

Notas técnicas de JAVA - Tip en detalle Nro. 1

(Lo nuevo, lo escondido, o simplemente lo de siempre pero bien explicado)

Cómo conformar un entorno de programación JAVA (serie de varios tips)

Parte I: Selección e instalación de un IDE gratuito

Tema:	IDE, Integrated Development Environment, Herramientas, jEdit, Eclipse, "freeware".
Descripción:	El objetivo de esta serie de Tips es brindar pautas y sugerencias sobre cómo conformar un entorno de desarrollo adecuado para JAVA. En éste, el primero de la serie, analizamos la opción de utilizar un IDE gratuito o tipo GNU (General Public License).
Nivel:	Básico
Fecha pub:	Septiembre 2003

"Notas Técnicas de JAVA" se envía con frecuencia variable y absolutamente *sin cargo* como un servicio a nuestros clientes. Contiene notas/recursos/artículos técnicos desarrollados en forma totalmente objetiva e independiente. Teknoda es una organización de servicios de tecnología informática y NO comercializa hardware, software ni otros productos. Si desea suscribir otra dirección de e-mail para que comience a recibir los tips envíe un mensaje desde esa dirección a develop@teknoda.com, indicando su nombre, empresa a la que pertenece, cargo y país.

Próximos Tips:

- Concepto de "frameworks" y STRUTS.
- Cómo conformar un entorno de Programación JAVA Parte II: Application Servers

Tabla de contenido

El objetivo de la serie de Tips "Cómo conformar un entorno de programación JAVA" es brindar pautas y sugerencias sobre cómo seleccionar, obtener, instalar las distintas herramientas que integran un ambiente JAVA: IDE, Application Servers, etc.. En éste, el primero de la serie, analizamos la opción de utilizar un **IDE gratuito (GNU)**. Los siguientes puntos serán cubiertos en el presente tip:

- I. Por qué un IDE ?**
La oferta de IDEs en el mundo JAVA
Plug-ins
Lista parcial de los IDEs más populares
- II. La alternativa de un IDE sin cargo**
Software "Freeware/Shareware" y Software con licencia GNU
Características deseables en un IDE JAVA
- III. Ejemplos de IDE GNU: jEdit, NetBeans y Eclipse**

IV. **Instalación y configuración de jEdit**

Prerrequisito de instalación
Descarga e instalación
Ejecución y configuración

V. **La alternativa IDE Eclipse**

Prerrequisito de instalación
Descarga e instalación

VI. **Conceptos Adicionales para tener en cuenta**

Diferencias entre J2SE, J2EE, J2ME
Empezar el aprendizaje “en crudo”, la mejor recomendación
Casos especiales: Los IDE vinculados con una determinada plataforma

VIII. **Dónde obtener información adicional**

I. **Por qué un IDE?**

Cualquiera sea el lenguaje que se utilice, es necesario contar con medios para que los desarrolladores puedan editar el código, compilar y ejecutar los programas. Los entornos de programación pueden variar desde la forma más rudimentaria, basada en una línea de comandos desde donde invocar al editor, al compilador y probar el programa, hasta los sofisticados entornos integrados de desarrollo (IDE).

Un “Integrated Development Environment” es una **herramienta de soporte al proceso de desarrollo de software** que integra las funciones básicas de edición de código, compilación y ejecución de programas con otras tales como:

- o Editor con corrección sintáctica y coloreo de la misma al momento de codificar.
- o Herramientas gráficas.
- o Soporte integrado para la compilación y ejecución de programas.
- o Relación entre errores de compilación y el código fuente, para facilitar su corrección.
- o Opciones de Debugging.

La oferta de IDEs en el mundo JAVA

Las características de un lenguaje como JAVA hacen que sea difícil encarar un desarrollo corporativo si no es de la mano de un buen IDE.

Si bien existe cierto consenso en que es mejor iniciarse en el lenguaje trabajando primero “en crudo”, con un editor raso desde una línea de comandos (ver notas finales), en algún punto será necesario cambiar esta modalidad de trabajo por un IDE. La particular curva de aprendizaje de JAVA, y el carácter modular del código es un factor determinante por el cual los desarrolladores recurren a un Entorno de Desarrollo Integrado (IDE) con el fin de facilitar su tarea de aprendizaje y desarrollo.

