

# Sesión 1: Introducción a la programación



# Índice

- ¿Qué es la programación?
- Programación Orientada a Objetos
- El lenguaje Java
- Historia, origen y creación de Java
- El entorno Java
- Entornos integrados de desarrollo
- Instalación del IDE
- Primer programa en Java

# ¿Qué es la programación?

- El concepto programación consiste en definir una serie de directrices con las que se obtiene un resultado que corresponde a la resolución de un problema.
- A través de los lenguajes de programación, podremos definir las directrices que indicarán al ordenador, de un modo comprensible para su procesador, cómo llegar a la resolución de un problema expuesto.
- Según su complejidad, podemos distinguir entre lenguajes de alto nivel (C++ o Java) y de bajo nivel (assembler).
- Los de alto nivel, permiten desarrollar programas simplificando el modo en que dirigimos las instrucciones al ordenador, mientras que los de bajo nivel se basan en instrucciones dirigidas directamente al hardware interno del ordenador.

# ¿Qué es un programa informático?

Un programa informático es una secuencia lógica de instrucciones escritas en un lenguaje de programación determinado que, al ser interpretadas por un ordenador, dan solución a un problema mediante la manipulación de un conjunto de datos.

- Una vez definidas, dichas instrucciones dan lugar a lo que denominamos *código fuente*.
- Todo lenguaje de programación está sujeto a una serie de normas sintácticas y estructurales propias.

