

INDICE

| | |
|---------------------------------|---|
| Historia de java..... | 2 |
| Curiosidades de Java | 3 |
| Tipos y métodos de java..... | 4 |
| Operadores Aritméticos | 5 |
| Estructuras condicionales | 6 |

Historia de java

Java actualmente puede ser considerado uno de los lenguajes más extendido y usado del mundo, pero esto no siempre fue así.

Para hablar de la historia de java, primero debemos remontarnos a los años 80s, donde C podía considerarse el lenguaje de programación por excelencia. Era un lenguaje versátil, que podía actuar a bajo nivel y resolvían problemas muy complejos. Era la cima de la programación estructurada, para resolver estos complejos algoritmos, se generaban grandes procedimientos con un código muy complicado de mantener a largo plazo. Por ello empezó a surgir como alternativa la programación orientada a objetos, y con ella nació C++.

Java nace en 1991 con el nombre "OAK", posteriormente cambiado por Green por problemas legales, y finalmente con la denominación actual JAVA.

El objetivo de java era crear un lenguaje de programación parecido a C++ en estructura y sintaxis, fuertemente orientado a objetos, pero con una máquina virtual propia. Esto se hizo bajo el principio, de poder ser usado bajo cualquier arquitectura "Write Once, Run Anywhere (escríbelo una vez, ejecútalo en cualquier sitio)".

En 1992 se presenta el proyecto verde, con los prototipos a bajo nivel. Entre 1993 y 1994 se trabaja para poder presentar un prototipo funcional (hotJava) donde se ve todo el potencial que JAVA puede ofrecer.

En 1995, es presentada la versión Alpha de java, y un año después en 1996 es lanzado el primer JDK (JDK 1.0). El desarrollo de java a partir de entonces es imparable, se van presentando nuevos paquetes y librerías hasta la actualidad.

A día de hoy, podemos decir, que Java es uno de los lenguajes más importantes del mundo. Con una comunidad extendida en todos los componentes y más de 4 millones de desarrolladores, existen millones de dispositivos que lo usan. Además, tras el surgimiento de Android, Java se estableció como el lenguaje de programación para móviles más extendido del planeta.

Curiosidades de Java

El nombre de JAVA tiene una gran polémica con respecto a su origen, se discuten varias teorías:

- Una de las teorías más difundidas viene del café, se dice que una cafetería cercana donde los desarrolladores tomaban café se llamaba de la misma forma, de ahí vendría también su logo con una taza de café humeante.
- Otra versión dice que viene del acrónimo: Just Another Vague Acronym ("sólo otro acrónimo ambiguo más").
- Aparte de estas teorías, la versión más plausible es que se eligió su denominación al azar de una lista de posibles nombres.



Tipos y métodos de java

- Tipos primitivos: int, short, long, double, float, boolean, byte, char
- Clases de derivador para tipos primitivos: java.lang.Integer, java.lang.Short, java.lang.Long, java.lang.Double, java.lang.Float, java.lang.Boolean, java.lang.Byte, java.lang.Character
- Tipo de serie: java.lang.String
- Tipos de fecha/hora: java.util.Date, java.util.Calendar, java.sql.Date, java.sql.Time, java.sql.Timestamp
- Tipos numéricos: java.math.BigInteger, java.math.BigDecimal.

Cuando un parámetro de conjunto de reglas es un parámetro XML o un parámetro Java de tipo de Java básico, el parámetro se correlaciona directamente con una columna de enlace del modo siguiente:

- Si el parámetro está correlacionado con una columna en un enlace de entrada, ese parámetro debe ser un parámetro IN o IN_OUT. La columna representa el valor que debe pasarse al parámetro cuando se invoca el conjunto de reglas.
- Si el parámetro está correlacionado con una columna en un enlace de salida, ese parámetro debe ser un parámetro OUT o IN_OUT. La columna representa el valor devuelto por el conjunto de reglas para el parámetro, cuando se invoca el conjunto de reglas.

Cuando el parámetro de conjunto de reglas no se correlaciona directamente con una columna de enlace, las construcciones de conjunto de reglas que se correlacionan con las columnas de enlace son los valores de retorno y argumentos de métodos básicos de la clase Java asociada con el parámetro de conjunto de reglas.

Operadores Aritméticos

Un operador lleva a cabo operaciones sobre uno (operador *unario*), dos (operador *binario*) o tres (operador *ternario*) datos u *operandos* de tipo primitivo devolviendo un valor determinado también de un tipo primitivo. El tipo de valor devuelto tras la evaluación depende del operador y del tipo de los operandos. Por ejemplo, los operadores *aritméticos* trabajan con operandos numéricos, llevan a cabo operaciones aritméticas básicas y devuelven el valor numérico correspondiente. Los operadores se pueden clasificar en distintos grupos según se muestra en los siguientes apartados.

El operador asignación `=`, es un operador binario que asigna el valor del término de la derecha al operando de la izquierda. El operando de la izquierda suele ser el identificador de una variable. El término de la derecha es, en general, una expresión de un tipo de dato compatible; en particular puede ser una constante u otra variable. Como caso particular, y a diferencia de los demás operadores, este operador no se evalúa devolviendo un determinado valor.

| Operador | Descripción | Ejemplo de expresión | Resultado del ejemplo |
|----------|---------------------|----------------------|-----------------------|
| = | Operador asignación | n = 4 | n vale 4 |

| Operador | Descripción | Ejemplo de expresión | Resultado del ejemplo |
|----------|------------------------------------|----------------------|-----------------------|
| - | operador unario de cambio de signo | -4 | -4 |
| + | Suma | 2.5 + 7.1 | 9.6 |
| - | Resta | 235.6 - 103.5 | 132.1 |
| * | Producto | 1.2 * 1.1 | 1.32 |
| / | División (tanto entera como real) | 0.050 / 0.2 7 / 2 | 0.25 3 |
| % | Resto de la división entera | 20 % 7 | 6 |

Estructuras condicionales

Los condicionales en Java, son una estructura de control esencial al momento de programar y aprender a programar. La mayoría de los lenguajes de programación utilizados actualmente (por no decir que todos), nos permiten hacer uso de estas estructuras para definir ciertas acciones, condiciones específicas en nuestros algoritmos y programas.

Un condicional, permite establecer una serie de reglas al interior de nuestro programa, que nos ayudan a determinar qué acciones llevar a cabo dadas ciertas circunstancias.

Los condicionales aumentan la "expresividad" de un programa en Java (y cualquier otro lenguaje de programación). Es decir, nos permiten considerar diferentes situaciones con antelación, evitando o permitiendo sortear diferentes tipos de situaciones que son del interés de nuestra aplicación.

Ahora bien, existen diferentes tipos de condicionales, cada uno tiene una utilidad y funcionalidad diferente. Cada uno, considera diferentes situaciones que se pueden llegar a presentar durante la ejecución de un algoritmo o programa en Java.

Depende entonces del conocimiento que tengamos acerca de cada uno de los condicionales saber determinar correctamente cuando es necesario implementar uno u otro. Tenemos a nuestra disposición los siguientes tipos de condicionales en Java:

- Condicional If en Java
- Condicional if-else en Java
- Condicional Switch en Java

Como mencioné anteriormente, cada uno de estos condicionales tiene ciertas características que lo hacen útil para algunos casos específicos.