La oferta de IDEs para lenguaje JAVA es realmente abundante y variada, y es destacable el nivel de sofisticación que se puede obtener aún de los productos sin costo (“freeware”, “shareware”, o GNU).

La elección de un IDE no es simple, y mucho menos en los entornos JAVA donde existen muchas variantes de programación. Además de las preferencias personales, la afinidad con otros IDEs que uno ya conozca de otros entornos es relevante. Asimismo, un producto puede ser ideal para quienes explotan determinada funcionalidad del lenguaje (por ejemplo, applets), pero no contemplar otras (por ejemplo, servlets). Otros sirven sólo para una plataforma (por ejemplo Linux), y no funcionan para otras (por ejemplo, Windows). Algunos tienen un “footprint” que permite instalarlos en casi cualquier máquina, mientras que otros serán más adecuados para un entorno Enterprise.

También existen casos especiales donde un IDE se presenta fuertemente integrado con un Application Server para permitir desarrollos asociados a ese producto. Por ejemplo, SAP ofrece el entorno NetWeaver Studio, basado en Eclipse junto con el SAP Web Application Server. IBM ofrece WebSphere Development Studio para desarrollar con el WebSphere Application Server. (Ver notas finales).

Probablemente será necesario considerar y probar más de una opción antes de establecerse por una de ellas. La mayoría de los programadores experimentados han trabajado con varios IDEs distintos. Cambiar de IDE no debería ser traumático, salvo en un proyecto muy grande, muy avanzado y con muchos programadores.

Además de los IDEs de licencia gratuita, la mayoría de los IDEs comerciales, ofrecen versiones “light” que también son gratis, pero con funcionalidades recortadas. Otros ofrecen la versión completa pero en calidad de evaluación por una cantidad limitada de días.

Plug-ins

Un aspecto a considerar también en la elección de un IDE es la posibilidad de acoplarle “plug-ins” (agregados) para cubrir cierta funcionalidad específica. Muchos IDE se componen de un Kernel básico, al que se le puede opcionalmente sumar uno o más “plug-ins” con funciones no cubiertas en el kernel: por ejemplo, un project viewer, una herramienta de debugging, un administrador de clases, etc. La integración de un IDE con una determinada plataforma servidora, como en el caso de SAP NetWeaver o IBM WebSphere, también se logra con un conjunto de “plug-ins”.

Algunos IDE ofrecen un “plug-in” manager para comandar la selección y el acoplado de los mismos.

Lista parcial de los IDEs más populares

A continuación presentamos una lista parcial de algunos de los IDE para JAVA más difundidos. Esta lista no pretende abarcar la totalidad de las herramientas disponibles, sino brindar una primera idea del universo de productos para explorar. La lista enumera los principales IDEs disponibles SIN COSTO, ya sea como “freeware”, “shareware”, o GNU General Public License.

Por “**freeware**” se entiende el software que se puede copiar SIN COSTO y distribuir libremente, pero sin la posibilidad de modificarlo, ya que **no se incluyen los fuentes** con la distribución del mismo.

En el caso de “**shareware**” se suma al concepto anterior, la necesidad de registrarse como usuario y pagar por ese mantenimiento. Por último, existen las licencias GNU, también sin costo, que se explican en la próxima sección.

IDE's Freeware o GNU (El concepto de GNU se explica más abajo)

- Sun ONE Studio 4 (Update 1) Community Edition
- JEdit
- JCreator
- NetBeans

Shareware

- WinEdit
- TextPad

Commercial Ware

- Sun ONE Studio 4 Enterprise Edition
- IntelliJ IDEA
- Metrowerks Code Warrior
- Borland JBuilder
- IBM Visual Age for Java
- IBM WebSphere Development Studio Client

II. La alternativa de un IDE SIN CARGO

Como se mencionara anteriormente, el objetivo de este tip es sugerir algunas alternativas de IDE entre las disponibles SIN COSTO, que posean las más importantes funcionalidades para la correcta utilización del lenguaje Java. Para esto necesitamos previamente realizar algunas consideraciones sobre software “freeware/shareware” y software con licencia GNU.

