



# Unidad 1 – Desarrollo de software

## Entornos de desarrollo – 1º DAM

# Software

- ¿Qué es el software?

El software es la parte de un sistema informático que no podemos tocar.

Sin el software, los ordenadores serían inútiles. Por ejemplo, sin un navegador de Internet (el programa Explorer, Firefox o Chrome), no sería posible navegar por Internet, y sin un sistema operativo, el navegador no podría funcionar.

Las aplicaciones y las App en los dispositivos móviles también son software.

# Clasificación del SW

- Según su función:
  - » **De sistema:** SO, Firmware / Drivers...
  - » **De aplicación:** Office, Firefox...
  - » **De programación:** Eclipse, JDK...
  - » **Malware:** Virus, troyanos, gusanos...

# Clasificación del SW



# Clasificación del SW

- Según el método de distribución:
  - » **Freeware:** Antivirus Avast
  - » **Shareware:** Demo WinRAR
  - » **Código abierto:** OpenOffice
  - » **Libre:** free software no significa gratis
  - » **Privativo:** cuyo código fuente no está disponible
  - » **Comercial:** que se vende

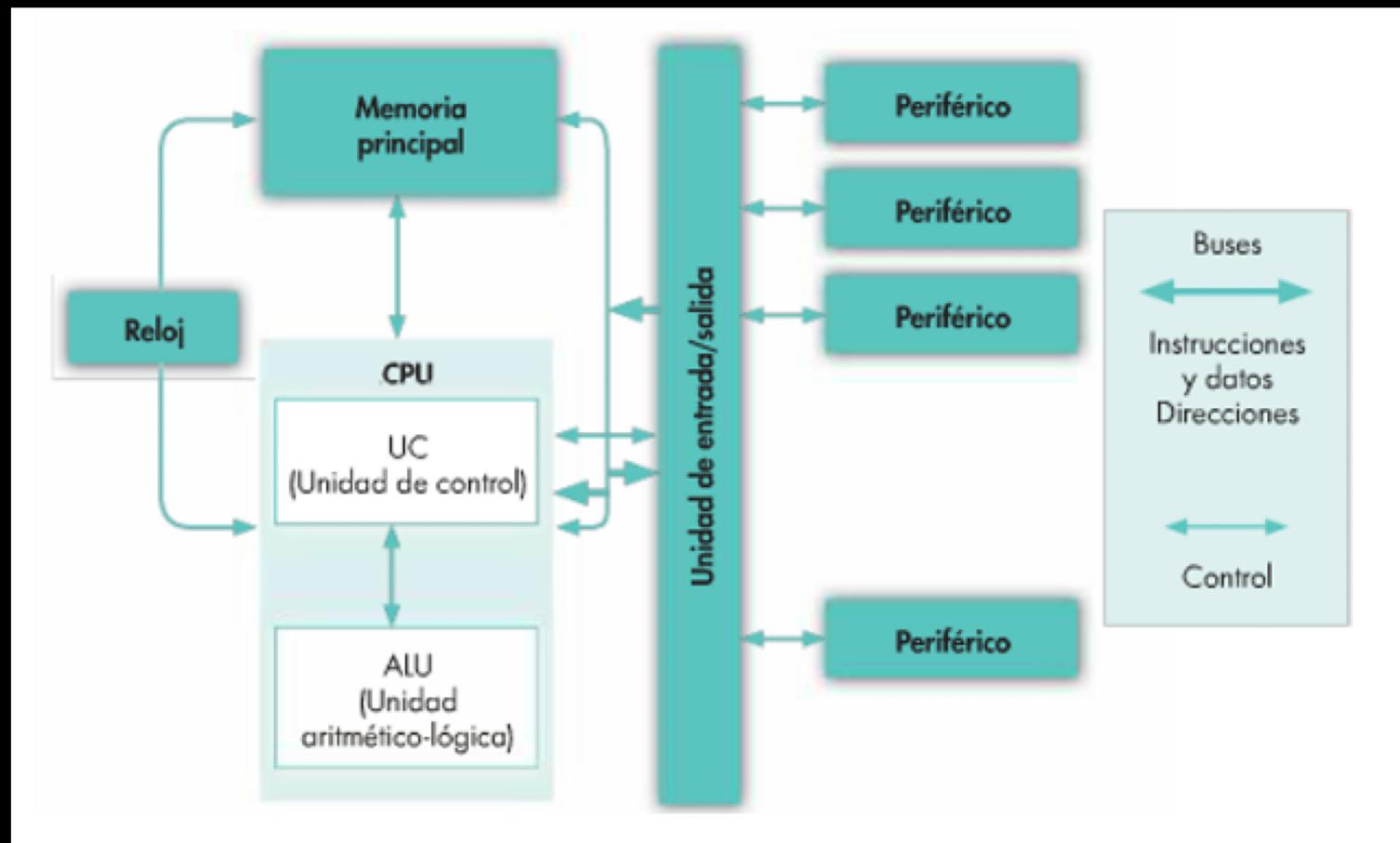
# Hardware

- » ¿Qué es el hardware?

En contraposición, el hardware es la parte de un sistema informático que podemos tocar:

- » Unidad central de proceso.
- » Tarjetas gráficas o wifi
- » Memorias.
- » Unidades de entrada y salida (periféricos).

# Hardware



# Programa informático

- ¿Un programa informático es hardware o software?
- Un **programa informático** es una secuencia de instrucciones, escritas para realizar una tarea específica en un ordenador.

# Programa informático

- ¿Un programa informático es hardware o software?
- Un **programa informático** es una secuencia de instrucciones, escritas para realizar una tarea específica en un ordenador.

# Programación

- Proceso por el cual se escribe, se prueba, se depura, se compila y se mantiene el código fuente de un programa informático.



# Herramientas de programación

- Editores de texto
- Compiladores
- Ensambladores
