



Curso de Java Standard



Ing. Octavio Robleto



octavio.robleto@gmail.com



<https://octaviorobleto.com>



Que es la programación?

Programar no es otra cosa que indicarle a una computadora, qué actividades realizar, de qué forma y en qué orden.

Siendo un poco mas formales, podemos decir que programar es escribir instrucciones para realizar una actividad en algún lenguaje de programación con la finalidad de que sean ejecutadas por la computadora para solucionar un problema.

Dicho de una manera mas simple, si el programador le “dice” a la computadora **“Apágate”**, la computadora lo hará. La computadora no hará nada que no le digamos que haga. Tampoco hará lo que nosotros suponemos que hará, ni hará lo que nosotros creamos que debería hacer.

La computadora únicamente hará lo que explícitamente le digamos que haga



Que es Java?

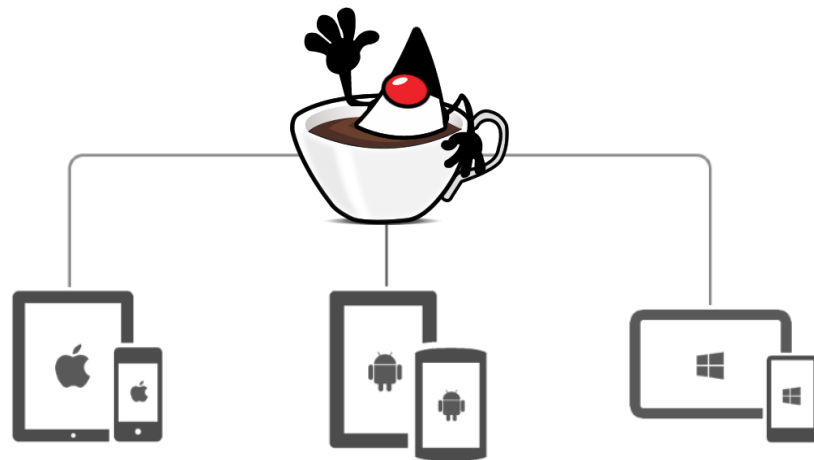
Java es un lenguaje de programación y una plataforma informática con una curva de aprendizaje baja (Se puede decir que es fácil de aprender)

Nació con el objetivo de ser un lenguaje de programación de estructura sencilla que pudiera ser ejecutado en diversos sistemas operativos, lo que llamamos Multiplataforma.

Puede ser utilizado en proyectos pequeños y de gran envergadura, además posee una de las comunidades de desarrolladores más grandes del mundo.

Es un lenguaje Multiparadigma, pero su base fundamental es el Paradigma orientado a objetos.

Es muy popular y se pueden desarrollar, proyectos WEB, Videos Juegos, Apis, Sistemas Operativos Móviles y muchas otras cosas.



Organización de Java



JME (Java Platform, Micro Edition) ofrece un entorno flexible y sólido para aplicaciones que se ejecutan en dispositivos móviles e integrados: teléfonos móviles, TDT, reproductores Blu-ray, dispositivos multimedia digitales, módulos M2M, impresoras y mucho más.



JSE (Java Standard Edition) ofrece un entorno de desarrollo de aplicaciones de escritorio, similares a las aplicaciones tipo ventanas creadas con Visual Basic o Delphi. Incluye la funcionalidad básica del lenguaje como manejo de clases, colecciones, entrada/salida, acceso a base de datos, manejo de sockets, hilos de ejecución, etc.



JEE (Java Enterprise Edition) Es un conjunto de tecnologías y especificaciones relacionadas que están orientadas hacia el desarrollo de aplicaciones empresariales a gran escala, extiende de JSE y lo complementa para lo antes indicado.



Como trabaja?

Java es un lenguaje de programación que necesita ser compilado e interpretado, pero ¿qué significa esto?

La diferencia radica en que un lenguaje compilado (C,C++,GO) con uno interpretado (Java Script, Ruby) es que el lenguaje compilado requiere un paso adicional antes de ser ejecutado, *la compilación*, que convierte el código que escribes a lenguaje de máquina. Un lenguaje interpretado, sin embargo, es convertido a lenguaje de máquina a medida que es ejecutado.

Java necesita los dos pasos, compilar a un nivel intermedio llamado **bytecode**, que después es interpretado.

Cuando creamos un archivo en Java este posee la extensión **NombreArchivo.java** y después de compilado se crea un archivo **NombreArchivo.class** que es el archivo en **bytecode**.

Para poder interpretar este **bytecode** debes tener instalado JRE (Java Runtime Environment).

Si deseas compilar código Java no es suficiente instalar el JRE, necesitas el JDK (Java Development Kit) que incluye el compilador, entre otras herramientas de desarrollo.