Software “Freeware/Shareware” y Software con licencia GNU

La licencia GNU o **General Public License**, desarrollada por la **FSF (Free Software Foundation)**, permite la instalación y uso de un programa GNU en una o más computadoras, sin ninguna limitación. También se puede modificar el programa para adaptarlo a necesidades particulares. Además, se puede distribuir el programa GNU habiéndole realizado alguna o ninguna modificación.

El software con licencia GNU se puede vender o ceder, siempre y cuando se facilite el código fuente junto con el código ejecutable que se está distribuyendo.

El software con licencia GNU es libre, es decir, no es ni "shareware" ni "freeware". Por ejemplo, en el caso de “freeware” o “shareware” se los puede conseguir en un cd-rom al comprar una revista o descargarlos de internet, pero no incluyen el código fuente y no se tiene permiso para modificarlos. Por lo tanto, no son “libres”.

Para información adicional sobre GNU (General Public License) se sugiere acceder al sitio oficial a través de: <http://www.gnu.org>

Características deseables en un IDE JAVA

Además de las características obvias y generales que un IDE debe ofrecer, mencionadas anteriormente (editor con corrección sintáctica y coloreo, herramientas gráficas, soporte integrado para la compilación y ejecución de programas, opciones de debugging, etc.) consideramos que un **IDE JAVA debe tener:**

- **Code Completion:** Capacidad de desplegar automáticamente el package, las clases que lo componen y los métodos disponibles para cada una de ellas.
- **Method Viewer:** Ventana con la lista de métodos de una clase, en un árbol expandible.
- **Class Viewer:** Ventana con la lista de clases del documento activo, en un árbol expandible.
- **File Viewer:** Ventana con la lista de todos los archivos de un proyecto, en un árbol expandible.

Sin esto, el IDE no sería más que un editor potente.

III. Ejemplos de IDEs GNU: jEdit, NetBeans y Eclipse

Como explicáramos en secciones anteriores, en la elección de un IDE intervienen una gran cantidad de variables. Nuestras sugerencias, por lo tanto, intentan ser una orientación para explorar algunos productos interesantes, pero cada desarrollador deberá hacer un análisis posterior, mirando distintas opciones y tratando de clarificar su necesidad para encontrar el más adecuado. Todos los productos tienen web sites desde donde se pueden ver sus características, requerimientos, realizar la descarga, y acceder a documentación de usuario.

Entre los productos SIN COSTO, como mínimo sugerimos evaluar los siguientes:

jEdit: (www.jedit.org)

jEdit es un entorno de desarrollo JAVA maduro y bien diseñado, que ha estado en evolución por más de cinco años. Está escrito en JAVA, y corre sobre Windows, Unix, Mac, OS/2 y VMS. Se ofrece con el código fuente completo, bajo los términos de una licencia GNU General Public License. jEdit se configura a partir de un kernel muy básico, y ofrece una gran cantidad de plug-ins para ampliar selectivamente su funcionalidad. A

través de los plug-ins disponibles, es posible configurarlo para reconocimiento de sintaxis de fuentes escritos en hasta 80 lenguajes diferentes. Durante la instalación del mismo, sugerimos revisar la lista de plug-ins, para incluir aquellos que sean necesarios.

NetBeans: (<http://java.sun.com/j2se/1.4.2/download.html>)

NetBeans es un IDE que cubre la codificación, compilación, testing, y debugging de aplicaciones J2SE. Incluye un editor muy completo con herramientas visuales. Se ofrece para bajar desde el Web Site de Sun, en un “bundle” (paquete) junto con el JSDK.

Eclipse: (www.eclipse.org)

Eclipse es una plataforma de desarrollo extensible, basada en Java y de tipo open-source. Originalmente fue desarrollada por Alphaworks, laboratorio de desarrollo de IBM, pero actualmente está manejado por un consorcio de varias empresas, (Borland, Hitachi, etc.). Lo interesante de Eclipse, es que conforma una suerte de “estándar” o framework. Muchos vendedores de software basan hoy sus ofertas de IDE en Eclipse, al que suman conjuntos de plug-ins. Por ejemplo, SAP NetWeaver Studio, e IBM WebSphere están basados en Eclipse.