- Enlazadores (linkers)
- Depuradores (debuggers)
- IDEs (Interfaz de Desarrollo Integrada)

# Lenguaje de programación

- Idioma artificial, relativamente próximo al lenguaje natural, diseñado para expresar instrucciones que puedan ser ejecutadas por un ordenador.
- Permite especificar de manera precisa qué datos debe operar un ordenador, cómo debe almacenarlos o transmitirlos y qué acciones debe tomar bajo una variada gama de circunstancias.

# Clasificación de los lenguajes

- Según su forma de ejecución:
  - Lenguajes compilados
  - Lenguajes interpretados

# Clasificación de los lenguajes

- Según su nivel de abstracción:
  - Lenguajes de bajo nivel
  - Lenguajes de alto nivel

# Clasificación de los lenguajes

- Según su propósito:
  - Lenguajes de propósito general
  - Lenguajes de propósito específico
  - Lenguajes de programación de sistemas
  - Lenguajes de script

# Paradigmas de programación

- Lenguajes estructurados (programación imperativa)
- Lenguajes orientados a objetos (programación orientada a objetos)
- Lenguajes declarativos (programación funcional)

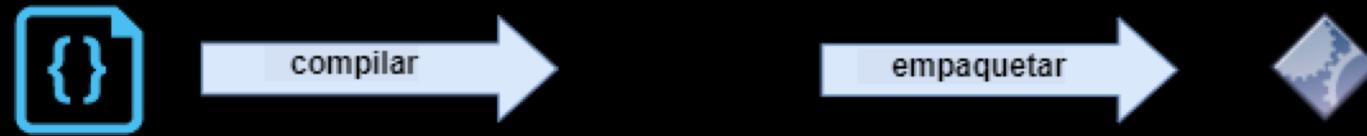
# Ciclo de vida de una aplicación

- Desarrollo del código (+ tests unitarios) y de la documentación
- Integración de la aplicación con el resto del ecosistema de aplicaciones
- Pruebas de regresión y de rendimiento
- Puesta en producción

# Modelos de desarrollo de software

- Modelo en cascada
- Modelo AGILE

# Pasos para ejecutar un programa



# Código fuente

- Es el conjunto de instrucciones escritas en un lenguaje de programación (java, C, PHP...). Estas instrucciones son legibles para un programador, de modo que es posible su estudio y el desarrollo de sus algoritmos.
- El código fuente NO puede ser ejecutado por una CPU.

# Compilador

- Programa informático que traduce código escrito en un lenguaje de programación a un programa equivalente que el procesador será capaz de ejecutar (código objeto).
- El compilador de java es 'javac'.