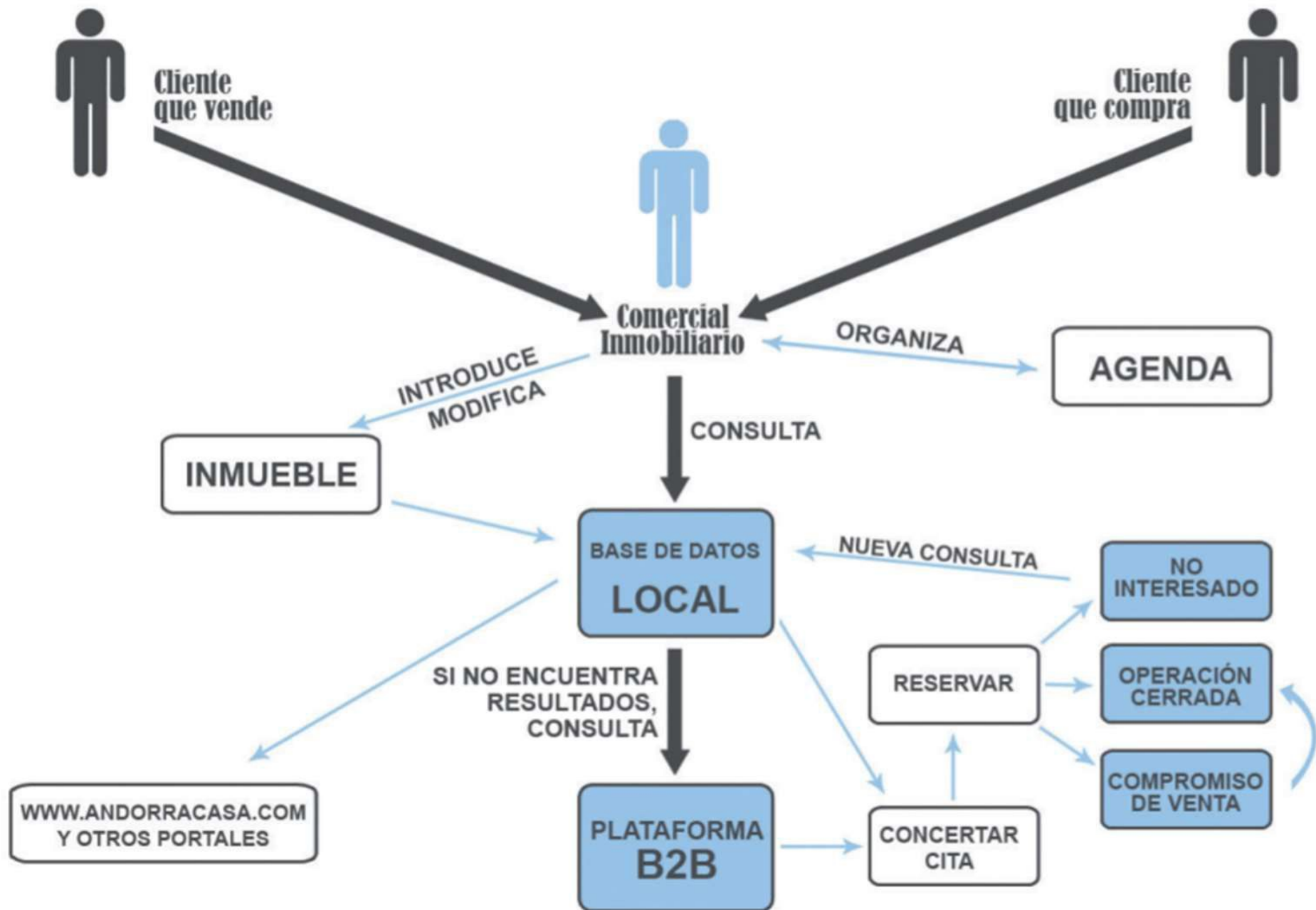
# Estructura y bloques fundamentales

**Bloque de declaraciones:** Conjunto de datos proporcionados al programa informático destinados a ser manipulados a través de las instrucciones definidas en él. Pueden ser de diferente origen y naturaleza.

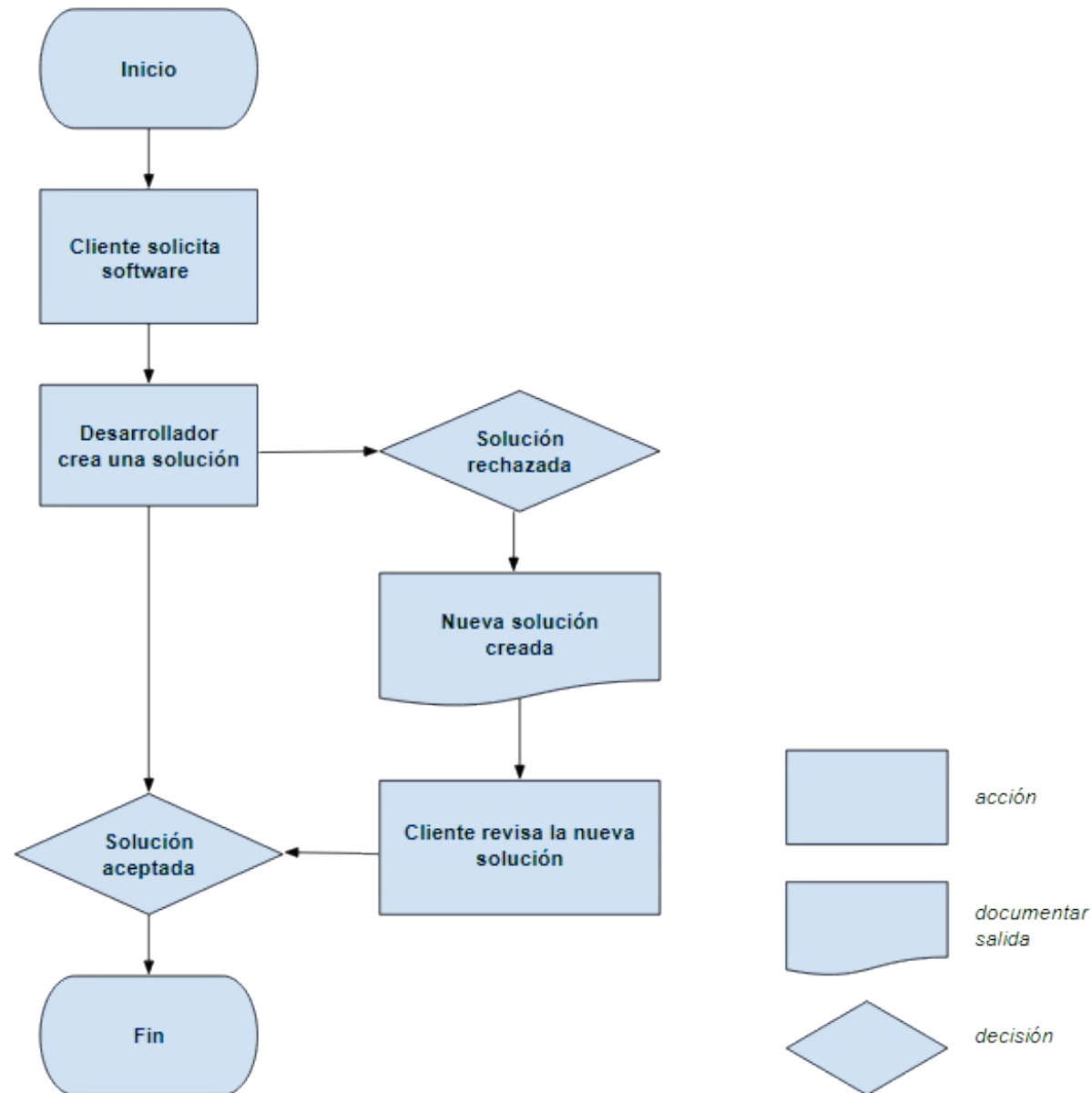
**Bloque de instrucciones:** Conjunto de acciones que usará el programa para manipular los datos proporcionados y obtener los resultados necesarios para la resolución del problema propuesto. Debe realizar las siguientes tareas:

- Entrada: Recogida de los datos externos al programa necesarios para resolver el problema.
- Transformación: Conversión de los datos para conseguir el resultado.
- Salida: Enviar a su destinatario la solución al problema, de modo que lo pueda entender.

# Diseño del programa



# Flujo del programa



# Programación orientada a objetos

- La Programación Orientada a Objetos (POO) es un **paradigma de programación** (o técnica de programación) que utiliza objetos en el diseño del programa.
- La POO es la base de Java. Todos los programas en Java están orientados a objetos.
- El lema de la POO: “Divide y vencerás”
- Otros lenguajes orientados a objetos: C++, C# ..



# Programación orientada a objetos

La Programación Orientada a Objetos es un sistema o **conjunto de reglas** que nos ayudan a descomponer la aplicación en **objetos**.

Beneficios:

- **Comprensión:** El código es fácil de comprender
- **Modularidad:** Código mejor organizado
- **Seguridad:** Menos probabilidad de cometer errores
- **Reusabilidad:** Los objetos son reutilizables

# Programación orientada a objetos

## Clases y Objetos

Simil de un molde y un flan, o una receta de una tarta y la tarta en sí.

Las clases son los moldes con los cuales se crean los objetos. A la acción de crear un objeto se le llama ***Instanciar***. Pueden existir muchos objetos de la misma clase, pero ella es única. Los objetos son siempre instancias de alguna clase, no de varias.

La vida de los objetos: durante la ejecución de un programa

- Se crean objetos que pueden interactuar con otros (*mensajes*)
- Se eliminan los objetos no necesarios

# Programación orientada a objetos

## *Los 4 principios de la programación orientada a objetos:*

### **1. Abstracción:**

Es el proceso por el cual definimos las características más importantes de un objeto, sin preocuparnos de cómo se escribirán en el código del programa.

El término abstracción consiste en ver a algo como un todo sin saber cómo está formado internamente.

En Java, el ejemplo de abstracción es una *clase*: Se define como un todo con el objetivo de que el usuario cree un objeto de esa clase y utilice sus métodos, sin preocuparse en cómo está implementada o diseñada.

