

# Fundamentos de programación

Introducción a Java

# Índice

Objetivos de aprendizaje

## 1. Introducción a Java

1.1. ¿Qué es Java?

1.2. Un poco de historia

1.3. Principios de diseño en Java

## 2. Componentes de Java

2.1. JRE: Java Runtime Environment

2.2. JDK: Java Development Kit

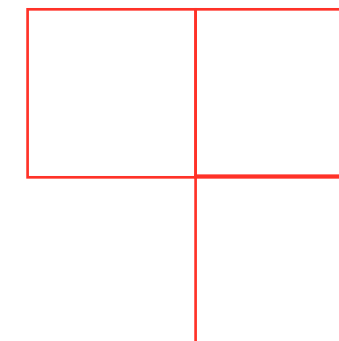
2.3. IDE: Integrated Development Environment

## 3. Configurar tu equipo para desarrollar en Java

## 4. Mi primer programa en Java: ¡Hola Mundo!

4.1. Consideraciones previas

Referencias bibliográficas





## Objetivos de aprendizaje

Los objetivos que se pretenden alcanzar con este recurso son los siguientes:

- Conocer el lenguaje de Programación Java y sus principales características.
- Conocer la plataforma de ejecución y de desarrollo (JRE – JDK) de Java.
- Instalar Java y las herramientas para desarrollo en Fundamentos de Programación.
- Implementar el primer programa en Java: *¡Hola Mundo!*



# 1. Introducción a Java

## 1.1. ¿Qué es Java?

**Java** es un lenguaje de programación así como una plataforma informática (Máquina Virtual de Java – *Java Virtual Machine*) que *Sun Microsystems* lanzó al mercado en 1995. Es muy importante para la comunidad de usuarios de sistemas informáticos dado que muchas aplicaciones, así como sitios web, funcionan con esta tecnología. Debe instalarse en los equipos para poder ejecutar dichos recursos (en algunos sistemas operativos macOS viene instalada de fábrica en los equipos).

Enlace



Para un mayor detalle sobre qué es Java y para que puede utilizarse, consulta este enlace.

`<https://www.java.com/es/download/faq/whatis\_java.xml>`



**Figura 1.** Logotipo de Java, una taza de café. fuente: [www.java.com](http://www.java.com)

# 1. Introducción a Java

## 1.2. Un poco de historia

- Java aparece a principios de los años 90.
- Es desarrollado por *James Gosling* en SUN Microsystems.
  - Escribió un manifiesto quejándose de la necesidad de adaptar cada programa al Sistema Operativo/Máquina.
  - En su compañía (SUN Microsystems) le encomendaron la misión de crear una solución para semejante reto.
- *Write Once, Run Anywhere* (escríbelo una vez, ejecútalo en cualquier lugar). Es la filosofía de Java, permitir que un mismo código funcione en diferentes Sistemas Operativos sin complicadas compilaciones para cada sistema.
- Java es el primer lenguaje "post-Web". Se diseñó teniendo en cuenta la comunicación a través de la red.



# 1. Introducción a Java

## 1.3. Principios de diseño en Java

- El paradigma utilizado en Java es la **Programación Orientada a Objetos** (POO). Sólo permite programar bajo este paradigma, **todo código se encuentra dentro de una clase**.
- Se puede ejecutar el mismo programa en diferentes Sistemas gracias a la Máquina Virtual de Java (JVM).
- Inclusión por defecto de soporte para trabajo en red.
- Ejecución segura de código en sistemas remotos.
- Facilidad de uso, al incorporar algunas de las mejores características de otros lenguajes como C++.





## 2. Componentes de Java

Para desarrollar y ejecutar programas Java, se necesitan tener instalados varios componentes. En concreto son estos 3:

1. **JRE** (*Java Runtime Environment*). El entorno de ejecución de Java, que se compone de la JVM (Java Virtual Machine) y las librerías básicas del lenguaje. Sirve para poder ejecutar programas Java.
2. **JDK** (*Java Development Kit*). Es el kit de desarrollo en Java y se compone del JDK y de herramientas para el desarrollo de programas: compilador, depurador, javadoc, etc.
3. **IDE** (*Integrated Development Environment*). Se trata del entorno de desarrollo integrado. Es un programa que permite trabajar con el código para crear nuevos programas, entre otras muchas funcionalidades.