# Código objeto o código máquina

- El código objeto está compuesto por un conjunto de datos en **lenguaje binario** fruto de la traducción realizada por el compilador.
- Los archivos java compilados se denominan ‘ByteCodes’.
- En esta etapa el programa está listo para ser ejecutado.

# Linker o enlazador

- Es un programa que empaqueta varios ficheros objeto en lenguaje binario para crear un único fichero ejecutable.
- Además de los ficheros objeto obtenidos tras compilar el código, el enlazador añadirá las librerías de las dependencias del programa.
- El enlazador de java es 'jar'.

# Programa ejecutable

- El programa tiene un formato ejecutable que la computadora puede utilizar directamente para ejecutar las instrucciones.
- Los archivos ejecutables de java tienen la extensión .jar.

# Ejecución de un programa

- Los programas se **almacenan** en una **memoria no volátil** de almacenamiento masivo.
- Cuando se solicita la **ejecución** de un programa, este tiene que ser cargado en la **memoria principal** del equipo (RAM).
- El SO es el encargado de orquestar las ejecuciones de los distintos programas, ya que puede acceder directamente al procesador.

# Programa en ejecución

- A un programa en ejecución se le denomina **proceso**.
- El procesador ejecuta el programa, instrucción a instrucción, hasta que termina.
- Un programa puede terminar su ejecución de forma normal o a causa de un error (de software o de hardware).

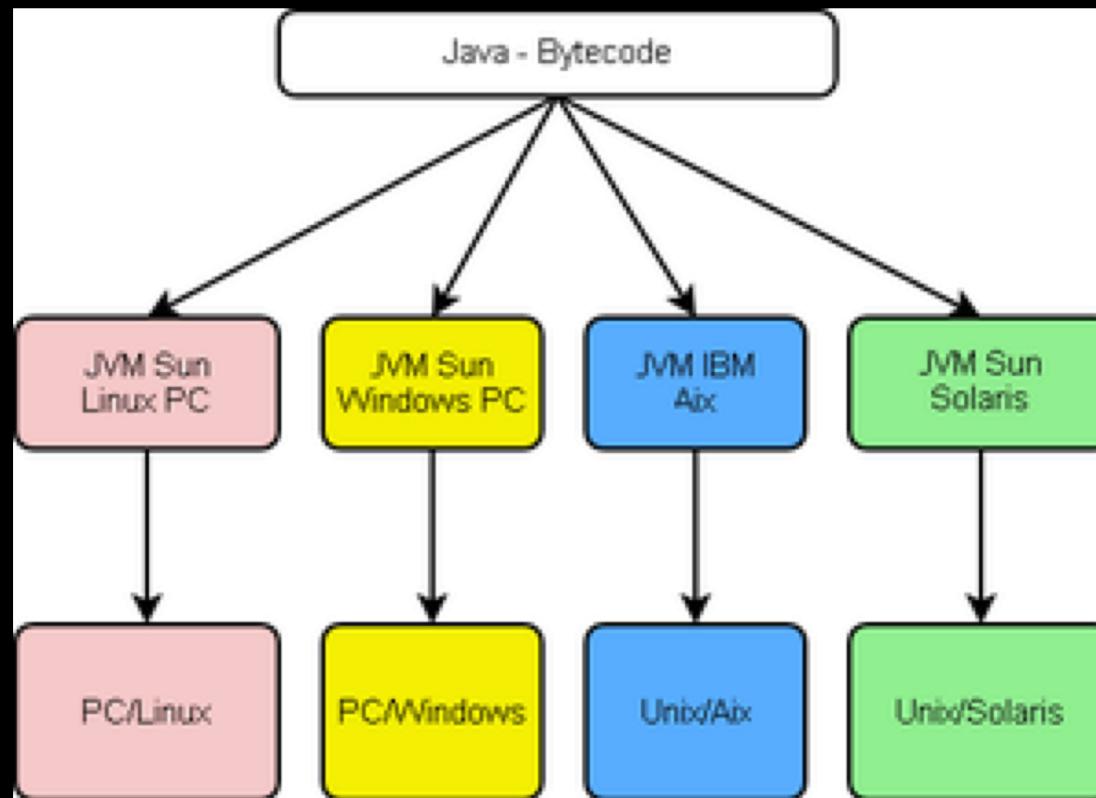
# Máquina virtual

- En informática, una máquina virtual es un software que simula un sistema de computación y puede ejecutar programas como si fuese una computadora real.
- Este software en un principio fue definido como "un duplicado eficiente y aislado de una máquina física".
- La acepción del término actualmente incluye a máquinas virtuales que no tienen ninguna equivalencia directa con ningún hardware real.

