDE proporcionan un entorno de trabajo amigable para la mayoría de los lenguajes existentes hoy en día. Sin ellos, la labor de realizar las aplicaciones que nos demandan actualmente sería prácticamente imposible.



Un IDE nos permite escribir el código de una forma sencilla resaltando la sintaxis y posee un corrector sintáctico, también dispone normalmente de un compilador o intérprete, un depurador, entre otras funcionalidades.

Editor de texto

El editor de texto es la parte que nos permite escribir el código fuente del programa.

Es capaz de reconocer, resaltar y cambiar los colores de las variables, las cadenas de caracteres, las palabras reservadas, las instrucciones, el inicio y fin de los corchetes...

De esta manera, el código fuente será mucho más visual y cómodo; además se podrán reconocer los errores a simple vista.



No solo nos corregirá en caso de detectar un error, también mostrará sugerencias e indicaciones del posible fallo cometido.

Compilador

El compilador existe porque, como vimos en la unidad anterior, el código fuente tecleado por el programador (escrito en un lenguaje de alto nivel) necesita ser traducido a un programa escrito en lenguaje de bajo nivel (lenguaje máquina).



El IDE realiza estas compilaciones de nuestro código para que podamos ir realizando pruebas de nuestro código mientras lo estamos desarrollando.

Intérpretes

Los intérpretes se diferencian de los compiladores en que **solo realizan la traducción a medida que se va ejecutando el código**.



En lenguajes interpretados, como PHP o Javascript, el IDE va ejecutando el código línea a línea como si se tratara del entorno de producción (se conoce como máquina virtual).

Depurador

El depurador (*Debugger*) es una herramienta que nos permite comprobar el funcionamiento de nuestro código, **examina instrucción a instrucción la ejecución de un programa** y los cambios que se produzcan en las variables de este o en los registros del procesador.

Permite detener el programa en un punto de ruptura para examinar los caminos o el valor de los datos que tienen en cada momento.

VENTAJAS

Ventajas de usar un entorno de programación:

La **curva de aprendizaje es muy baja**; al disponer de herramientas visuales, facilita la comprensión en la realización de tareas.

Formatea el código para ayudarnos a realizar un código ordenado, realizando las tabulaciones, incorporando llaves...

Funciones de refactorización para renombrar variables, funciones...

Nos muestra **warnings y errores** de sintaxis en pantalla de algo que no va a funcionar al interpretar o compilar.

Crea proyectos y ayuda a mantener todos los archivos del programa localizados.

3.2. Tipos de entornos de desarrollo

Casi todos los lenguajes (C++, PHP, Python, Java, C#, Delphi, Visual Basic...) tienen un entorno de desarrollo específico, como Visual Studio para los lenguajes de Microsoft. Podremos usar también otros más genéricos que admiten lenguajes diferentes, como Eclipse o NetBeans.

Imagen 1. Ejemplos de entornos integrados de desarrollo (IDE)



Imagen 1. Ejemplos de entornos integrados de desarrollo (IDE).

Fuente: elaboración propia a partir de NetBeans®, Visual Studio®, Oracle®, IntelliJ Idea®, Microsoft Visual C++®, J®, Eclipse®, Clarion 10®, Android

Eclipse

Entorno de desarrollo de *software* libre, capaz de trabajar con muchos lenguajes. Es uno de los principales entornos para programar con Java y está muy extendido en las empresas. Tiene la posibilidad de instalar *plug-in* para dotarlo de más posibilidades.

NetBeans

Entorno de desarrollo muy utilizado en la formación (entorno académico). Con él podemos realizar codificaciones en PHP, C++, lenguajes web, aunque es muy utilizado para el lenguaje Java. Tiene la posibilidad de ampliar sus funciones instalando *plug-in*.

BlueJ

Entorno de desarrollo dirigido al aprendizaje del lenguaje Java, no está muy extendido a nivel profesional. Destaca por su sencillez al incluir algunas funcionalidades dirigidas a tener mayor facilidad para comprender la programación orientada a objetos.

