**¿Qué es Java?**

Java es una plataforma informática de lenguaje de programación creada por Sun Microsystems en 1995. Ha evolucionado desde sus humildes comienzos hasta impulsar una gran parte del mundo digital actual, ya que es una plataforma fiable en la que se crean muchos servicios y aplicaciones. Los nuevos e innovadores productos y servicios digitales diseñados para el futuro también siguen basándose en Java.

Aunque la mayoría de aplicaciones Java modernas combinan el tiempo de ejecución y la aplicación de Java, todavía existen algunas aplicaciones e incluso sitios web que no funcionan sin instalar Java para escritorio. El sitio web Java.com está pensado para consumidores que todavía necesitan Java en sus aplicaciones de escritorio, sobre todo las aplicaciones que tienen como destino Java 8. Los desarrolladores y los usuarios que quieran aprender sobre programación Java, deben consultar el sitio web [dev.java](https://dev.java/) y los usuarios profesionales, [oracle.com/java](https://www.oracle.com/java/) para obtener más información.

JRE incluye Java Virtual Machine (JVM), las clases del núcleo de la plataforma Java y bibliotecas de la plataforma Java de compatibilidad. JRE representa la parte de tiempo de ejecución del software Java, que es todo lo que necesita para ejecutar las aplicaciones de Java WebStart desde un navegador web compatible. Sin embargo, no incorpora herramientas de desarrollo, dado que ya forman parte de Java Development Kit (JDK).

Java Virtual Machine es solo una parte del software de Java que se encarga de ejecutar una aplicación. Java Virtual Machine está incluido en la descarga del software Java, forma parte de JRE y ayuda a ejecutar aplicaciones Java.

¿Cómo obtener Java?

Los desarrolladores encontrarán todos los kits de desarrollo y otras herramientas útiles en <https://www.oracle.com/javadownload/>.

**¿Qué es la JVM?**

La máquina virtual Java (JVM) es un software que permite ejecutar código Java en cualquier plataforma. La JVM es responsable de traducir el código de bytes Java, que es el código que se genera después de compilar un archivo Java, en código nativo que puede ser ejecutado por el hardware de la computadora.

El entorno de ejecución Java (JRE - Java Runtime Environment) es un conjunto de software que incluye la JVM, así como otras bibliotecas y herramientas necesarias para ejecutar aplicaciones Java. El JRE se puede descargar e instalar en cualquier computadora para permitir que se ejecuten aplicaciones Java.

**Estructura de la JVM**

La JVM está compuesta por los siguientes componentes:

* Cargador de clases: Carga los archivos de clases Java en la memoria.
* Verificador: Verifica que los archivos de clases Java sean seguros y compatibles con la JVM.
* Área de memoria: Aloja las variables, objetos y otros datos de las aplicaciones Java.
* Motor de ejecución: Ejecuta el código de bytes Java.
* Bibliotecas de clases: Proporciona funciones y métodos comunes a todas las aplicaciones Java.
* Proceso de ejecución de una aplicación Java

El proceso de ejecución de una aplicación Java es el siguiente:

* El usuario inicia la aplicación Java.
* El cargador de clases carga los archivos de clases Java en la memoria.
* El verificador verifica los archivos de clases Java.
* El área de memoria aloja las variables, objetos y otros datos de la aplicación Java.
* El motor de ejecución ejecuta el código de bytes Java.
* La aplicación Java finaliza.
* Aplicaciones de la JVM

La JVM es una herramienta versátil que se puede utilizar para una variedad de aplicaciones, incluyendo:

* Aplicaciones de escritorio
* Aplicaciones web
* Aplicaciones móviles
* Juegos
* Aplicaciones científicas y de ingeniería
* Ventajas de la JVM

La JVM ofrece una serie de ventajas, incluyendo:

* Portabilidad: El código Java puede ejecutarse en cualquier plataforma que tenga una JVM instalada.
* Seguridad: La JVM incluye mecanismos de seguridad que ayudan a proteger las aplicaciones Java de ataques maliciosos.
* Eficiencia: La JVM puede ejecutar código Java de manera eficiente, lo que hace que las aplicaciones Java sean rápidas y receptivas

**¿Cómo funciona la JVM?**

La JVM funciona de la siguiente manera:

1. El usuario inicia la aplicación Java.
2. El cargador de clases carga los archivos de clases Java en la memoria.
3. El verificador verifica los archivos de clases Java.
4. El área de memoria aloja las variables, objetos y otros datos de la aplicación Java.
5. El motor de ejecución ejecuta el código de bytes Java.
6. La aplicación Java finaliza.
7. Cargador de clases

El cargador de clases es el componente de la JVM que se encarga de cargar los archivos de clases Java en la memoria. Los archivos de clases Java son archivos que contienen el código de bytes Java, que es el código que se genera después de compilar un archivo Java.

El cargador de clases busca los archivos de clases Java en el classpath, que es una lista de directorios donde se pueden encontrar los archivos de clases Java. Si el cargador de clases no encuentra un archivo de clases Java en el classpath, buscará el archivo en el sistema de archivos.

Verificador

El verificador es el componente de la JVM que se encarga de verificar que los archivos de clases Java sean seguros y compatibles con la JVM. El verificador verifica los archivos de clases Java en busca de errores y vulnerabilidades.

Área de memoria

El área de memoria es el componente de la JVM que aloja las variables, objetos y otros datos de las aplicaciones Java. La JVM tiene varios tipos de memoria, cada uno de los cuales tiene una función específica.

Los tipos de memoria de la JVM son los siguientes:

1. Pila: La pila almacena los datos locales de los métodos, como parámetros, variables y resultados.
2. Pila de objetos: La pila de objetos almacena las referencias a los objetos Java.
3. Área de datos: El área de datos almacena los datos globales de la aplicación, como constantes y variables.
4. Área de heap: El área de heap almacena los objetos Java creados por la aplicación.

Motor de ejecución

El motor de ejecución es el componente de la JVM que ejecuta el código de bytes Java. El motor de ejecución traduce el código de bytes Java en código nativo que puede ser ejecutado por el hardware de la computadora.

El motor de ejecución utiliza un conjunto de instrucciones que están optimizadas para ejecutar código Java. Estas instrucciones permiten que el motor de ejecución ejecute código Java de manera eficiente.

Bibliotecas de clases

Las bibliotecas de clases son un conjunto de clases Java que proporcionan funciones y métodos comunes a todas las aplicaciones Java. Las bibliotecas de clases se incluyen en el JRE, que es el entorno de ejecución Java.

Las bibliotecas de clases más importantes de Java son las siguientes:

Biblioteca estándar de Java: La biblioteca estándar de Java proporciona funciones y métodos comunes para la manipulación de datos, entrada/salida, networking, etc.

Biblioteca gráfica de Java: La biblioteca gráfica de Java proporciona funciones y métodos para la creación de interfaces gráficas de usuario.

Biblioteca de acceso a datos de Java: La biblioteca de acceso a datos de Java proporciona funciones y métodos para el acceso a datos de bases de datos.

Ventajas de la JVM

La JVM ofrece una serie de ventajas, incluyendo:

1. Portabilidad: El código Java puede ejecutarse en cualquier plataforma que tenga una JVM instalada.
2. Seguridad: La JVM incluye mecanismos de seguridad que ayudan a proteger las aplicaciones Java de ataques maliciosos.
3. Eficiencia: La JVM puede ejecutar código Java de manera eficiente, lo que hace que las aplicaciones Java sean rápidas y receptivas.

Bibliografía:

https://www.java.com/es/download/help/whatis\_java.html