### Importante



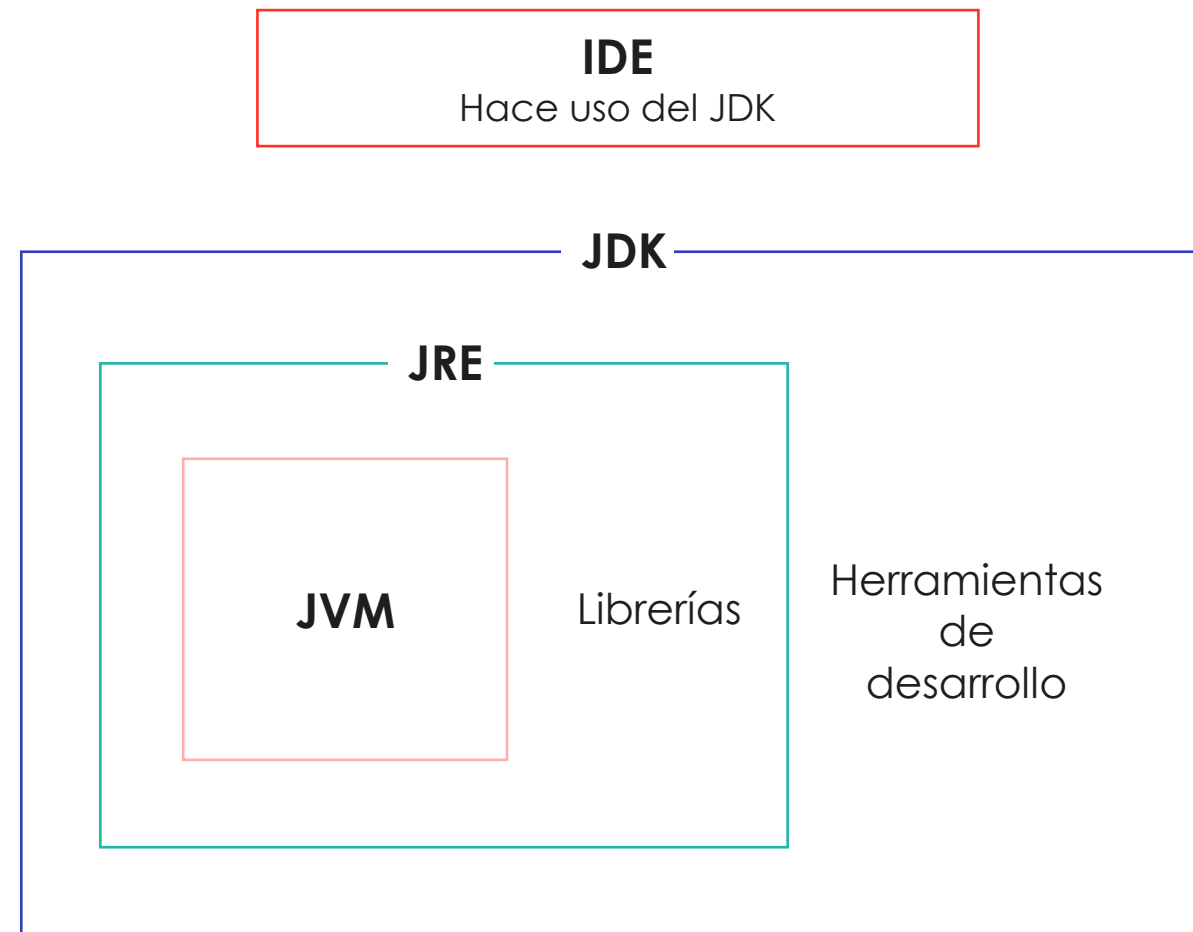
Dichos 3 componentes deben instalarse en ese orden (JRE, JDK e IDE) dado que cada uno tiene dependencias de los anteriores. Ver el punto “3. Configurar equipo para desarrollar en Java” de esta presentación para saber cómo instalarlos.



## 2. Componentes de Java

Vista gráfica de los elementos de Java

1. **JRE** (*Java Runtime Environment*)
  - **JVM** + Librerías
2. **JDK** (*Java Development Kit*)
  - **JRE** + Herramientas de desarrollo



**Figura 2.** JVM, JRE, JDK e IDE para Java



## 2. Componentes de Java

### 2.1. JRE: Java Runtime Environment

*Java Runtime Environment (JRE)* o el Entorno de Ejecución de Java consiste en un paquete de software que **permite la ejecución de programas desarrollados en Java**. Por norma general, debe instalarse en el sistema operativo (SO) para poder utilizarlo como es el caso de Windows o Linux. En otros casos, como en los sistemas macOS de Apple, el JRE viene instalado de fábrica.

