



Gobierno del
CHACO

Ministerio
de la Producción y el Desarrollo
Económico Sostenible



INFORMATARIO



INTRODUCCIÓN A JAVA



Temas de hoy

1**INTRODUCCION****2****CARACTERÍSTICAS**

Erase una vez.. A mediados de los 90s



Lenguaje mágico porque

- ▶ Los programas desarrollados eran portables
- ▶ Eran ejecutados por internet

Muy popular y utilizado

- ▶ Muchas personas comenzaron a aprender Java
- ▶ Lo usaron para crear todo tipo de cosas, desde juegos hasta aplicaciones empresariales.

JAVA - INTRODUCCION

Java todavía está vivo y en uso hoy en día

Muchas aplicaciones que usas todos los días, como tu navegador web y aplicaciones de teléfonos móviles, están escritas en Java

Stone Chad programador en Java le habla a los simples mortales



imgflip.com

JAVA - ¿Por qué fue exitoso?

Hombre

- ▶ Muy bueno generando patrones, asociando y con la recursividad.
- ▶ Son malos cuantitativamente hablando, errores por cansancio, desmotivación y muy lentos.

Máquina

- ▶ Son buenas, no tienen errores y van a toda velocidad
- ▶ Son tontas

Hemos domesticado y creado un animal → La máquina

JAVA - CARACTERÍSTICAS

Todo lenguaje tiene:

- ▶ Lexicografía :
 - ▷ Palabras reservadas
 - ▷ Símbolos
 - ▷ Identificadores
- ▶ Sintaxis
- ▶ Semántica (Gran identificador):
 - ▷ Asignación, bucles o no
 - ▷ Funciones o no, clases
 - ▷ Objetos o no

JAVA - CARACTERÍSTICAS

Todo lenguaje clasificación por nivel:

- ▶ Bajo nivel. Enfocados en la máquina:
 - ▷ Ensamblador
 - ▷ Binario

Están acoplados a la máquina.

JAVA - CARACTERÍSTICAS

Todo lenguaje clasificación por nivel:

- ▶ Medio nivel. Enfocados al desarrollador y máquina :
 - ▷ C
 - ▷ C++

A veces acoplados a la máquina depende del diseño del hardware.

JAVA - CARACTERÍSTICAS

Todo lenguaje clasificación por nivel:

- ▶ Alto nivel. Enfocados solo al desarrollador :
 - ▷ JAVA
 - ▷ JAVASCRIPT
 - ▷ PYTHON
 - ▷ PHP
 - ▷ ...

100% portables y legibles

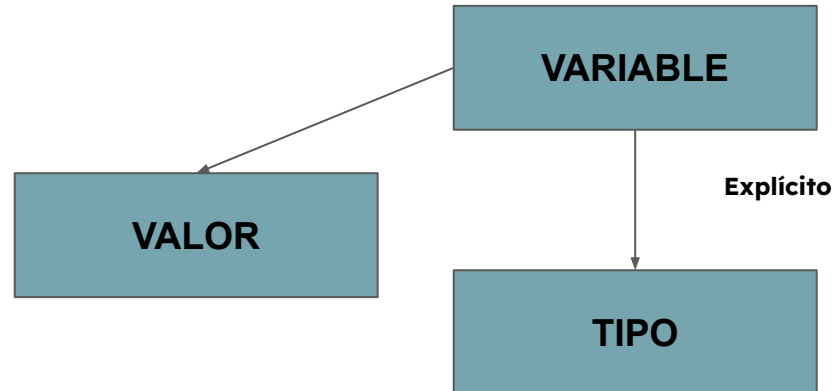
1

Lenguajes tipados:

Todos los lenguajes son tipados, excepto
los de bajo nivel.

JAVA - CARACTERÍSTICAS

- ▶ Se dice explícitamente el tipo de la variable
- ▶ En todo momento la variable tendrá un cierto valor de un tipo

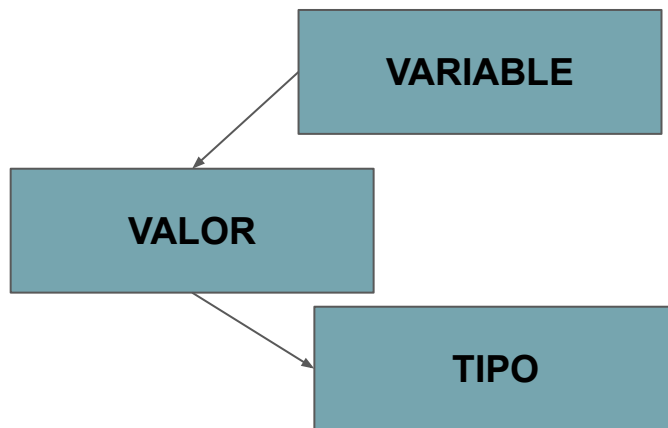


2

Lenguajes débilmente tipado:

La variable está abierta a tener un valor de cualquier tipo

JAVA - CARACTERÍSTICAS



3

Lenguaje inferencia de tipos:

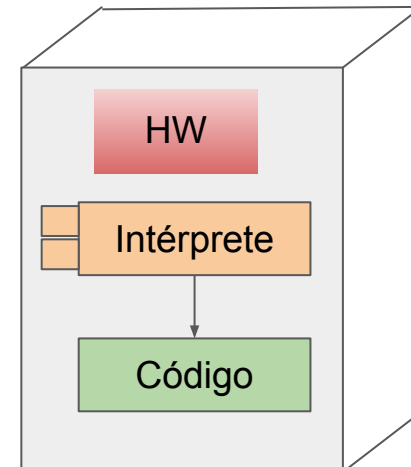
Visto en el paradigma funcional, donde implícitamente se indica el tipo, reduciendo la verbosidad del lenguaje.

JAVA - CARACTERÍSTICAS

Clasificación por ejecución:

- Interpretado : Según lee el código (En ejecución) ejecuta el programa. Cada vez que se ejecuta lo lee por ello es más lento . **Ejemplo : Js, Typescript,**

TIEMPO DE DESARROLLO
Bajo
TIEMPO DE EJECUCIÓN
Bajo
NIVEL DE PORTABILIDAD
ALTA

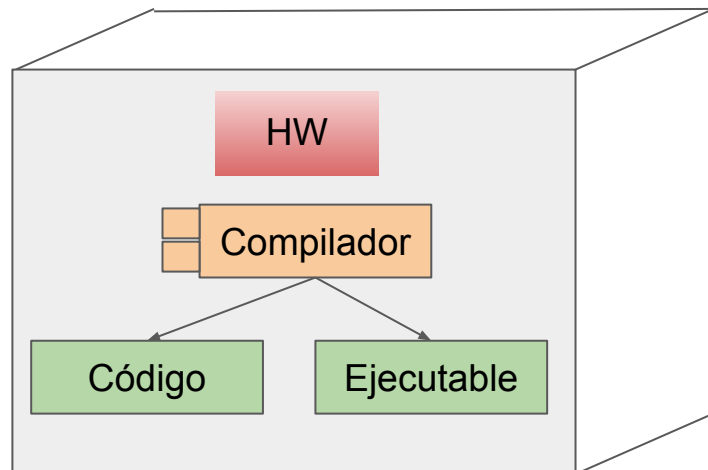


JAVA - CARACTERÍSTICAS

Clasificación por ejecución:

- **Compilado** : Se escribe código, un compilador genera el fichero para la máquina. El compilado genera un ejecutable para un HW y SO específico. Se lee una vez el código, genera una vez el programa y ya no lo vuelvo a hacer. Ejemplo : C/C++

