# Java JRE y JDK: compile y ejecute su primer programa

## Conclusión 1: ¿Qué es Java?

En la clase introductoria ya hemos aprendido varios temas fundamentales sobre Java.

Hablamos sobre las principales características de Java como:

* Orientado a objetos.
* Parecido a C++.
* Muchas librerías y una gran comunidad.

Además de eso, aprendimos:

* La diferencia entre código fuente y Bytecode.
* Para ejecutar el Bytecode necesitamos tener la máquina virtual de Java.
* El Bytecode es independiente del sistema operativo.

Vimos también los principales componentes de la plataforma Java, que son:

* Java Virtual Machine (JVM).
* Lenguaje Java.
* Librerias Java (API).

En la próxima clase vamos a escribir nuestro primer código en Java. ¡Continuemos!

|  |
| --- |
|  |

## Conclusión 2: El primer programa

* Cuál es la diferencia entre JRE y JDK.
* Cómo compilar el código fuente de Java desde la línea de comandos (javac).
* Cómo ejecutar Bytecode en la línea de comando (java).
* Un programa Java debe escribirse dentro de una clase (class).
* Cada instrucción Java debe terminar con ;.
* Para abrir y cerrar un bloque usaremos las llaves {}.
* Un programa Java tiene una entrada que es una función (método) main.
* Para imprimir algo en la consola, usamos la declaración System.out.println ().

|  |
| --- |
|  |

## Conclusión 3: Programando con Eclipse

* El papel de un IDE y su diferencia con respecto a un editor.
* Cómo descargar el IDE de Eclipse.
* ¿Para qué sirve workspace?
* El concepto de perspectiva.
* Cómo crear un proyecto Java, incluidas las clases y cómo ejecutarlo.
* Cómo mostrar diferentes views.

|  |
| --- |
|  |

## Conclusión 4: Tipos y variables

En esta clase comenzamos nuestro aprendizaje con variables y tipos primitivos de Java. Los tipos vistos con más detalle fueron int (entero) y double (decimal). Los cuales usamos para hacer operaciones aritméticas y también concatenar con texto.

Durante el capítulo hablamos de buenas prácticas al nombrar clases y también variables. Comenzamos las clases con mayúscula y nuestras funciones y variables con minúsculas. Hablaremos más sobre esto en el futuro.

Fue posible comprender un poco de conversión de tipos y cómo podemos pasar un valor de un tipo para una variable de otro. En algunos casos no necesitamos hacer nada, ya que el casting es implícito y en otros debemos dejar en claro al compilador que sabemos lo que estamos haciendo, mostrando entre paréntesis el tipo que queremos usar.

Esto concluye esta lección. Siguiente paso: Caracteres. ¡Te espero allí!

|  |
| --- |
|  |

## Conclusión 5: Trabajando con caracteres

* El concepto y cómo declarar char y String.
* Cómo concatenar Strings.
* Atajo para crear método main.
* Las variables almacenan valores y no referencias.

|  |
| --- |
|  |

## Conclusión 6: Practicando condicionales

* Cómo usar el if.
* Cómo usar las operaciones lógicas AND (&&) y OR (||).
* Trabajar con el alcance de las variables.
* También hemos visto algunos atajos en Eclipse:
* main Ctrl + espacio, para generar el método main.
* ctrl + shift + f, para formatear el código fuente.
* sysout + ctrl + espacio, para generar la instrucción System.out.println ().

|  |
| --- |
|  |

## Conclusión 7: Controlando flujo con Loops

## Anexos:

* <https://www.jetbrains.com/es-es/lp/devecosystem-2019/java/>
* <https://www.oracle.com/java/technologies/downloads/#java8-windows>
* <https://docs.oracle.com/javase/8/docs/>
* <https://www.oracle.com/cl/java/technologies/javase/javase8-archive-downloads.html>
* <https://stackoverflow.com/questions/9298980/variable-out-of-type-printstream-error-occured>
* <https://www.eclipse.org/ide/>
* No se puede usar una variable o print si no se declara completa

## Código:

## Vocabulario:

* Features
* Software de código abierto (Open-source software)
* Bytecode
* Opcode
* Tools: Herramientas
* Workspace
* Pistas
* Outline
* Package: estructura de carpetas donde se guardan las clases
* Compilar
* Inicializar variables