JBuilder

Software comercial, aunque se pueden obtener versiones de prueba gratuitas en la web.

JCreator

Se pueden obtener versiones de prueba o versiones simplificadas gratuitas en la web. Este IDE está escrito en C++ y permite herramientas para desarrollos gráficos, lo cual lo hace más rápido y eficiente que otros IDE.

3.3. Características de un buen entorno de desarrollo

Dependiendo del lenguaje de programación, los requerimientos que solicitamos a un entorno de desarrollo pueden ser diferentes, pero se considera que un buen entorno de desarrollo debe tener las siguientes **características**:

Poder ser instalado en múltiples plataformas.

Soporte para varios lenguajes de programación.

Control de versiones.

Asistentes de sintaxis.

Extensiones y *plugins* adicionales.

Integración con *framework*.

Depurador.

Importar y exportar proyectos.

Manual de usuario y ayuda.

3.4. Instalación de un entorno de desarrollo

Como hemos comentado, existen multitud de entornos de desarrollo en el mercado. En esta unidad veremos cómo instalar los entornos de desarrollo Eclipse y NetBeans, dos de los IDE más extendidos en el desarrollo de aplicaciones.

3.4.1. Instalación del JDK

Tanto en el caso de Eclipse como de NetBeans, necesitamos instalar el entorno de desarrollo JDK, pues tanto si vamos a utilizar el lenguaje JAVA como si no, están realizados en este lenguaje y lo necesitan para su funcionamiento.

Primer paso



El primer paso es descargar el JDK desde [la página de Oracle](#).

En esta página (Imagen 2.), pulsamos el botón de descarga, en el momento de escribir estas líneas se encuentra en la versión Java Platform (JDK) 8u111/8u112.



Imagen 2. Página de descargas Oracle.

Fuente: Oracle.

Segundo paso

El siguiente paso es escoger la plataforma con la que estamos trabajando, el tipo de SO en el que vamos a instalar esta aplicación. Tenemos a nuestra disposición la descarga para todos los SO actuales.



Es importante, si estamos trabajando con Windows, ver si nuestro sistema es de 32 bits (x86) o de 64 bits. Una vez aceptados los términos de la licencia, procedemos a su descarga (Imagen 3.).

Oracle Menu Search Account Country/Region Call

Oracle Technology Network / Java / Java SE / Downloads

Java SE Development Kit 8 Downloads

Thank you for downloading this release of the Java™ Platform, Standard Edition Development Kit (JDK™). The JDK is a development environment for building applications, applets, and components using the Java programming language.

The JDK includes tools useful for developing and testing programs written in the Java programming language and running on the Java platform.

See also:

- Java Developer Newsletter: From your Oracle account, select **Subscriptions**, expand **Technology**, and subscribe to **Java**.
- Java Developer Day hands-on workshops (free) and other events
- Java Magazine

JDK 8u161 checksum
JDK 8u162 checksum

Java SE Development Kit 8u161

You must accept the **Oracle Binary Code License Agreement for Java SE** to download this software.

☐ Accept License Agreement ☐ Decline License Agreement

Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	77.92 MB	jdk-8u161-linux-arm32-vfp-hflt.tar.gz
Linux ARM 64 Hard Float ABI	74.88 MB	jdk-8u161-linux-arm64-vfp-hflt.tar.gz
Linux x86	168.96 MB	jdk-8u161-linux-i586.rpm
Linux x86	183.76 MB	jdk-8u161-linux-i586.tar.gz
Linux x64	166.09 MB	jdk-8u161-linux-x64.rpm
Linux x64	180.97 MB	jdk-8u161-linux-x64.tar.gz
macOS	247.12 MB	jdk-8u161-macosx-x64.dmg
Solaris SPARC 64-bit (SVR4 package)	139.59 MB	jdk-8u161-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	99.29 MB	jdk-8u161-solaris-sparcv9.tar.gz
Solaris x64	140.57 MB	jdk-8u161-solaris-x64.tar.Z
Solaris x64	97.02 MB	jdk-8u161-solaris-x64.tar.gz
Windows x86	198.54 MB	jdk-8u161-windows-i586.exe
Windows x64	206.51 MB	jdk-8u161-windows-x64.exe

Imagen 3. Selección de plataforma para la instalación del JDK.