# Programación orientada a objetos

## 2. Encapsulado:

También llamada "ocultamiento de la información".

La encapsulación o encapsulamiento es el mecanismo básico para ocultar la información interna de un objeto a los demás objetos de la aplicación.

Con la encapsulación, un objeto puede ocultar la información que contiene al mundo exterior, o bien restringir el acceso a la misma para evitar ser manipulado de forma inadecuada.

En Java, el encapsulado es la *clase*, y un objeto sería la *instancia de una clase*.

# Programación orientada a objetos

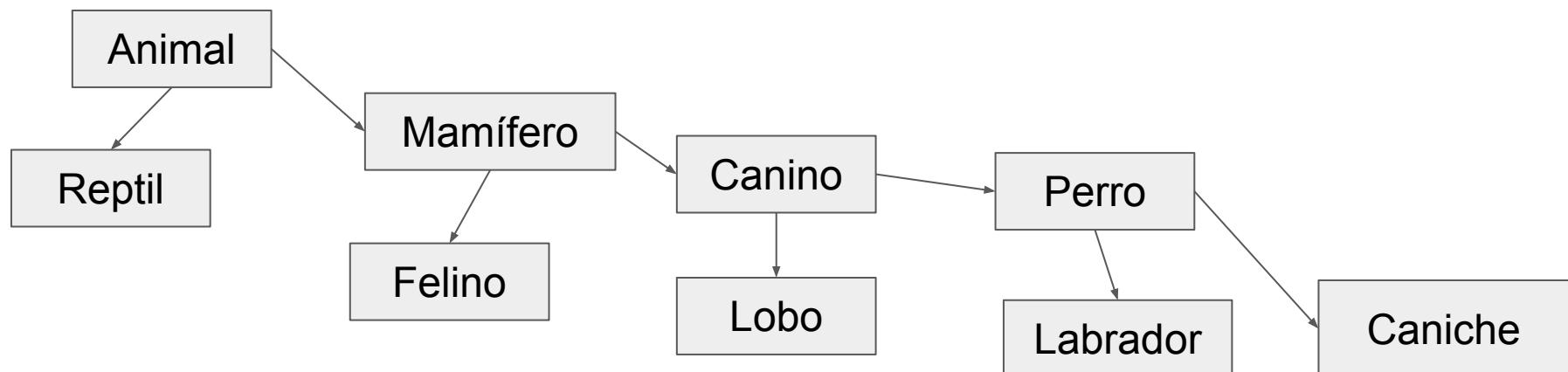
## 3. Herencia:

Es el proceso por el cual una clase u objeto adquiere las propiedades de otro.

Permite definir relaciones de jerarquías entre clases. Es decir, permite crear una clase nueva en términos de otra clase ya existente, obteniendo sus características.

Sin la jerarquía, cada objeto necesitaría definir todas sus características.

*Clase padre o superclase:* La clase superior - *Clase hija o subclase:* la clase que hereda



# Programación orientada a objetos

## 4. Polimorfismo:

Este término está relacionado con la herencia. Es un término un poco complejo de entender, que a veces sólo se lo entiende con la práctica.

Esta propiedad indica la capacidad de que varias clases creadas a partir de una antecesora realicen una misma acción de forma diferente.

En POO, un método con el mismo nombre se puede comportar diferente según el objeto o la información recibida.

Un ejemplo podría ser un teléfono. La función “hacer llamada” se comporta de forma diferente para cada número marcado. En este caso, el número sería la información recibida que recibe el método “hacer llamada.”

# El lenguaje Java

## ¿Qué y cómo es Java?

Java es un lenguaje sencillo de aprender, orientado a objetos, y que permite la utilización de gran cantidad de bibliotecas ya definidas, evitando reescribir código que ya existe.

Java está relacionado con C++, que es descendiente directo de C. Se podría decir que C++ estableció el entorno necesario para Java.