Sintaxis



Como cualquier lenguaje de programación, Java tiene su propia estructura, reglas de sintaxis, y este no es la excepción; Java es un derivado del lenguaje C, por lo que sus reglas de sintaxis se parecen mucho a C.

Poco a poco veremos hay varias formas de escribir palabras en Java por lo que tocaremos estos conceptos de forma genérica.

Yo voy a salir a correr



Yo correr a voy salir



Bloques de Código y Sentencias

Bloque de código: Es un conjunto de sentencias los cuales están delimitados por llaves.

En algunos casos lo que necesitamos que se ejecute es tan simple que solo requerimos ejecutar una sentencia en el bloque, para estos casos podemos omitir las llaves, suele confundir al principio pero veremos que con el tiempo estaremos mas seguros de cuando vamos a hacer esto.

Sentencias: Es la unidad mínima de ejecución de un lenguaje de programación para resolver un problema en particular y siempre debe terminar con un punto y coma “;”

```
bloque { // inicio de bloque  
    sentencia1;  
    sentencia2;  
} // fin de bloque
```

Comentarios

Sabemos que hay unas reglas para escribir en este lenguaje, pero, ¿qué pasa si nos toca revisar un código realizado por un colega o desarrollador, o si, simplemente nos toca ver algo hecho por nosotros que no hemos visto hace tiempo?

Esto también lo tienen contemplado los lenguajes de programación con los bloques de comentarios, que nos ayudan a escribir y a explicar el código y cuando el compilador de Java ve estas estructuras omiten la evaluación de la sintaxis:

Comentarios de documentación: se comienza con barra y doble asterisco `/**`, cada línea intermedia debe poseer un asterisco `*` al principio y para finalizar el comentario un asterisco con barra `*/`.

Comentarios de Múltiples Líneas: se hacen con barra y asterisco `/*` para empezar y asterisco barra `*/` para finalizar el comentario.

Comentarios de una sola Línea: se hacen doble barra `//`.

```
/**Inicio del comentario de documentación
 *Para este comentario se pueden aprovechar las etiquetas HTML
 *Por ejemplo <h1> Inicio del Software </h1>
 *fin del comentario */

/*Inicio del comentario
Comentario de
Varias Lineas
fin del comentario */

//Comentario de una sola linea
```


Nomenclatura

Así como cuando escribimos un carta o un correo que debemos colocar los nombres de personas en mayúsculas, comenzar una oración en mayúsculas; en Java también hay ciertos estándares que aunque no dan error es bueno seguirlos.

Estos estándares nos ayudan a escribir palabras compuestas en general, además, nos ayudan a leer mas cómodamente el código y que sea más fácilmente entendible

Upper Case: Todas las letras del nombre se encuentran en mayúsculas, por ejemplo: EJEMPLODENOMENCLATURA. Es usado por lo general para las constantes.

Camel Case: El nombre viene porque se asemeja a las dos jorobas de un camello, y se puede dividir en dos tipos:

- **Upper Camel Case**, es cuando la primera letra de cada una de las palabras es mayúscula. Ejemplo: EjemploDeNomenclatura. Es usado para nombrar las clases.
- **Lower Camel Case**, igual que la anterior con la excepción de que la primera letra es minúscula. Ejemplo: ejemploDeNomenclatura. Es usado para paquetes, variables y métodos.

Snake Case: Cuando cada una de las palabras, se separa por un guión bajo (_). Ejemplo: ejemplo_de_nomenclatura. Es usado también para variables.



IDE (Integrated Development Environment)

Aunque podemos usar el editor de texto mas básico que posea el sistema operativo para construir programas en Java, debemos de realizar unos comandos de consola como **javac**(Compilar) y **java**(Interpretar), pero existen aplicaciones que nos hacen el trabajo de forma mas fácil, eficaz y nos evitan cometer errores de sintaxis.

Eclipse es uno de los programas que tiene este fin, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica en algunos casos.