Fuente: Oracle.

La instalación es completamente asistida, por lo que al ejecutar el archivo descargado se procederá a incorporarlo a nuestro SO.

<https://player.vimeo.com/video/265298061>

3.4.2. Instalación de Eclipse



Para instalar Eclipse, lo primero que tenemos que hacer es **descargar** de la [web oficial](#) el **programa**. Esta página nos facilita un enlace para descargar la última versión publicada, pero existen muchos paquetes y versiones disponibles.

Si queremos ver las diferentes posibilidades de instalación podemos darle al enlace: [Download Packages](#). Aquí encontramos distintas compilaciones de la aplicación, así como versiones anteriores del mismo.

Si estás empezando y vas a programar en Java, se recomienda instalar **Eclipse IDE for Java EE Developers** (Imagen 4.).

The screenshot shows the Eclipse website's download page. At the top, there's a navigation bar with links like 'GETTING STARTED', 'MEMBERS', 'PROJECTS', and 'MORE+'. Below this, a breadcrumb trail shows 'HOME / DOWNLOADS / ECLIPSE IDE DOWNLOADS'. The main content area features a large banner for 'Try the Eclipse Installer' with a download button and a link to 'Find out more'. Below the banner, there's a list of download packages. The 'Eclipse IDE for Java EE Developers' package is highlighted, showing its size (331 MB) and download count (462,860). Other packages listed include 'Eclipse IDE for Java Developers', 'Eclipse IDE for C/C++ Developers', and 'Eclipse for PHP Developers'. To the right, there's a 'RELATED LINKS' section with links to 'Compare & Combine Packages', 'New and Noteworthy', 'Install Guide', 'Documentation', 'Updating Eclipse', and 'Forums'. Below that, a 'MORE DOWNLOADS' section lists other builds like 'Eclipse Oxygen (4.7)', 'Eclipse Neon (4.6)', 'Eclipse Mars (4.5)', 'Eclipse Luna (4.4)', and 'Eclipse Kepler (4.3)'. A 'Hint' section at the bottom right states that a Java runtime environment (JRE) is needed to use Eclipse.

Imagen 4. Selección de paquetes de Eclipse.
Fuente: Eclipse.

Una vez descargado el archivo de instalación, solo tenemos que ejecutarlo, cambiar la ruta donde queremos que se instale y esperar, ya que la instalación es desatendida.