TIEMPO DE DESARROLLO
Bajo
TIEMPO DE EJECUCIÓN
Alto
NIVEL DE PORTABILIDAD
Bajo

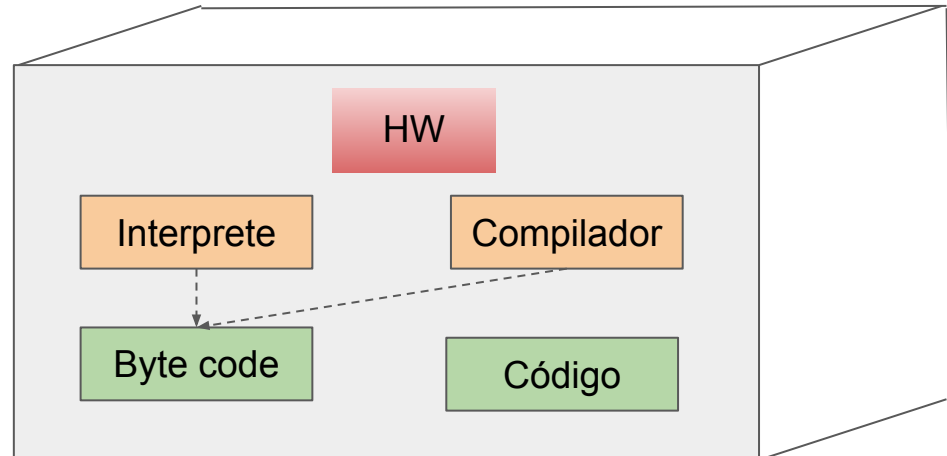


JAVA - CARACTERÍSTICAS

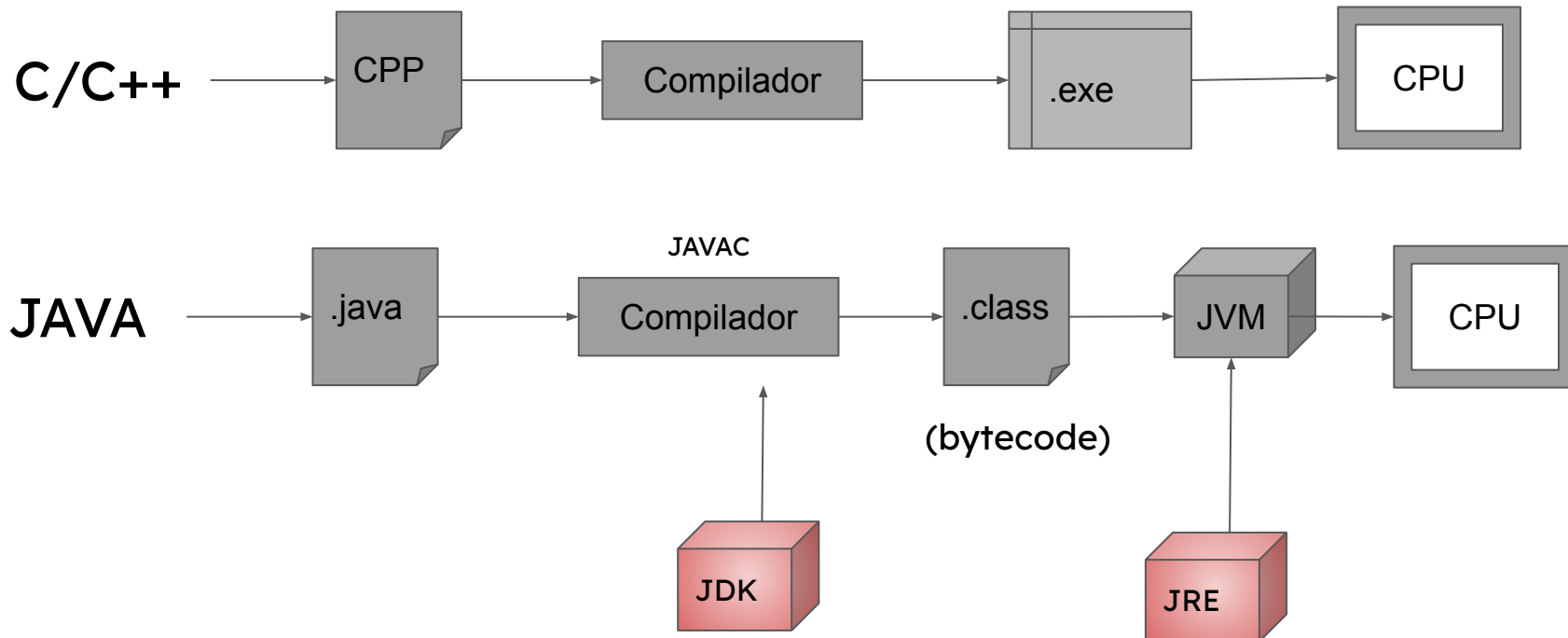
Clasificación por ejecución:

- **Híbrido** : El código lo toma un compilador, genera el código para la máquina virtual de java (JVM). Se compila el código en bytecode y este es interpretado generando el lenguaje de máquina dependiendo de quien lo ejecute. Ejemplo : Java

