

Síntesis conceptual

Grado: Administración de sistemas informáticos en red
Asignatura: Programación básica
Unidad: 3. Lenguajes de programación y entornos de desarrollo integrado

Resumen

Existen una multitud de lenguajes de programación, pero entre ellos destaca con diferencia Java, tanto como un lenguaje para trabajar como introductorio para nuevos programadores. Una de las principales características de este lenguaje es su gran versatilidad, que le ha permitido adentrarse en todos los aspectos de la informática, al tiempo que sirve de base a los estudiantes para trabajar con todos estos aspectos con un único lenguaje. Su gran uso lo ha llevado a poseer múltiples versiones de carácter más especializado.

Oracle ofrece entornos de desarrollo para Java, destacando JDK, Java Development Kit, en sus dos tipos de licencias:

- Binary Code License Agreement for the Java SE Platform Products and JavaFX: Licencia de JDK
- GNU General Public License, version 2: Licencia de OpenJDK.

Es necesario saber que los JDK requieren de JRE, Java Runtime Environment, para ejecutarse.

A la hora de escribir programas, los lenguajes de alto nivel proporcionan, en relación con los de bajo nivel, un nivel superior de abstracción sobre la arquitectura física. Utilizar compiladores se hace necesario para la obtención de programas partiendo de ellos. En el mundo del desarrollo algunas tecnologías que, mediante compiladores, traducen a un código intermedio esos programas. Este código intermedio no se puede ejecutar directamente en el sistema operativo, requiriendo un elemento especial para ello, mediante una segunda traducción. Este código intermedio se conoce como bytecode en el mundo de Java.

CIL o MSIL es como se le denominaría al código intermedio en el mundo .NET y la máquina virtual que lo traduciría sería CLR (Common Language Runtime). Para evitar que el código intermedio se traduzca una y otra vez, se incluyó un compilador JIT (Just In Time) en el JRE.

Durante el proceso de creación se emplean archivos con diversas extensiones, como: .java, para el código fuente; .class, código intermedio; .jar, para un archivo comprimido; y otros como .war o .ear.

Recolector de basura, Java, mediante esta tecnología, intenta aligerar el peso de los archivos eliminando elementos "inútiles" que consumen recursos.

Es posible programar sin IDE, aunque esto conllevará diversos inconvenientes:

Deberemos compilar el código intermedio con el comando Javac, *Javac<opciones><ficheros>*.

Podemos crearlo de dos maneras, Como simples contenedores de recursos o ejecutables comprimidos. Por último, para su ejecución podemos emplear JRE con el siguiente comando:
Java [opciones] <clase principal> [argumentos...]

Los entornos de desarrollo integrados o IDE contienen una multitud de herramientas, como depuradoras, editores, compiladores, etc., generalmente basadas en aplicaciones gráficas. Algunos de los IDE que podemos destacar son NetBeans y Eclipse como gratuitas o IntelliJ IDEA y JCreator como versiones de pago.

Conceptos fundamentales

- **Código fuente:** programa en lenguaje de alto nivel, aquel cercano al lenguaje humano y no reconocido por la computadora sin pasar por traducción.
- **Código máquina o código objeto:** código traducido para que pueda ser interpretado por el microprocesador.
- **Enlazador:** llamado *linker*, une el código compilado, los recursos del código fuente y las bibliotecas empleadas.
- **Servlet:** programa de Java que al ejecutarse permite responder peticiones basadas en tablas dinámicas.
- **Shortcut:** Atajo de teclado, conjunto o secuencia de teclas que desencadenan una acción específica.