

IDES Y EDITORES DE TEXTO

CODE UD1 – Tema 1

IES Plurilingüe Antón Losada Diéguez

Adrián Fernández González



Tabla de contenido

1. Introducción.....2

 1.2 Editor de textos vs IDE.....2

2. Partes de un IDE.....2

 2.1 Editor de código fuente2

 2.2 Automatizador de compilación3

 2.3 Depurador3

IDEs y editores de texto

1. Introducción

Un contorno de desarrollo o IDE (Integrated Development Environment) es una aplicación diseñada para ayudar a diseñar, codificar, probar, ejecutar y analizar software. Este incluye diversas utilidades en un mismo programa, bajo una misma interfaz.

1.2 Editor de textos vs IDE

Un editor de textos, tal como su nombre indica, permite la edición de archivos de texto de cualquier tipo. Existen múltiples editores avanzados diseñados para la programación, más o menos sofisticados, pero que están pensados para la parte de codificación, pero no así las demás.

Sí es cierto que ciertos lenguajes como los de marcas o aquellos interpretados, al no necesitar de elementos intermedios para su prueba, pueden y suelen programarse en editores de texto más o menos sofisticados. Pero, aquellos lenguajes que necesitan de compiladores o máquinas virtuales, es más rápido y cómodo el uso de un IDE.

A su vez, los editores están bien para pequeños o medianos proyectos, pero a medida que el proyecto aumenta en complejidad y tamaño, se hace necesario el uso de IDEs.

Por otro lado, los IDEs tienen incorporado el editor de textos, pero ofrecen a mayores muchas otras utilidades.

En cambio, los IDEs tienden a ser más pesados y lentos que un editor de textos, por lo que, para pequeñas modificaciones o pequeños proyectos, pueden resultar más útiles los editores.

2. Partes de un IDE

Como ya se ha mencionado antes, un IDE es un conjunto de utilidades empaquetadas en un mismo programa. Estas se pueden dividir en tres grandes bloques, el editor de código fuente, el automatizador de compilación y el depurador.

2.1 Editor de código fuente

Es la parte que nos permite codificar, programar. Incluye diversas utilidades:

- Ayudas visuales como indexación del texto
- Comprobación básica de errores de escritura.
- Colores identificativos de las partes del lenguaje.
- Visor de documentos.
- Autocompletado.
- Guía de buenas prácticas.

-Buscador y reemplazado de textos en múltiples documentos.

-Refactorizador de código.

Entre otras muchas utilidades.

2.2 Automatizador de compilación

Esta parte es la encargada de coger nuestro código y enviarlo al compilador, intérprete o máquina virtual dependiendo del lenguaje.

Luego, recogerá la información de los errores y resultados y nos la mostrará de una forma amigable.

2.3 Depurador

Esta última parte nos ayuda a analizar el código y detectar errores, ya sea mientras lo escribimos o mientras se ejecuta.

La mayoría nos permiten ejecutar el código línea a línea viendo el estado de la memoria en cada momento y los resultados de cada instrucción.

A mayores, algunos nos dan información adicional como uso de recursos o estadísticas.