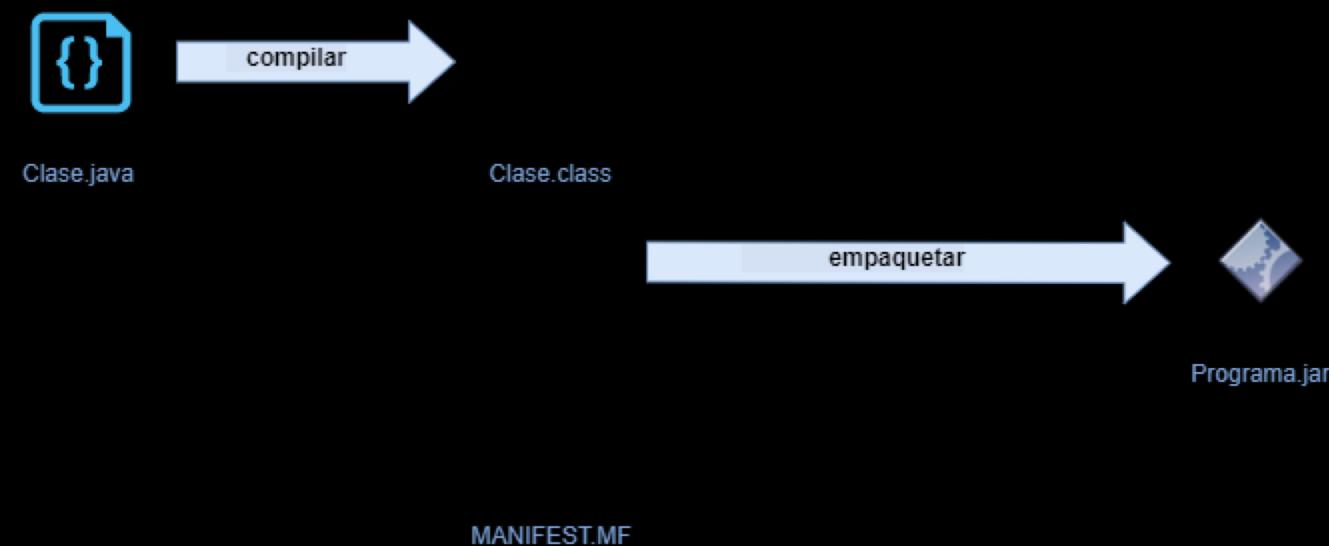
# Máquina virtual de sistema



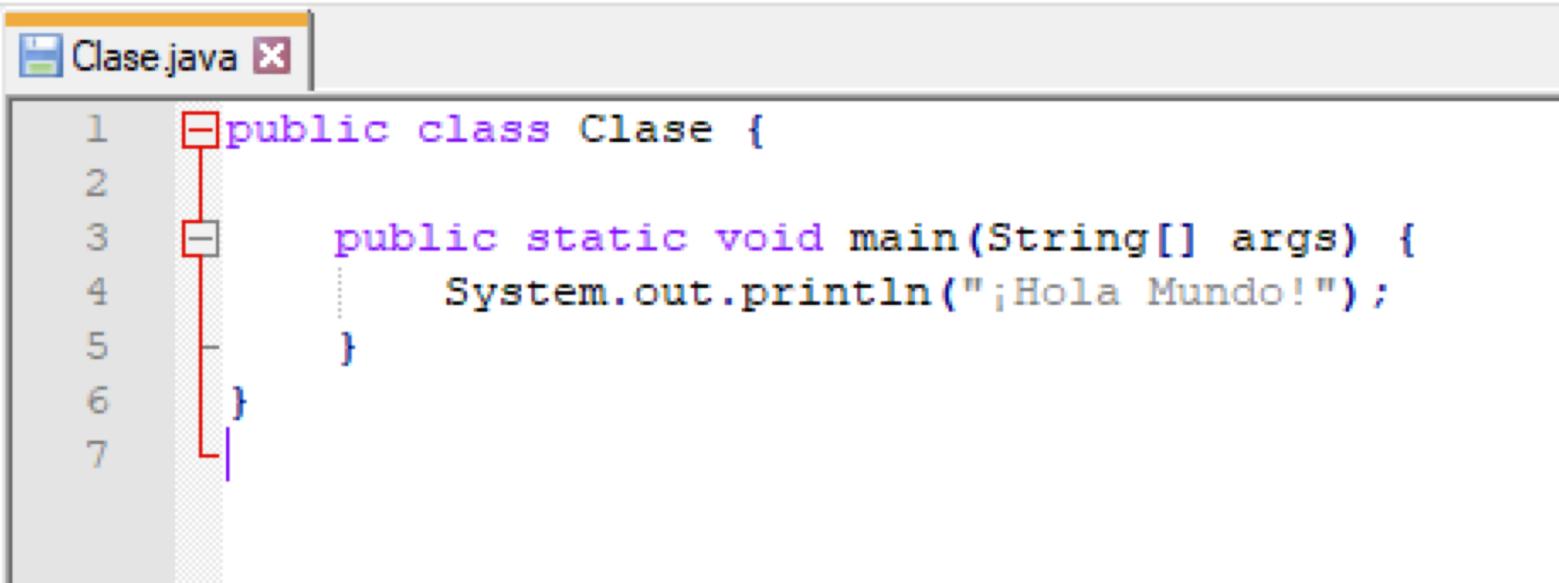
# Máquina virtual de proceso



# Ejemplo: pasos para ejecutar un programa java



# Ejemplo: Clase.java



```
1  public class Clase {  
2  
3      public static void main(String[] args) {  
4          System.out.println("Hola Mundo!");  
5      }  
6  }  
7
```

# Ejemplo: Clase.class

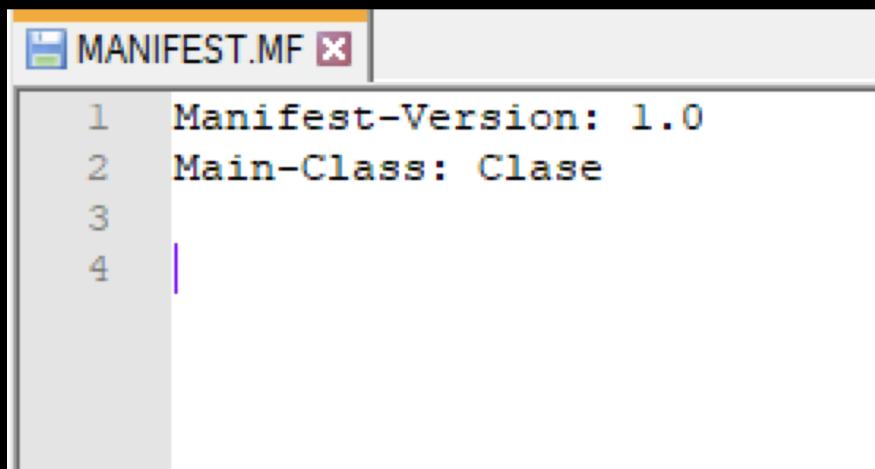
- Los ficheros ByteCode generados por el compilador se pueden ejecutar con el siguiente comando:

*> java Clase*

```
D:\BIBLIOTECAS\Escritorio\prueba>java Clase
¡Hola Mundo!
```

# Ejemplo: MANIFEST.MF

- Es un archivo especial que contiene “meta” información acerca de los archivos empaquetados dentro de un archivo JAR.
- De este modo se consigue que los archivos JAR puedan servir para variedad de propósitos.



A screenshot of a code editor window titled "MANIFEST.MF". The window contains the following manifest entries:

```
1 Manifest-Version: 1.0
2 Main-Class: Clase
3
4
```

# Ejemplo: Programa.jar

- Los archivos ejecutables se pueden ejecutar de dos formas:  
Mediante la GUI: haciendo doble click (al finalizar el programa se cerrará la consola).  
Mediante una consola de comandos: utilizando el comando *java -jar Programa.jar*

```
D:\BIBLIOTECAS\Escritorio\prueba>java -jar Programa.jar  
¡Hola Mundo!
```