TIEMPO DE DESARROLLO
Medio
TIEMPO DE EJECUCIÓN
Medio
NIVEL DE PORTABILIDAD
Alta



JAVA - CARACTERÍSTICAS



JAVA - Tipos

JDK

`javac, jar, debugging tools,
javap`

JRE

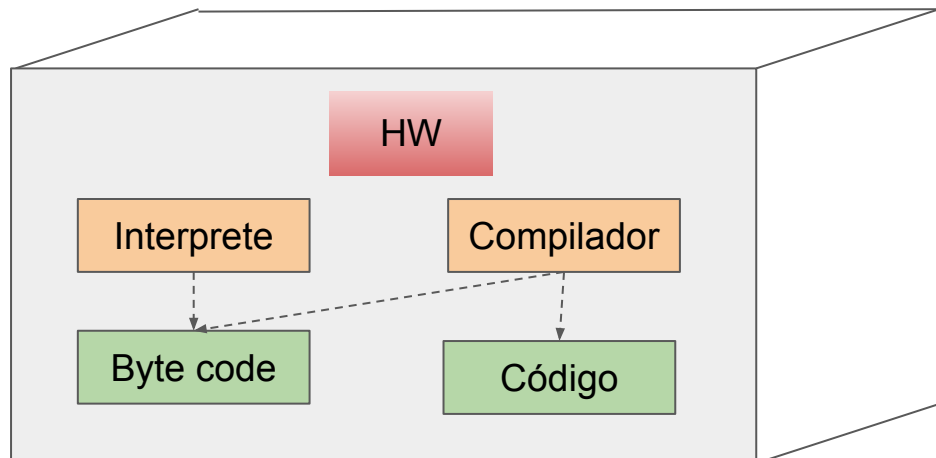
`java, javaw, libraries,
rt.jar`

JVM

Just In Time
Compiler (JIT)

JAVA - CARACTERÍSTICAS

Para cada archivo .java (Clase) se genera un archivo .class generado por el JVM. El .Class tiene bytecode utilizado por el javac (Compilador) que es el que genera el programa (.java)



JAVA - CARACTERÍSTICAS

Paradigma imperativo = Modelo de turing

Basado en asignaciones de valores a variables para cambios de estado.

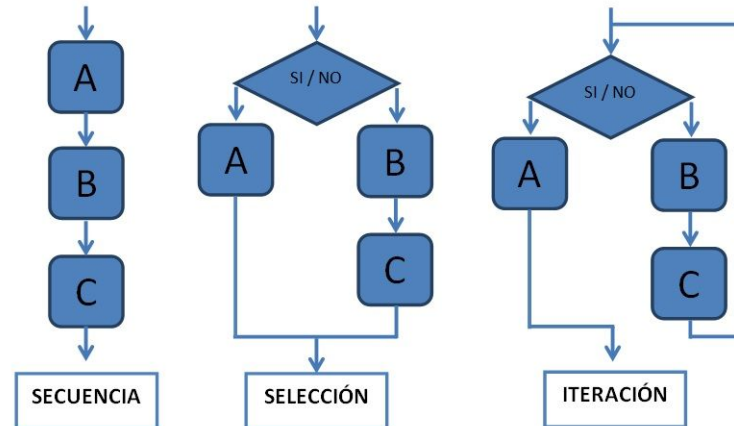
Al igual que la máquina de Turing leemos y escribimos en una variable. Estaba basada en órdenes a una máquina.

Hay variables y estados.

JAVA - CARACTERÍSTICAS

Paradigma estructurado

En esta forma de programación, las tareas se dividían en pequeñas partes que eran más fáciles de entender. Era como dividir la lista de tareas en secciones más pequeñas y fáciles de manejar. Esto hacía que la programación fuera más fácil de mantener y mejorar.



JAVA - CARACTERÍSTICAS

Paradigma orientado a procesos

Tiende más a operaciones y no a datos, las funciones ya no son las de lambda, estas si permiten cambiar de estado, devolver valores, etc.

Paradigma orientado a datos (SQL)

Tiende más a datos y no a operaciones

Paradigma orientado a objetos

Equilibrio entre datos y operaciones conformando una clase de objetos
(unidades) instanciable.

JAVA - CARACTERÍSTICAS

Paradigma orientado a objetos

En este paradigma, las tareas se dividían en pequeñas partes llamadas objetos.

Cada objeto tenía su propia función y trabajaba junto con otros objetos para realizar una tarea más grande.

JAVA - CARACTERÍSTICAS

Java es entonces

- ▷ Secuencia/Concurrente
- ▷ Imperativo
- ▷ Orientado a objetos

No existe el multiparadigma! Hoy en día los lenguajes y frameworks tienen o usan mecanismos de estos paradigmas por ejemplo el funcional, pero no por ello hace a estos lenguajes de ese tipo de paradigma.

JAVA - Logros

- ▶ Simplificó C++
- ▶ Seguro con sistema de tipos
- ▶ Independiente de la plataforma
- ▶ Robusto
- ▶ Portable
- ▶ Dinámico
- ▶ Interpretado y también compilado
- ▶ Alto rendimiento

JAVA - Tipos

JAVA SE (Standard Edition)

- Utilizadas en ordenadores

JAVA SE (Standard Edition)

- Usadas en servidores web
- Necesita JAVA SE para ejecutarse
- Define un conjunto de librerías para el desarrollo de aplicaciones web

JAVA MICRO EDITION

- Usadas en dispositivos móviles, software de electrodomésticos, entre otras.



Gobierno del
CHACO

Ministerio
de la Producción y el Desarrollo
Económico Sostenible



INFORMATARIO



FIN DE LA PRESENTACIÓN

