

# 1.F. Introducción al IDE.

Sitio: Aula Virtual  
Curso: Programación\_DAM  
Libro: 1.F. Introducción al IDE.  
Imprimido por: LUIS PUJOL  
Día: martes, 5 de noviembre de 2019, 09:34

# Tabla de contenidos

1 Entornos Integrados de Desarrollo (IDE).

2 ¿Qué son?

3 IDE's actuales.

# 1 Entornos Integrados de Desarrollo (IDE).

En los comienzos de Java la utilización de la línea de comandos era algo habitual. El programador escribía el código fuente empleando un editor de texto básico, seguidamente, pasaba a utilizar un compilador y con él obtenía el código compilado. En un paso posterior, necesitaba emplear una tercera herramienta para el ensamblado del programa. Por último, podía probar a través de la línea de comandos el archivo ejecutable. El problema surgía cuando se producía algún error, lo que provocaba tener que volver a iniciar el proceso completo.



Estas circunstancias hacían que el desarrollo de software no estuviera optimizado. Con el paso del tiempo, se fueron desarrollando aplicaciones que incluían las herramientas necesarias para realizar todo el proceso de programación de forma más sencilla, fiable y rápida. Para cada lenguaje de programación existen múltiples entornos de desarrollo, cada uno con sus ventajas e inconvenientes. Dependiendo de las necesidades de la persona que va a programar, la facilidad de uso o lo agradable que le resulte trabajar con él, se elegirá entre unos u otros entornos.

Para el lenguaje de programación Java existen múltiples alternativas, siendo los principales entornos de desarrollo **NetBeans** (que contó con la ayuda de la empresa Sun, y actualmente con el apoyo de la empresa Oracle), **Eclipse** (desarrollado originalmente por IBM para sustituir a su VisualAge), **JCreator** (producto comercial de Xinox Software y que cuenta con versión gratuita y de pago) y **IntelliJ IDEA** (desarrollado por JetBrains y que ha sido adaptado posteriormente por Google para crear el potente **Android Studio** para la programación de móviles). Los dos primeros son gratuitos, con soporte de idiomas y multiplataforma (Windows, Linux, MacOS) y sin ningún género de duda los más usados.

¿Y cuál será con el que vamos a trabajar? El entorno que hemos seleccionado llevar a cabo nuestros desarrollos de software en este módulo profesional será principalmente **Eclipse**, pero no dejaremos de lado **NetBeans**. Ambos son de código abierto y ofrecen capacidades profesionales. Aunque, no te preocupes, también haremos un recorrido por otros entornos destacables.

Puedes echar un vistazo al siguiente enlace para tratar de aclarar qué entorno es más conveniente, pero debemos incidir en que es una decisión difícil de tomar, dada la gran calidad de ambos IDE's.

<http://codejavu.blogspot.com/2013/10/eclipse-o-netbeans.html>

## 2 ¿Qué son?

Son aplicaciones que ofrecen la posibilidad de llevar a cabo el proceso completo de desarrollo de software a través de un único programa. Podremos realizar las labores de edición, compilación, depuración, detección de errores, corrección y ejecución de programas escritos en Java o en otros lenguajes de programación, bajo un entorno gráfico (no mediante línea de comandos). Junto a las capacidades descritas, cada entorno añade otras que ayudan a realizar el proceso de programación, como por ejemplo: código fuente coloreado, plantillas para diferentes tipos de aplicaciones, creación de proyectos, etc.



Imagen extraída de curso Programación del MECD.

Hay que tener en cuenta que un entorno de desarrollo no es más que una fachada para el proceso de compilación y ejecución de un programa. ¿Qué quiere decir eso? Pues que si tenemos instalado un IDE y no tenemos instalado el compilador, no tenemos nada.

### **Para saber más**

Si deseas conocer algo más sobre lo que son los Entornos Integrados de Desarrollo (IDE) accede al siguiente enlace:

Definición de Entorno Integrado de Desarrollo en Wikipedia

### 3 IDE's actuales.

Existen en el mercado multitud de entornos de desarrollo para el lenguaje Java, los hay de libre distribución, de pago, para principiantes, para profesionales, que consumen más recursos, que son más ligeros, más amigables, más complejos que otros, etc.

Entre los que son gratuitos o de libre distribución tenemos:

- **NetBeans**
- **Eclipse**
- **BlueJ**
- **Jgrasp**
- **Jcreator LE**

Entre los que son propietarios o de pago tenemos:

- **IntelliJ IDEA**
- **Jbuilder**
- **Jcreator**
- **JDeveloper**

#### Debes conocer

Cada uno de los entornos nombrados más arriba posee características que los hacen diferentes unos de otros, pero para tener una idea general de la versatilidad y potencia de cada uno de ellos, accede a la siguiente tabla comparativa:

[https://en.wikipedia.org/wiki/Comparison\\_of\\_integrated\\_development\\_environments#Java](https://en.wikipedia.org/wiki/Comparison_of_integrated_development_environments#Java)

Pero, ¿cuál o cuáles son los más utilizados por la comunidad de programadores Java? El puesto de honor se lo disputan entre **Eclipse**, **NetBeans** y **IntelliJ IDEA**. En los siguientes epígrafes haremos una descripción de NetBeans y Eclipse, para posteriormente desarrollar los puntos claves del entorno Eclipse y del entorno NetBeans.

#### Autoevaluación

**¿Cuál de los siguientes entornos sólo está soportado en la plataforma Windows?**

- ☐ Eclipse.
- ☐ IntelliJ IDEA.
- ☐ Jcreator.