IV. Instalación y configuración de jEdit

Entre los IDEs listados anteriormente, hemos elegido a modo de ejemplo el producto **jEdit**. Como alternativa, mostramos también en la próxima sección la instalación del IDE **Eclipse**.

Prerrequisito de instalación

Para que la instalación de jEdit resulte exitosa, **deberá primero descargarse el JDK (Java Standard Development Kit)** desde la página oficial de Sun www.java.sun.com. **Este prerrequisito deberá tenerse en cuenta para todos aquellos IDEs que no lo incluyan.**

Los siguientes pasos indican como descargar e instalar el **JDK Java 2 Platform - Standard Edition (J2SE)**:

1. Ingresar al site www.java.sun.com/downloads.
2. Hacer un click en el primer campo desplegable de la ventana, con leyenda **Full J2SE Technology Downloads List**.
3. Seleccionar en la lista desplegada **All platforms** dentro de **J2SE 1.4.2**. Luego hacer un click en el botón **Go**.
4. En la siguiente ventana, posicionarse en la leyenda **Download J2SE v 1.4.2_01**. Luego seleccionar el sistema operativo correspondiente y comenzar la descarga haciendo un click en el campo **Download** de la columna **SDK** (El Standard Development KIT incluye al JRE Java Runtime Environment).
5. En la siguiente ventana, presionar **Accept** para aceptar los términos de la licencia y comenzar la descarga en una carpeta creada para este fin.
6. Una vez finalizada la descarga, continuar con la instalación de la manera habitual.

Descarga e Instalación de jEdit

Los siguientes son los pasos necesarios para realizar la descarga e instalación del IDE jEdit:

- 1) Acceder a la página principal de jEdit. Para realizar esta tarea, en el Browser escribir: <http://www.jedit.org/>
- 2) Sobre la página que se muestra en la pantalla siguiente, en el panel izquierdo de la misma, hacer un Click sobre **Download jEdit** (con círculo en rojo).

jEdit - Open Source programmer's text editor

<p>www.jedit.org: Quick Start Guide Download jEdit Compatibility</p>	<h3 style="margin: 0;">Overview</h3> <p>jEdit is a mature and well-designed programmer's text editor that has been in development for over 5 years.</p>	<h3 style="margin: 0;">News</h3> <p>Plugin Central updates (8) <i>Posted on 2003-09-29 08:05</i> <i>2 Comments</i></p>
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------

- 3) Posicionarse en la **Option 2: “Manual download and install on any OS”**, seleccionar **“Latest Development Version”** y hacer Click sobre la componente **Java _Based Installer**.
- 4) Elegir un “mirror” (servidor) en la columna Download desde donde descargar el archivo indicado en el paso anterior, el que permitirá realizar la instalación de jEdit. Este archivo tiene extensión .jar y se denomina **jedit42pre5install.jar**, que corresponde a la última versión. Cuando se le solicite guardar el archivo, hacerlo sobre una carpeta previamente creada. En este ejemplo la carpeta se denomina “javaIDE”.
- 5) Una vez ya descargado el archivo de extensión .jar, copiar el siguiente grupo de instrucciones en el block de notas:


```
C:  

      cd \javaIDE  

      C:\jdk1.4.2\bin\java -classpath C:\javaIDE -jar jedit42pre5install.jar
```

 y guardarlo como **instalador.bat**.
- 6) Abrir una ventana de DOS, posicionarse en el directorio donde **instalador.bat** fue guardado y ejecutarlo. Aparece la ventana **Aceptar licencia GNU**. Aceptar presionando **Next**.
- 7) Luego se visualizará la siguiente ventana que promptea el path donde se desea instalar el producto, por ejemplo:

Install Program in: **C:\Program Files\jEdit 4.2pre5**

Una vez indicado, presionar **Next** para continuar con la instalación.