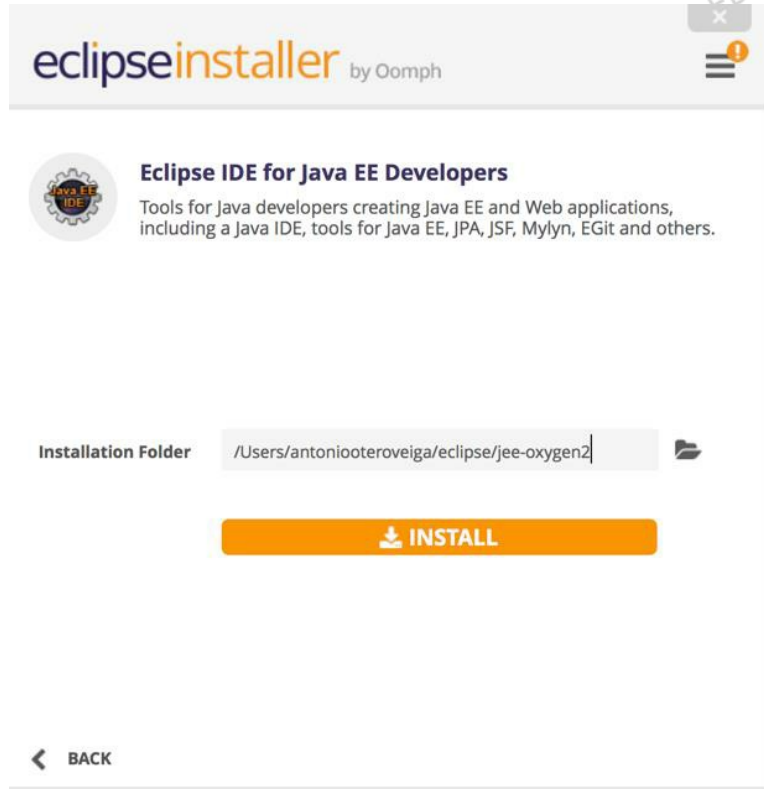


Imagen 5. Instalador.

Fuente: Eclipse.

Al terminar la instalación, nos preguntará dónde queremos ubicar el *workspace* (Imagen 6.), esta carpeta es el lugar en el cual queremos guardar nuestros proyectos. Podemos tener varios *workspace*, según nuestras necesidades.

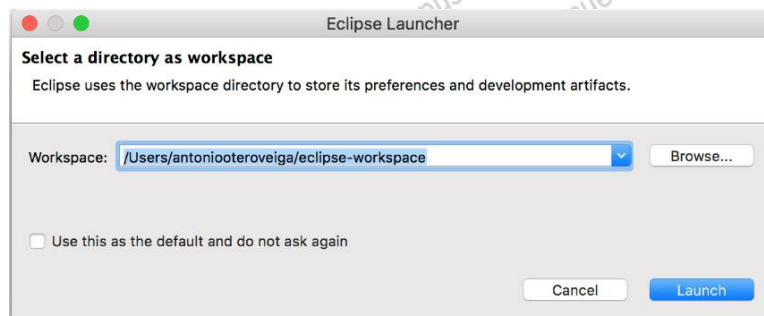


Imagen 6. Selección de *workspace*.

Fuente: Eclipse.

Ahora ya tenemos nuestro **IDE Eclipse instalado**. Primero muestra una pantalla de bienvenida. En cuanto la cerremos, veremos los paneles de nuestra aplicación.

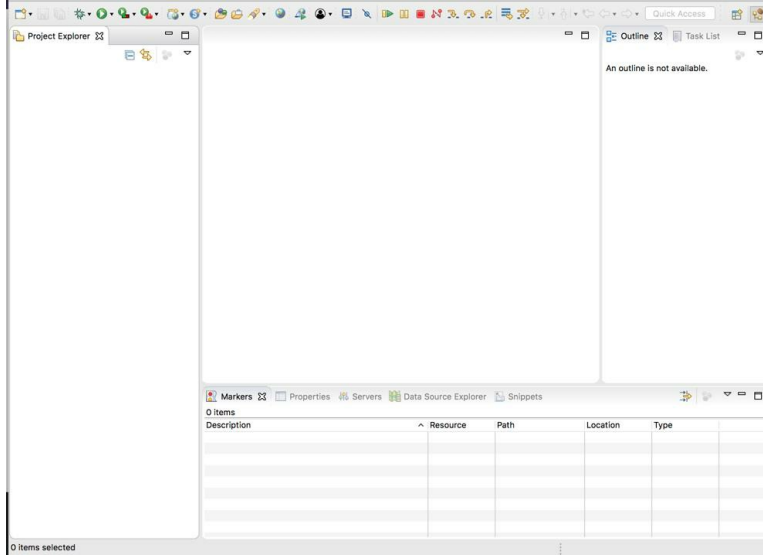


Imagen 7. Pantalla del entorno de desarrollo Eclipse.

Fuente: Eclipse.