El JRE no solo se encuentra instalado en equipos de consumo (ordenadores), sino que también existen variantes para dispositivos móviles (en Android en para la ejecución de APK), IOT, equipos industriales, equipos empotrados (en industria o automóviles), y un largo etc.

EL JRE se compone de dos elementos:

1. JVM
2. Librerías



## 2. Componentes de Java

### 2.1. JRE: Java Runtime Environment

#### **JVM (Java Virtual Machine o Máquina Virtual de Java)**

- Es una capa de software que se encuentra entre el Sistema Operativo y los programas Java, a los que da soporte.
- Hace independiente el programa Java del Sistema Operativo sobre el que ejecuta. Hay JVM para diversos SO.
- Gestiona los recursos que ofrece el SO y los pone a disposición de las aplicaciones Java.
- Gestión inteligente de la memoria, que entre otras cosas libera aquella que en el código deja de utilizarse.

#### **Librerías (core clases y ficheros de soporte)**

- Dan servicio a las aplicaciones ofreciendo todo tipo de ayudas al desarrollo disponibles para los desarrolladores.
- Algunos ejemplos son: Gestión de Entrada/Salida, Interfaces gráficas, Conexión con bases de datos, Impresión, Sonido, Red y concurrencia, Colecciones de datos, Seguridad, ... entre otras muchas.



## 2. Componentes de Java

### 2.2. JDK: Java Development Kit

*Java Development Kit* (JDK) o el Kit de Desarrollo Java consiste en el SDK (*Software Development Kit* – Kit de desarrollo Software) para java. Se trata de un conjunto de herramientas software que **permiten la creación de programas codificados en Java**. En todos los casos debe instalarse en el sistema operativo (SO) para poder utilizarlo, dado que en ninguno viene instalado de fábrica.

Algunos de los componentes más importantes del JDK son los siguientes:

- **javac** - Compilador de java que transforma el código fuente en código ejecutable por la JVM.
- **jar** – Generador de ficheros JAR. Sirve para agrupar varios ficheros fuente así como para crear ejecutables java.
- **javadoc** – Generador de documentación de código. Sirve para documentar las clase y librerías desarrolladas con java.
- Otros componentes importantes son el depurador (**jdb**), visor de applets (**appletviewer**), consola de java (**Jconsole**). En total, son más de 30 componentes para la asistencia en el desarrollo y la validación de programas java.



## 2. Componentes de Java

### 2.3. IDE: Integrated Development Environment

Integrated Development Environment (IDE) o el Entorno de Desarrollo Integrado no es un elemento como tal de Java, pero es realmente útil para el desarrollo de programas en general, y los programas Java no son una excepción. Ofrece a los desarrolladores asistencia en el desarrollo mediante diferentes componentes, todos integrados en la misma ventana:

- Editor de código. Con asistencia y sugerencias, detección de errores sintácticos, autocompletar, etc.
- Compilación y Ejecución. Permite ejecutar el programa desde el propio entorno simplemente pulsando un botón.
- Depuración. Permite validar cada paso del código, verificar valores de variables, insertar puntos de ruptura, etc.
- Gestión de proyectos y ficheros. Mediante exploradores de ficheros y visores de los mismos.
- Además de otras como el desarrollo en múltiples lenguajes, instalación de extensiones y complementos, etc.

En el desarrollo con Java suelen utilizarse diferentes IDE como Eclipse (el que se usa en este curso) , NetBeans o Visual Studio. Pero hay otros muchos, la variedad es muy extensa.



### 3. Configurar tu equipo para desarrollar en Java

Se deben instalar los componentes descritos previamente en el siguiente orden:

1. **JRE:** Acudir a la web de Java ([www.java.com](http://www.java.com)) y comprobar si se tiene instalada la última versión. Si no es así, descargar la versión compatible con el sistema operativo de usuario e instalarla.
2. **JDK:** Acudir a la web de Oracle ([www.oracle.com](http://www.oracle.com)) y buscar el JDK de java. Una vez localizada la última versión, descargar aquella compatible con el sistema operativo del usuario e instalarla. No permitirá hacerlo si no se ha instalado previamente el JRE.
3. **IDE:** Acudir a la web de Eclipse ([www.eclipse.org](http://www.eclipse.org)) y descargar la última versión. Se descargará una herramienta de instalación en la que se debe indicar que se quiere desarrollar en Java (Eclipse sirve para otros muchos lenguajes de programación). Eclipse no funcionará si no se ha instalado previamente el JDK.