- 8) En la ventana **“Choose components to install”**, seleccionar los cuatro componentes a instalar. Se recomienda marcar todos los disponibles. Sus nombres son:
 - jEdit text editor (5404 Kb)
 - Default set of macro (highly recommended (168 Kb)
 - API documentation (for macro and plugin development) (8880 Kb)
 - Windows desktop integration (jEditLauncher) (400 Kb)

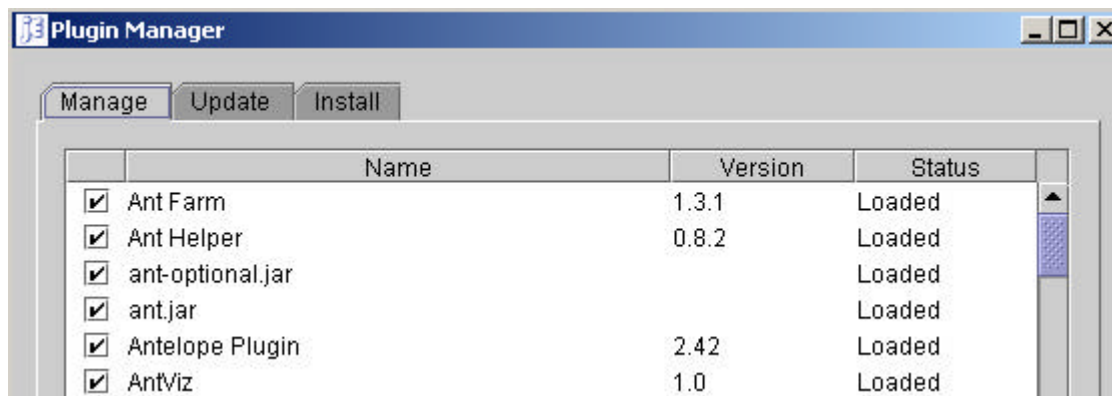
Presionar **Install**. Se visualizará la barra de progreso de la instalación que permanecerá en pantalla por algunos segundos.

- Al terminar el proceso de instalación aparecerá una ventana informando que la instalación se ha completado. Para terminar el proceso de instalación presionar "Finish".

Ejecución y Configuración de jEdit

Una vez completado el proceso de instalación:

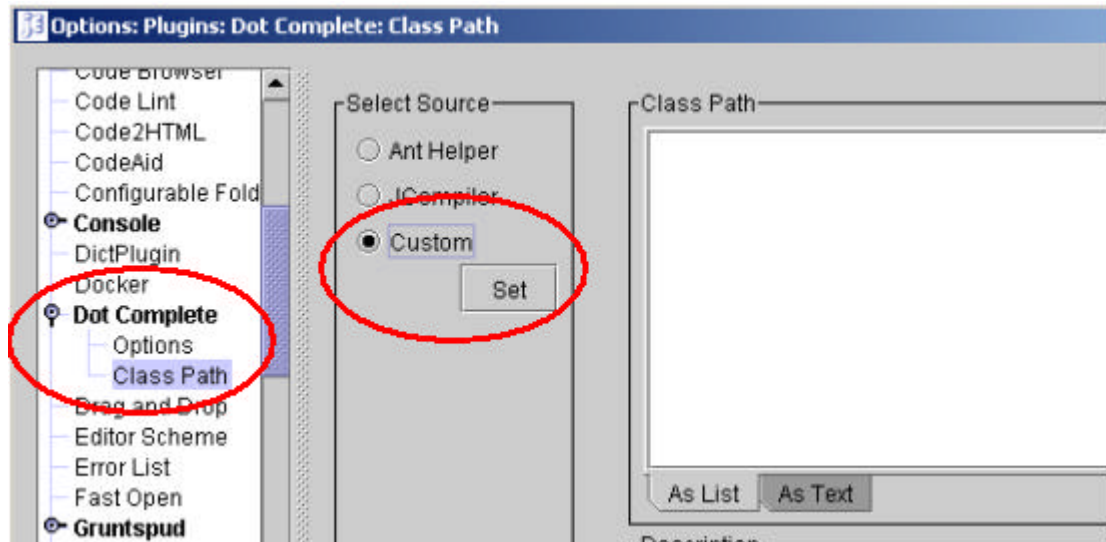
- 1) Desde el escritorio, ejecutar **jEdit.exe**, desde el ícono correspondiente:
- 2) Agregar los plugins necesarios al IDE jEdit. Dentro del jEdit, ubicarse en la parte superior en la opción "Plugins" y luego seleccionar "Plugin Manager".
- 3) En la pantalla del "Plugin Manager" ir a la solapa **Install** y tildar los plugins necesarios. **Entre los plugins disponibles deberá seleccionarse el de nombre AntHelper. Posee la funcionalidad de compilar y ejecutar lenguaje Java.** Observar que en caso de seleccionar un plugin que posea prerequisites, los nombres de los plugins que deberán ser instalados con anterioridad se visualizarán en la lista en color rojo. Luego presionar "Install" en la parte inferior de la pantalla.