<https://player.vimeo.com/video/265300560>

<https://player.vimeo.com/video/265302144>

3.4.3. Instalación de NetBeans



Para instalar NetBeans, nos dirigimos a la URL y buscamos el botón de descargas.

Una vez en esta pantalla (Imagen 9.), elegimos los lenguajes que queremos que nos dé soporte el programa. También podemos escoger versiones anteriores del programa o el idioma en que queremos la aplicación (aunque puede parecer útil instalar la versión en castellano, **es aconsejable instalar la versión en inglés**, ya que mayoritariamente la documentación que encontremos estará en este idioma).

NetBeans IDE 8.2 Download 8.1 | 8.2 | Development | Archive

Email address (optional):

Subscribe to newsletters: ☒ Monthly ☐ Weekly ☒ NetBeans can contact me at this address

IDE Language: English Platform: Mac OS X

Note: Greyed out technologies are not supported for this platform.

NetBeans IDE Download Bundles

Supported technologies *	Java SE	Java EE	HTML5/JavaScript	PHP	C/C++	All
NetBeans Platform SDK	•	•				•
Java SE	•	•				•
Java FX	•	•				•
Java EE		•				•
Java ME						—
HTML5/JavaScript		•	•	•		•
PHP			•	•		•
C/C++					•	•
Groovy						•
Java Card™ 3 Connected						—
Bundled servers						
GlassFish Server Open Source Edition 4.1.1		•				•
Apache Tomcat 8.0.27		•				•

Download buttons and sizes: Java SE (Free, 116 MB), Java EE (Free, 242 MB), HTML5/JavaScript (Free, 142 MB), PHP (Free, 142 MB), C/C++ (Free, 147 MB), All (Free, 277 MB)

Imagen 9. Opciones de descarga de NetBeans.

Fuente: NetBeans.

Una vez descargado el archivo de instalación, lo ejecutamos y es totalmente desatendida, por lo que ya disponemos de nuestro entorno de desarrollo.

<https://player.vimeo.com/video/265291147>

3.5. Uso básico de un entorno de desarrollo

A lo largo del módulo, iremos utilizando las herramientas que nos proporcionan los entornos de desarrollo para mostrar diferentes funciones. En nuestro caso, nos centraremos en Eclipse. Al tratarse de una aplicación que está muy extendida en las empresas, parece la opción más lógica.



- Documentación oficial de Eclipse.
- Documentación oficial de NetBeans.

3.5.1. Vídeos sobre las funciones básicas de Eclipse y NetBeans

Vídeo: Creación de proyectos y primeros pasos con el código

<https://player.vimeo.com/video/265301530>

Vídeo: Ayuda y acceso a la documentación

<https://player.vimeo.com/video/265303942>

Vídeo: Configuración y personalización de Eclipse

<https://player.vimeo.com/video/265301522>

Vídeo: Asistentes de código

<https://player.vimeo.com/video/265305700>

Vídeo: Codificación y generación de GET-SET

<https://player.vimeo.com/video/265310875>

Vídeo: Detección de errores

<https://player.vimeo.com/video/265307861>

Vídeo: Exportación de proyectos

<https://player.vimeo.com/video/265307884>

Vídeo: Importación de proyectos

<https://player.vimeo.com/video/265307885>

Vídeo: Instalación de plugins y módulos adicionales con Markplace

<https://player.vimeo.com/video/265308976>

Vídeo: Instalación de dependencias externas con Eclipse

<https://player.vimeo.com/video/265310206>

IV. Actividades interactivas

Actividad 1: Encuentra los tipos de entorno de desarrollo

Tipos de entorno de desarrollo.



Sopa de letras

Actividad 2: Relaciona los siguientes elementos

Qué debe y qué no debe tener un IDE.



Relacionar elementos

V. Resumen



En esta unidad, hemos visto:

- Qué es un entorno de desarrollo y sus funciones básicas.
- Ha quedado constatada la importancia que tienen estas herramientas en la vida profesional de un desarrollador y lo importante que es tener un conocimiento de las mismas.