Como otros lenguajes de programación que le precedieron, Java es una mezcla de los mejores elementos de su herencia, combinándolos con conceptos innovadores.

# El lenguaje Java

## Popularidad

Existen una serie de sitios en internet que miden la popularidad de un lenguaje.

- <https://www.tiobe.com/tiobe-index/>
- <https://pypl.github.io/PYPL.html>
- <https://survey.stackoverflow.co/2023/>

Nota: estos índices no indican cuál lenguaje es el mejor..



# Java

## Un poco de historia

- Antes de C, se usaban lenguajes ensamblador como FORTRAN, COBOL o BASIC.
- A comienzos de los 70, tuvo lugar la revolución informática y, por tanto, una demanda de software que superó la capacidad de los programadores para producirlo.
- Consecuencia: nace el lenguaje C. Lenguaje de programación potente, eficiente y estructurado, y relativamente fácil de aprender.
- A finales de los 70, C se convirtió en el lenguaje de programación dominante. Pero la complejidad de los sistemas hacía que se necesitase algo más.
- En 1979 nace C++, inicialmente llamado *C con clases*, cambió su nombre en 1983. Es una extensión de C donde se añaden las características orientadas a objetos.

# Java

## Todo dispuesto para Java

- A finales de los 80 dominaba la POO de C++. Parecía que se había encontrado el lenguaje perfecto.
- En 1991 fue concebido Java. Se tardó 18 meses en desarrollar su primera versión. En principio se llamaba Oak, pero fue renombrado como Java en 1995.
- Su impulso inicial no fue internet, sino la necesidad de un lenguaje de programación independiente de su plataforma.
- Durante el desarrollo de Java, se desarrollaba a su vez la World Wide Web, factor importante para que Java se implementara también para la Web.
- Fue la creación de internet quién finalmente condujo a Java hacia el éxito.

# Java

## Características

- Programación Orientada a Objetos
- Portabilidad: se puede ejecutar en cualquier máquina, multiplataforma
- Bibliotecas de Java: permite realizar cualquier tipo de aplicación
- Seguridad: Java no accede a zonas delicadas de memoria o del sistema
- Bytecode: código máquina al que se compila el código fuente.
- Simple: Curva de aprendizaje muy rápida
- Robusto: Comprobaciones de código en compilación y ejecución
- Multihilo: permite que los programas hagan varias cosas simultáneamente.