4) Una vez instalados todos los componentes (Plugins) deseados, debemos cerrar la aplicación y volverla a ejecutar para que se actualicen todos los plugins anexados.

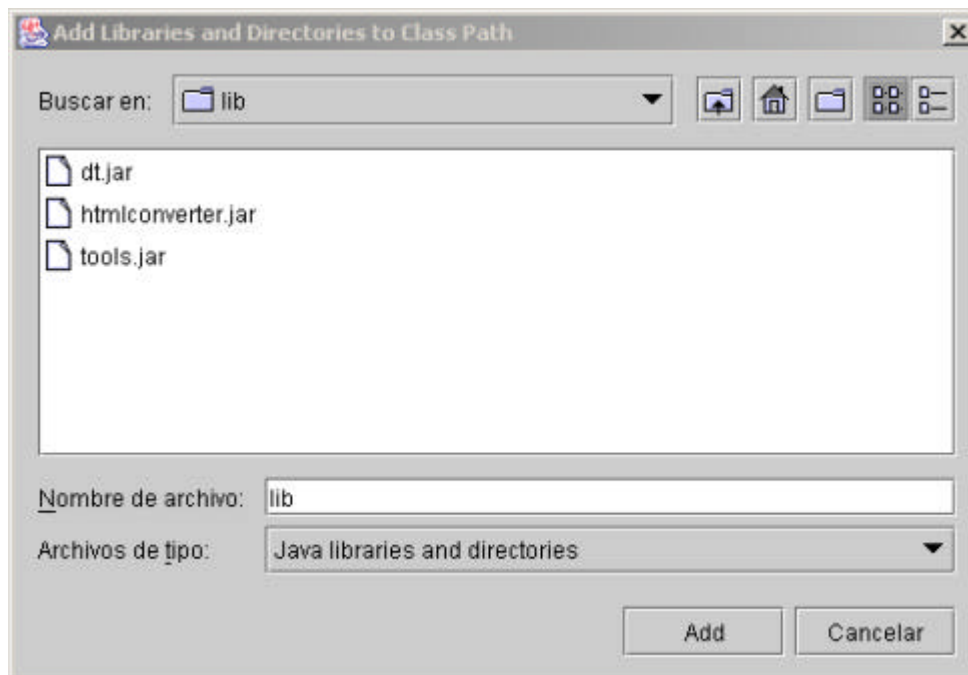
5) Luego proceder a la instalación del plugin "DotComplete". **La existencia de esta componente es importante porque permite desplegar los métodos y atributos de una clase luego de presionar el punto.** Para instalarla:

- a) Ingresar nuevamente al producto, ir a **Plugins** y luego **Plugins Options**.
- b) En la ventana siguiente, sobre el margen izquierdo, se visualiza un árbol desplegable con toda la lista de los plugins instalados. Seleccionar **DotComplete**. Haciendo doble click sobre él, se visualizarán dos opciones. Seleccionar la opción **Class Path** y marcar el radio button **Custom**.



Luego presionar **Set**.

- c) En la ventana **Edit Class Path**, presionar **Add**. Luego, en la ventana **Add Libraries and Directories to Class Path**, deberán seleccionarse los archivos .jar almacenados en el directorio C: \<directorio creado para bajar el JSDK>\J2SDK1.4.2\lib.



