

Programación Orientada a Objetos

Se trata de un paradigma de programación donde se organiza el código en unidades denominadas clases, de las cuales se crean objetos que se relacionan entre sí, simplificando el desarrollo de software y el mantenimiento de las aplicaciones

POO vs Programación estructurada

Programación Orientada a Objetos	Programación Estructurada
Forma de programar más cercana a cómo expresarías las cosas en la vida real	Pretende resolver un problema de principio a fin en una sola estructura de código
Análisis orientado a los objetos	Análisis orientado a los procesos del sistema
Existen diferentes formas de afrontar la solución a un problema	Técnicas estructuradas básicas: Secuencia, selección, repetición
Las funciones y los datos son encapsulados en una entidad	Las funciones y los datos son manejados como entidades separadas

Clases y Objetos

// Clases

Grupo de variables y operaciones agrupadas en módulos coherentes. Una clase puede tener atributos, constructores y métodos. Podría decirse que es una plantilla que define la forma de un objeto.

```
v public class Libro {
    String nombre;
    String autor;
}
```

```
public class NombreClase {
    //Declaración variables de instancia
    tipoDato variable1;
    tipoDato variable2;
    //Declaración de métodos
    ...
}
```

// Objetos

Son instancias de una clase, es decir, cuando creas un objeto, este pertenece a cierta clase y existe una representación física de esa clase en memoria.

```
Libro libro = new Libro();
```



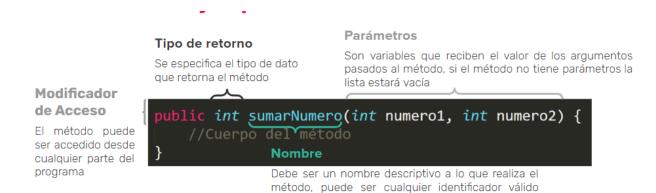
Métodos

Conjunto de instrucciones definidas dentro de una clase que realizan una tarea determinada.

¿Cómo se escribe un método?

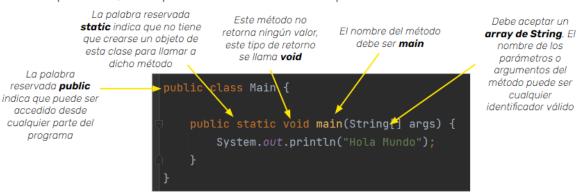
Se identifican por tener paréntesis después de su nombre. Un método tendrá:

- Un modificador de acceso
- Un tipo de retorno
- El nombre (debe ser descriptivo)
- Puede o no tener parámetros.



Método main()

Los programas en Java inician su ejecución mediante el método main(), en otras palabras, es el punto de entrada de la aplicación.



Métodos de instancia vs Métodos de Clase

Los $\mathbf{m\acute{e}todos}$ de $\mathbf{instancia}$ pueden ser accedidos mediante una instancia u objeto de una clase.

Métodos de clase

que no haya sido utilizado en el contexto actual

Un **método de clase** es aquel que puede ser llamado sin existir una instancia de la clase. Un ejemplo típico de uso es la clase java.lang.Math:

tipo de dato en la firma del método

Constructores

Toda clase en Java tiene un constructor, aunque no lo escribamos. Si no lo haces, Java creará un **Constructor por defecto** sin parámetros. El constructor se encarga de inicializar el valor de cada atributo de la nueva instancia.

Los atributos de tipo primitivo se inicializan a **0 o false** por defecto, mientras que los atributos de tipo objeto (referencia) se inicializan en **null**.

Java no permite que las variables miembro de una nueva instancia queden sin inicializar, por lo que si las mismas no se inicializan apuntan a null.

En este caso Libro() es un método ya que está seguido de paréntesis. Se trata de un tipo especial de método llamado constructor, el cual nos permite crear un nuevo objeto.

Declaración de un constructor

- El nombre del constructor debe coincidir con el nombre de la clase
- No tiene tipo de retorno
- La existencia de parámetros es opcional
- Puede existir más de uno, aunque sólo uno se ejecutará al momento de crear el objeto

```
//Constructor sin parámetros
public Libro() {
}
```

```
//Constructor con parámetros
public Libro(String nombre, String autor) {
   this.nombre = nombre;
   this.autor = autor;
}
```

Sobrecarga de un constructor (Overloading)

La sobrecarga permite que puedan declararse *varios constructores* para una misma clase (debe tener el mismo nombre que la clase), siempre y cuando, tengan un tipo y/o número de parámetros distinto. Hacerlo permite construir un objeto de diferentes maneras.