# Entorno Java

## JDK (Java Development Kit)

Development Tools

-----  
Herramientas de  
desarrollo

## JRE (Java Runtime Environment)

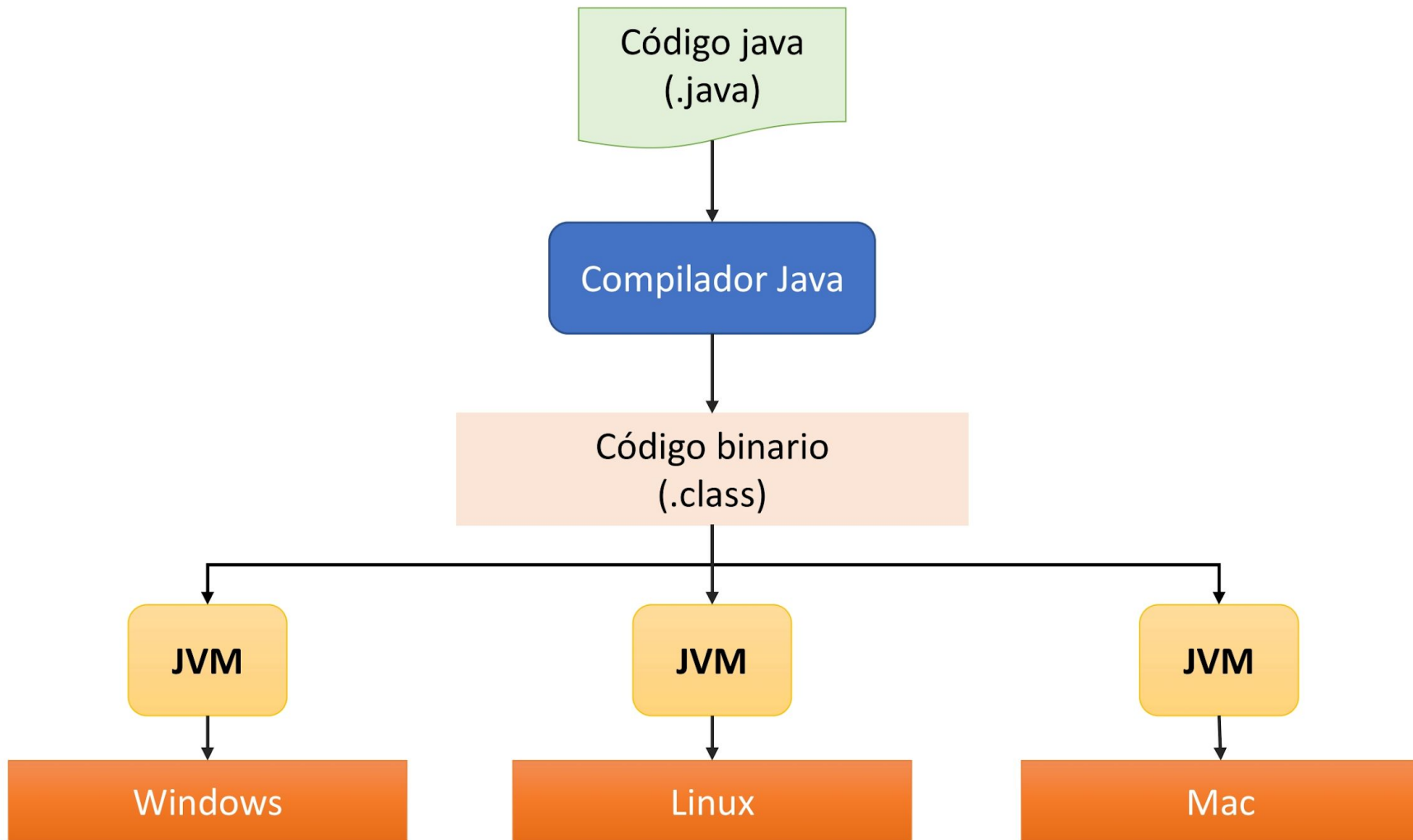
Java class libraries

-----  
Librerías de Java



**JVM**

# Entorno Java



# Entorno Java



descargar jdk



Todo

Videos

Imágenes

Noticias

Libros

⋮ Más

Herramientas

17

Android

15

8

Para NetBeans

64 bits

11

Aproximadamente 459.000 resultados (0,24 segundos)



Oracle

<https://www.oracle.com> › java · [Traducir esta página](#) ⋮

## Java Downloads | Oracle España

Download the **Java** including the latest version 17 LTS on the **Java SE** Platform. These downloads can be used for any purpose, at no cost, under the **Java SE** ...

[Java Development Kit 8...](#) · [Java Archive](#) · [Tools and Resources](#)

# Entorno Java

## Java 22, Java 21, and Java 17 available now

JDK 21 is the latest long-term support release of Java SE Platform.

Learn about Java SE Subscription

JDK 22    JDK 21    JDK 17    GraalVM for JDK 22    GraalVM for JDK 21    GraalVM for JDK 17

### JDK Development Kit 21.0.2 downloads

JDK 21 binaries are free to use in production and free to redistribute, at no cost, under the [Oracle No-Fee Terms and Conditions](#) (NFTC).

JDK 21 will receive updates under the NFTC, until September 2026, a year after the release of the next LTS. Subsequent JDK 21 updates will be licensed under the [Java SE OTN License](#) (OTN) and production use beyond the [limited free grants](#) of the OTN license will [require a fee](#).