Una vez seleccionados los tres archivos .jar, presionar **Add**. En la ventana siguiente **OK** y luego **Apply** y **OK** nuevamente.

- d) Para completar el proceso de instalación del plugin, cerrar el producto y luego volver a ingresar.

Existen otros plugins disponibles que habilitan distintas funcionalidades en jEdit. Su proceso de instalación es similar al descrito anteriormente.

V. La alternativa IDE Eclipse

Eclipse es una plataforma de desarrollo extensible, basada en Java y de tipo open-source. Está comprendida por un framework y un conjunto de servicios para construir un entorno de desarrollo a través de plug-ins. Eclipse incluye los siguientes componentes: un conjunto standard de plug-ins, un JDT (Java Development Tools) y un PDE (Plug-in Development Environment). Este último habilita a los desarrolladores a extender Eclipse y construir así herramientas que puedan ser integradas a su vez, de manera transparente, con el entorno Eclipse.

En la actualidad existe **Eclipse.org**, un consorcio de vendedores de herramientas de desarrollo de software liderado por IBM, e integrado por Borland, Hitachi, Oracle y SAPt entre otras. La comunidad Eclipse fue formada para promover la creación de nuevas herramientas y crear mejores entornos de desarrollo e integración de productos. Para más información, ingrese a www.eclipse.org.

Prerrequisito de instalación

Requiere de la instalación previa del JSDK. Se recomienda seguir los pasos descritos en los prerrequisitos de instalación del jEdit.

Descarga e instalación

1) Ingresar la siguiente URL: <http://www.eclipse.org/downloads/index.php>. En la página que se visualiza, seleccionar el link "Main Eclipse Download Servers". Luego aparecerá la siguiente ventana:

eclipse project downloads

latest downloads from the eclipse project



Latest Downloads

On this page you can find the latest [builds](#) produced by the [Eclipse Project](#). To get started run the program and and go through the user and developer documentation provided in the online help system. If you have problems downloading the drops, contact the [webmaster](#). If you have problems installing or getting the workbench to run, [check out the Eclipse Project FAQ](#), or try posting a question to the [newsgroup](#). All downloads are provided under the terms and conditions of the [Eclipse.org Software User Agreement](#) unless otherwise specified.

Build Type	Build Name	Build Date
Latest Release	2.1.1	Fri, 27 Jun 2003 -- 15:45 (-0400)

Ingresar en el link marcado en rojo.

- 2) En la ventana siguiente, clickear en la columna **Download File**, habiendo seleccionado previamente el sistema operativo. En este caso, se selecciona la plataforma **Windows** y el tipo de descarga **http**.

Eclipse SDK

The Eclipse SDK includes the Eclipse Platform, Java development tools, and Plug-in Development Environment, including source and both user and programmer documentation. If you aren't sure which download you want... then you probably want this one.

Eclipse does not include a Java runtime environment (JRE). You will need a 1.3 level Java runtime or Java development kit (JDK) installed on your machine in order to run Eclipse. [Click here](#) if you need help finding a Java runtime.

Status Platform

- ✓ Windows 98/ME/2000/XP
- ✓ Linux (x86/Motif) ([Supported Versions](#))

Download File

- ([http](#)) ([ftp](#)) eclipse-SDK-2.1.1-win32.zip ([md5](#))
- ([http](#)) ([ftp](#)) eclipse-SDK-2.1.1-linux-motif.zip ([md5](#))

- 3) Luego continuar con los pasos necesarios para cualquier descarga habitual.
- 4) Descomprimir el archivo .zip generado. Luego, buscar en la carpeta el archivo eclipse.exe y ejecutarlo.

VI. Conceptos adicionales para tener en cuenta

Diferencias entre J2SE, J2EE, J2ME

En este tip se ha trabajado con la edición estándar de Java, denominada J2SE (Java 2 Standard Edition). **J2SE** es un conjunto de herramientas y APIs que permiten desarrollar applets y aplicaciones. En los orígenes de Java este kit cubría las necesidades del mercado ya que se podía desarrollar todo lo demandado, pero si se trata de utilizar Java en un entorno corporativo aparecen nuevas exigencias a cubrir.