### 3. Configurar tu equipo para desarrollar en Java



Figura 3. Descargar java. fuente: [www.java.com](http://www.java.com)

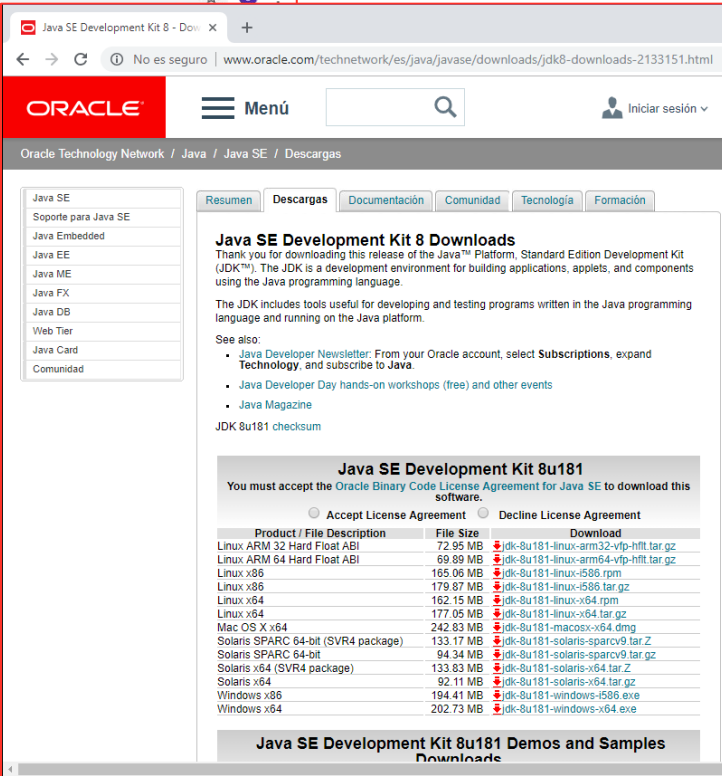


Figura 4. Descargar JDK. fuente: [www.oracle.com](http://www.oracle.com)

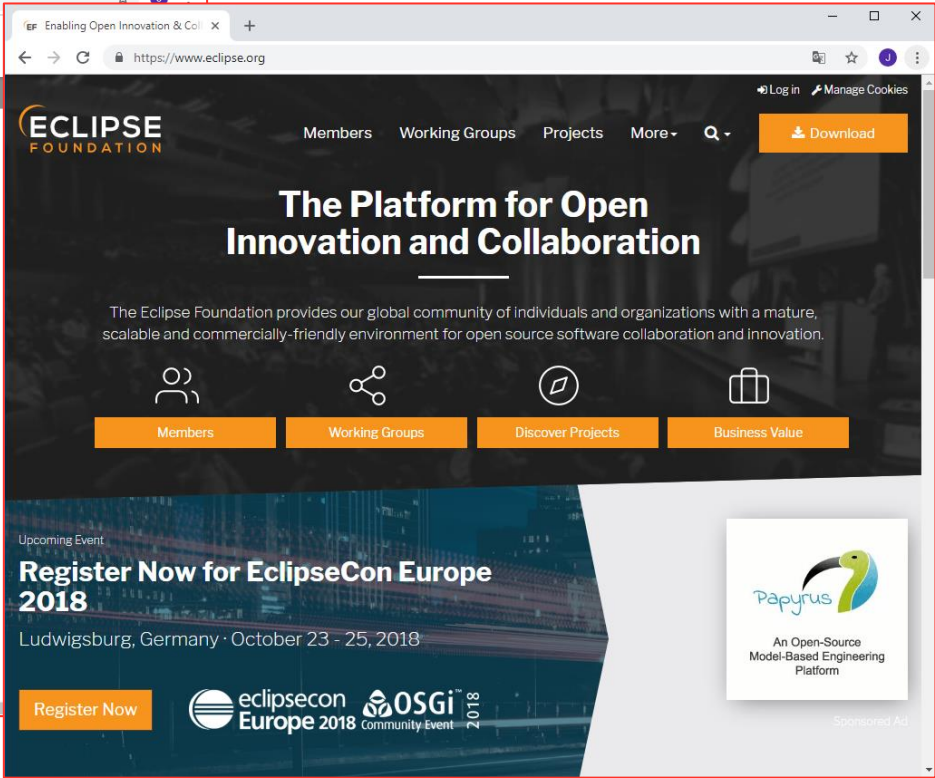


Figura 5. Descargar Eclipse. fuente: [www.eclipse.org](http://www.eclipse.org)