La instalación y el manejo de estas herramientas necesitará de una actualización constante por parte del programador, por la velocidad con la que surgen nuevas utilidades para aumentar el rendimiento de la producción de las aplicaciones.

VI. Lecturas obligatorias



- Si deseas aprender más sobre cómo funcionan los entornos de programación, puedes visitar este enlace: [Entornos de programación](#).
- Para aprender más sobre cómo desarrollar aplicaciones en el Entorno de Desarrollo Eclipse tienes un completo manual en este enlace: [Manual Eclipse](#).
- Si prefieres utilizar NetBeans para desarrollar tus aplicaciones, puedes aprender todo lo necesario en el siguiente enlace: [Manual NetBeans](#).

Ejercicios

Ejercicio 1

Duración estimada del ejercicio



240
minutos



Este ejercicio es **opcional**, si tienes alguna duda puedes contactar con tu tutor.

Se pide

El alumno debe de crear un manual mostrando paso a paso, con capturas de pantalla y textos explicativos. Las instalaciones de los siguientes elementos:

- Eclipse, ultima versión disponible.
- Gradle (Buidlship Integration).
- UML Generator (generación de diagramas UML).
- WindowBuilder (aplicaciones con interfaz de usuario gráfica).

Una vez instalado, se debe generar un proyecto JAVA explicando los pasos necesarios para su creación.

Recursos

Enlaces de Interés



<https://www.eclipse.org/documentation>
<https://www.eclipse.org/documentation>



<https://netbeans.org/kb/index.html>
<https://netbeans.org/kb/index.html>

Bibliografía

- **Eduardo Basterrechea Molina . (2000). Ingeniería del software de gestión II. 2017, de Scrib Sitio web:** <https://es.scribd.com/doc/51957036/errores-clasicos-en-el-desarrollo-de-software>.: Eduardo Basterrechea Molina . (2000). Ingeniería del software de gestión II. 2017, de Scrib Sitio web: <https://es.scribd.com/doc/51957036/errores-clasicos-en-el-desarrollo-de-software>.
- **Ian Sommerville. (2011). Ingeniería del Software. España: Pearson Educación.** Ian Sommerville. (2011). Ingeniería del Software. España: Pearson Educación.
- **Rapid Development. Steve McConnell. Microsoft Press 1996. McGrawHill. Desarrollo y gestión de proyectos informáticos.** Rapid Development. Steve McConnell. Microsoft Press 1996. McGrawHill. Desarrollo y gestión de proyectos informáticos.

Glosario.

- **Compilador:** El código fuente tecleado por el programador (escrito en un lenguaje de alto nivel), necesita ser traducido a un programa escrito en lenguaje de bajo nivel (lenguaje máquina).
- **Depurador (Debugger):** El depurador es una herramienta que nos permite comprobar el funcionamiento de nuestro código
- **Eclipse:** Entorno de desarrollo de software libre, capaz de trabajar con muchos lenguajes. Es uno de los principales entornos para programar con JAVA y está muy extendido en las empresas. Tiene la posibilidad de instalar plugins para dotarlo de más posibilidades.
- **IDE:** Un entorno de desarrollo integrado (IDE), es una aplicación que nos ayuda en la tarea de creación de nuestros proyectos de software.
- **Intérprete:** Los intérpretes se diferencian de los compiladores en que solo realizan la traducción a medida que se va ejecutando el código.
- **JDK:** Java Development Kit (JDK) Software con herramientas de desarrollo para la creación de programas en Java.
- **NetBeans:** Entorno de desarrollo muy utilizado en la formación (entorno académico). Con él podemos realizar codificaciones en PHP, C++, Lenguajes Web, aunque es muy utilizado para el lenguaje JAVA. Tiene la posibilidad de ampliar sus funciones instalando plugins.

- **Refactorización:** Herramienta para realizar cambios en nuestros códigos, capaz de modificar todas las dependencias que se encuentre en el contexto del programa.