Hoy en día los productos de software tienden a ser distribuidos y a compartir información en distintas ubicaciones físicas, esto es lo que se denomina comúnmente “Enterprise Applications”. Para cubrir las nuevas necesidades del mercado Sun ofrece **J2EE** (Java EnterpriseEdition). La razón por la que no se integró el paquete J2EE en el J2SE, que ya era conocido, es que las aplicaciones distribuidas suponen tratar tareas específicas de red, operaciones de E/S y manejo de Base de datos con un enfoque totalmente distinto al que requiere una aplicación convencional. Además estas aplicaciones requieren manejar una significativa cantidad de almacenamiento y de memoria. **J2EE** surgió por los nuevos requisitos que presentaban las aplicaciones .

También se encuentra el **J2ME** que es un paquete que brinda soporte a las aplicaciones para móviles, PDAs, agendas electrónicas, etc. Estos dispositivos no cuentan con mucha memoria, tienen limitaciones computacionales, los displays son pequeños, por lo tanto **J2ME** resume la funcionalidad de J2SE adaptándola a los requisitos mínimos necesarios para aplicarlos a los dispositivos móviles, inalámbricos.

Empezar el aprendizaje “en crudo”, la mejor recomendación

La mayoría de los especialistas involucrados en la enseñanza de JAVA, recomiendan aprender los rudimentos del lenguaje sin el auxilio de un IDE, y este enfoque parece ser el más acertado.

En primer lugar, sólo un contacto con el lenguaje “a la intemperie” permite entender su filosofía, la sintaxis de las construcciones, etc. En segundo lugar, trabajar de entrada con un IDE implica sumar el aprendizaje del IDE propiamente dicho al del lenguaje, sumando y mezclando complejidades. Si bien la mayoría de los IDE son muy sencillos, es necesario conocer como funcionan para poder operar con ellos. Por último, se pretende que un programador JAVA no esté circunscripto a un determinado entorno, y que tenga la flexibilidad de poder desarrollar con distintos productos.

Una vez desarrollados los primeros programas, antes de avanzar a los elementos más complejos del lenguaje, se puede introducir un IDE en el aprendizaje. En un comienzo, tal vez convenga no utilizar toda la funcionalidad del IDE, sino aprovecharlo como un poderoso editor.

Casos especiales: Los IDE vinculados con una determinada plataforma

Como se mencionara anteriormente, existen casos especiales donde un IDE se presenta fuertemente integrado con un Application Server para permitir desarrollos asociados a ese producto. Por ejemplo, SAP ofrece el entorno NetWeaver Studio, basado en Eclipse junto con el SAP Web Application Server. IBM ofrece WebSphere Development Studio para desarrollar con el WebSphere Application Server.

En algunos de estos casos, el IDE ofrecido contempla funcionalidad específica imposible de encontrar en otro. Por ejemplo, WebSphere Development Studio, entre otras facilidades, permite trabajar con herramientas de WebFacing para aplicaciones AS/400. SAP NetWeaver permite desarrollar Web Dynpros desde cero o partiendo de Dynpros (programas característicos de SAP que se utilizan en entornos desktop).

Esta funcionalidad suele presentarse como un conjunto de “plug-ins”.

VII. Dónde Obtener Información Adicional

www.thefreecountry.com

www.java.sun.com

Copyright 2003 Teknoda S.A. Septiembre 2003. JAVA es marca registrada de Sun. SAP, R/3 y ABAP son marcas registradas de SAP AG. AS/400 es marca registrada de IBM. Todas las marcas mencionadas son marcas registradas de las empresas proveedoras.

La información contenida en este artículo ha sido recolectada en la tarea cotidiana por nuestros especialistas a partir de fuentes consideradas confiables. No obstante, por la posibilidad de error humano, mecánico, cambios de versión u otro, Teknoda no garantiza la exactitud o completud de la información aquí volcada.

Dudas o consultas develop@teknoda.com