## 4. Mi primer programa en Java: ¡Hola Mundo!

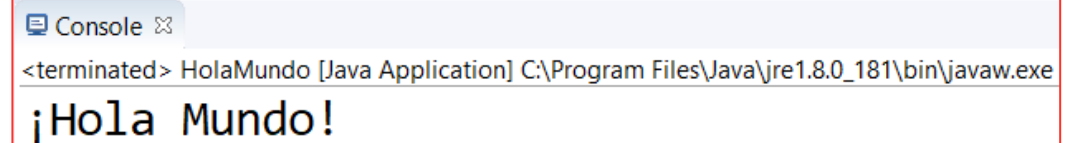
En todo lenguaje de programación, al desarrollo más básico para probar “que todo funciona” y que está debidamente instalado y configurado se le denomina ¡Hola Mundo! (*Hello World!*). Es un clásico entre desarrolladores.


A continuación se muestra el código Java del ¡Hola Mundo! así como el resultado (la salida por consola) de su ejecución:

### Ejemplo



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



Console 

<terminated> HolaMundo [Java Application] C:\Program Files\Java\jre1.8.0\_181\bin\javaw.exe

¡Hola Mundo!



## 4. Mi primer programa en Java: ¡Hola Mundo!

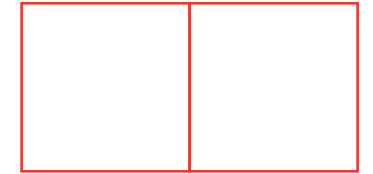
### 4.1. Consideraciones previas

Para poder ejecutar un programa, lo más sencillo es realizarlo con el IDE y ejecutarlo desde allí, como se mostraba en la diapositiva anterior. Para ello, hay que realizar una serie de pasos:

1. Abrir el IDE (Por ejemplo Eclipse) y crear un nuevo proyecto Java: Darle un nombre representativo, en este caso *HolaMundo* o *MiPrimerProyecto*. Es indistinto que se escriba en mayúsculas o minúsculas y es recomendable sin espacios en blanco.
2. Crear una nueva clase: Darle un nombre representativo, con la primera letra en mayúscula **H**olaMundo. Esto es un formalismo para las clases, su nombre en mayúsculas. Se puede habilitar el check para generar automáticamente el “main” o programa principal.
3. Escribir el código dentro de la clase y Guardar.
4. Ejecutar el programa



## Referencias bibliográficas



Differences between JDK, JRE and JVM. <<https://www.geeksforgeeks.org/differences-jdk-jre-jvm/>>

Eckel, Bruce (2008). "Thinking in Java" 4th ed. Prentice Hall.

Eclipse.org <<https://www.eclipse.org/>>

Horstmann, C.S. (2018). "Core Java I – Fundamentals" 11th ed. Prentice Hall.

Horstmann, C.S. (2016). "Core Java II – Advanced Features" 10th ed. Prentice Hall.

Java <<https://www.java.com/es/download/>>

Java in Visual Studio Code <<https://code.visualstudio.com/docs/languages/java>>

NetBeans <<http://netbeans.apache.org/>>

Oracle JDK <<https://www.oracle.com/java/technologies/javase/javase-jdk8-downloads.html>>

Schildt, H. (2018). "Java. A Beginner's Guide" 8th ed. Oracle Press.

Schildt, H. (2018). "Java. The Complete Reference" 11th ed. Oracle Press.

What is java. <[https://www.java.com/es/download/faq/whatis\\_java.xml](https://www.java.com/es/download/faq/whatis_java.xml)>



Todos los derechos de propiedad intelectual de esta obra pertenecen en exclusiva a la @ Universidad Europea. Queda terminantemente prohibida la reproducción, puesta a disposición del público y en general cualquier otra forma de explotación de toda o parte de la misma.

La utilización no autorizada de esta obra, así como los perjuicios ocasionados en los derechos de propiedad intelectual e industrial de la @ Universidad Europea, darán lugar al ejercicio de las acciones que legalmente le correspondan y, en su caso, a las responsabilidades que de dicho ejercicio se deriven.

**Ve más allá**