Linux    macOS    **Windows**

Product/file description	File size	Download
x64 Compressed Archive	185.52 MB	<a href="https://download.oracle.com/java/21/latest/jdk-21_windows-x64_bin.zip">https://download.oracle.com/java/21/latest/jdk-21_windows-x64_bin.zip</a> (sha256)
x64 Installer	163.91 MB	<a href="https://download.oracle.com/java/21/latest/jdk-21_windows-x64_bin.exe">https://download.oracle.com/java/21/latest/jdk-21_windows-x64_bin.exe</a> (sha256)

# Java: Un primer programa de ejemplo

```
/*
```

```
Este es un programa de prueba en Java
```

```
Este archivo se llama "Ejemplo.java"
```

```
*/
```

```
class Ejemplo {
```

```
    //El programa comienza con una llamada a main().
```

```
    public static void main(String args[]) {
```

```
        System.out.println("Este es un programa de prueba en  
Java");
```

```
    }
```

```
}
```



```
public class HolaMundo {  
    public static void main(String[] args) {  
        System.out.println("Hola Mundo");  
    }  
}
```

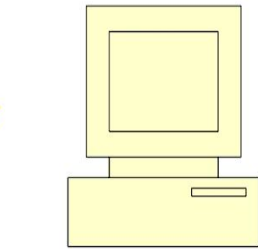
HolaMundo.java

**javac**

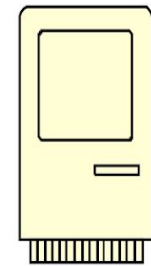
Bytecodes

HolaMundo.class

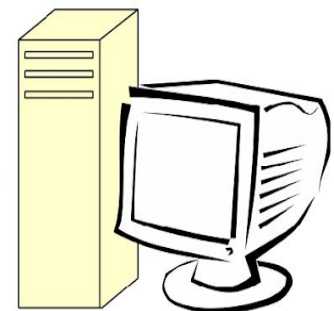
**java**



Windows



MacOS



Solaris

# Entornos integrados de desarrollo (IDE's)

Es un programa informático que ofrece un conjunto de herramientas y recursos necesarios para el desarrollo de aplicaciones.

Facilita el desarrollo de aplicaciones empaquetando los recursos en un marco de trabajo estable y visual. En general los IDE están formados por los siguientes bloques:

- **Editor de código.** Ofrece ayudas visuales, como el formato textual para resaltar los aspectos relevantes del código.
- **Compilador o intérprete.** Herramienta destinada a interpretar el código escrito y evaluar la sintaxis del código generado.
- **Depurador.** Detecta y solventa los errores que pueden surgir en la escritura del código. Permite, por ejemplo, ejecutar línea a línea, ver el valor de algunos datos, etc.
- **Constructor de interfaz gráfica.** Actúa como un conjunto de opciones visuales que permiten la comunicación entre el usuario y el ordenador a la hora de escribir e interpretar el código.

# Entornos integrados de desarrollo (IDE's)

Desarrollado originalmente por IBM en 2004, ahora por Eclipse Foundation.

- Herramienta de código abierto.
- Autocompletado de código.
- Depuración.
- Integración con herramientas de control de versiones.
- Soporte para desarrollo con Android, pruebas unitarias, refactorización..
- Una gran variedad de plugins impulsada por su gran comunidad.



*Otros IDE's populares para Java: Netbeans , IntelliJ IDEA , Visual Studio Code..*

# Java: Un primer programa de ejemplo

```
/*
```

```
Este es un programa de prueba en Java
```

```
Este archivo se llama "Ejemplo.java"
```

```
*/
```

```
class Ejemplo {
```

```
    //El programa comienza con una llamada a main().
```

```
    public static void main(String args[]) {
```

```
        System.out.println("Este es un programa de prueba en  
Java");
```

```
    }
```

```
}
```