Maven

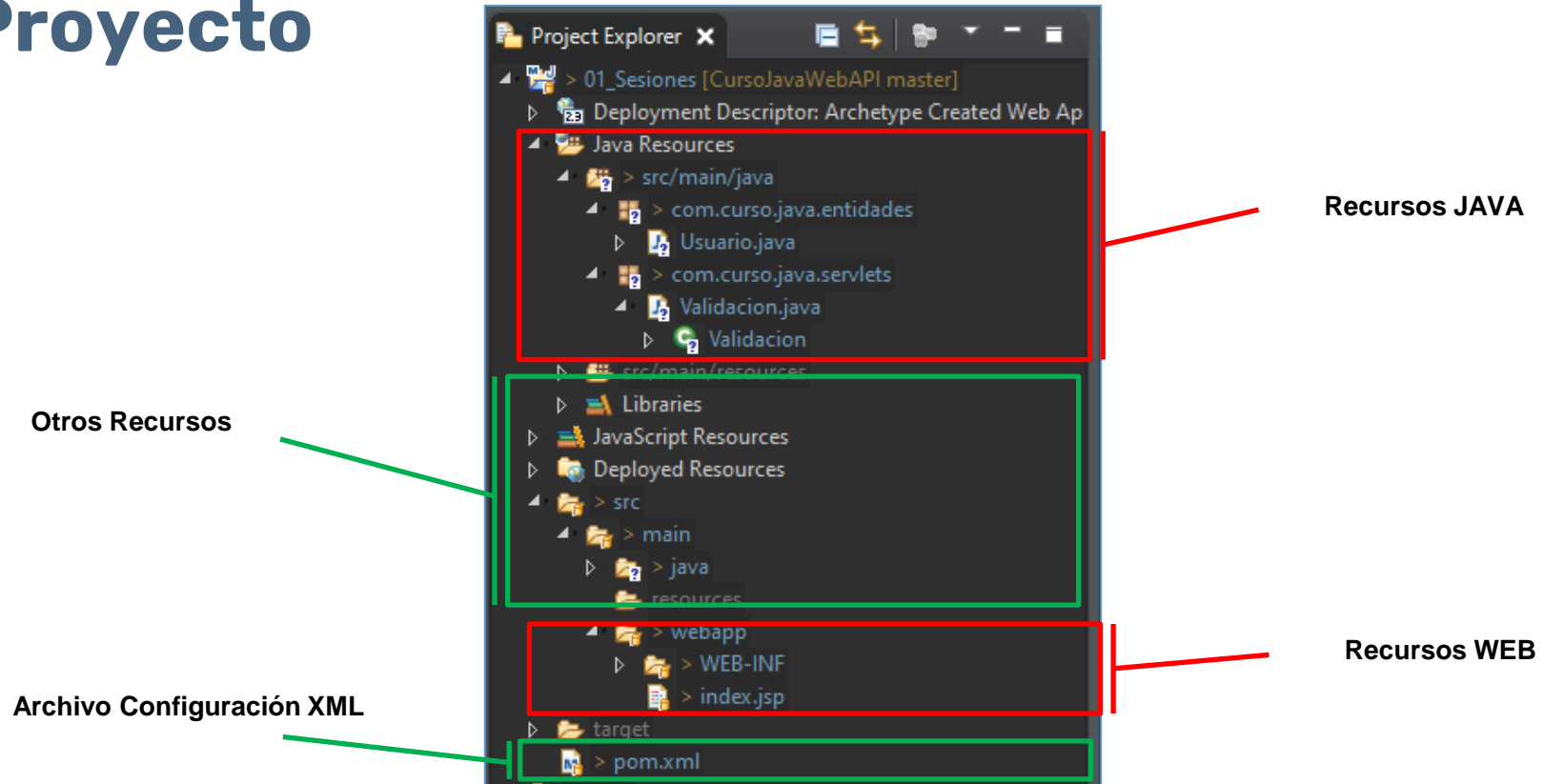
Como comenzamos a hacer nuestros proyectos y que estructura van a tener?

Los IDE nos dan diferentes formas de crear un proyecto, pero como todo, en el mundo IT es bueno registrarse por estándares y buenas practicas y Maven solventa esto y mucho mas como por ejemplo:

- Gestionar de dependencias (librerías) y sus versiones
- Construcción de proyectos (de código a entregable .war, .ear, .zip, etc.)
- Estructura única de proyecto compatible con todos los entornos de desarrollo y sistemas de integración continua



Proyecto



Descargas



<https://www.oracle.com/technetwork/es/java/javase/downloads/index.html>



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



Nuestro Primer Programa

El primer programa que todo desarrollador hace cuando tenemos el contacto inicial con un nuevo lenguaje de programación es un “Hola Mundo”.

Para esto debemos crear el método **main**, que en java es un estándar utilizado por la **JVM** para iniciar la ejecución de cualquier programa, es como la llave de encendido de un automóvil, por allí empieza todo.

```
public class Principal {  
    public static void main(String[] args) {  
        //Salida de datos por Pantalla  
        System.out.println("Hola Mundo");  
    }  
}
```

Mostrar información

Como vimos en la lamina anterior, mostramos un mensaje en la consola del IDE, para ello usamos la utilidad `System.out.println()` o también podemos usar `System.out.print()`.

Básicamente estos dos métodos son útiles para mostrar texto por pantalla y la única diferencia es que `println()` muestra el texto y luego deja un salto de línea, mientras que `print()` no lo hace.

```
System.out.println("Hola Mundo con un salto de linea al final");  
System.out.print("Hola Mundo");
```