**Programando con B4X**

Tema 9 – Clases

Version 1.0, marzo 2021

Autor original: [Prokopis Pliroforikos](https://github.com/pliroforikos)

Traducido al español por [LaMashino](https://github.com/Lamashino)

[](https://creativecommons.org/licenses/by/4.0/)

# 

# Tema 9 – Clases

* ¿Qué es una clase?
* ¿Qué es un objeto?
* ¿Qué son los atributos y los métodos?
* Crear y usar una clase sencilla en B4J.

Lo que los estudiantes aprenderán

**3h**

A menudo en programación necesitamos referirnos a objetos similares de una única forma. Por ejemplo, los estudiantes en el colegio. Cada estudiante posee un nombre, una dirección, unas clases a las que asistir, títulos, etc. Para gestionar esta información un programador debe organizar estos datos sistemáticamente.

**Consejo para el profesor**

Las Clases son un tema complejo en programación. Evita entrar en detalles profundos como herencia, encapsulación, etc.



## Clases

En el ejemplo de los estudiantes podríamos decir que cada estudiantes posee la siguiente información:

|  |  |
| --- | --- |
| **Estudiante** | |
| 1 | Identificador Escolar |
| 2 | Nombre |
| 3 | Apellidos |
| 4 | Dirección |
| 5 | Teléfono |
| 6 | Correo electrónico |

También necesitamos las siguientes funciones:

* Registrar un nuevo estudiante
* Cambiar la información del estudiante
* Traspasar un estudiante
* Mostrar la información de un estudiante
* Borrar un estudiante

Así pues, para 3 estudiantes, tendríamos los siguientes elementos:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Estudiante 1** | |  | **Estudiante 2** | |  | **Estudiante 3** | |
| 1 | 125310 |  | 1 | 125311 |  | 1 | 125312 |
| 2 | Augusta |  | 2 | Maria |  | 2 | Muḥammad |
| 3 | Ada Byron |  | 3 | Curie |  | 3 | al-Khwarizmi |
| 4 | Londres |  | 4 | Varsovia |  | 4 | Jiva |
| 5 | 37535795 |  | 5 | 678433 |  | 5 | 646456456 |
| 6 | ada@lon.uk |  | 6 | maria@var.pol |  | 6 | algor@jiva.uz |

Del ejemplo anterior, podemos concluir que los estudiantes poseen datos y les podemos aplicar funciones similares. El hecho de agrupar los datos de los estudiantes y las funciones que podemos hacer con ellos en un único trozo de código se llama “**clase**”. Cada estudiante es un **Objeto** o **Instancia** de la clase. Las variables que caracterizan a un estudiante (nombre, teléfono, dirección, etc.) se llaman **propiedades** y las funciones que podemos aplicarles se llaman **Métodos**.

**Recuerda**

Llamamos **clase** al agrupamiento de datos y de funciones en un único trozo de código independiente.

**Objeto** o **Instancia** de la clase son todos los elementos independientes que resultan de usar la clase.

Las variables de un objeto se llaman **propiedades.**

Las funciones que aplicamos a un objeto se llaman **Métodos.**



Las ventajas de usar clases son la flexibilidad en el uso del código, mayor velocidad y facilidad de desarrollo de aplicaciones y la posibilidad de reutilizar el código en otros programas.

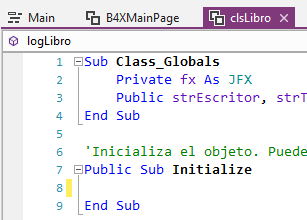
## Ejemplo de clase en B4J

Una biblioteca posee un conjunto de libros que presta a los lectores registrados. Cada libro tiene propiedades como el título, el autor, la editorial o el año de publicación. Los libros se puede insertar, mostrar o cambiar.

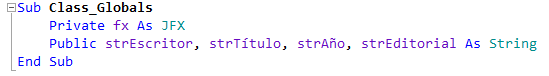
Crea un aplicación en B4J que implemente la clase Libro con las propiedades y métodos descritos.

### Metodología de implementación

1. Crea un aplicación **B4XPages** y ponle de nombre “biblioteca”.
2. Elige la opción **Proyecto – Añadir nuevo módulo – Módulo de Clase - Standard Class.**
3. En la ventana, ponle de nombre **clsLibro** (significa: clase Libro)
4. Aparecerá una nueva pestaña llamada **clsLibro**.



1. Dentro de la rutina **Class\_Globals**, añade las variables que representarán las propiedades de la clase:
   1. Título del libro
   2. Nombre del autor
   3. Editorial
   4. Año de publicación



1. Finalmente, implementa las subrutinas que ejecutarán los métodos:
   1. Insertar un libro
   2. Mostrar un libro
   3. Cambiar un libro

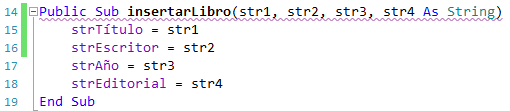
**Consejo para el profesor**

Ten cuidado de explicar que todas las variables y métodos que hemos creado son públicos, con lo que los objetos podrán usarlos.



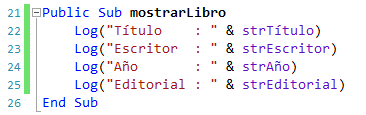
### Insertar un libro

Esta subrutina aceptará 4 cadenas de texto como parámetros y las asignará en el orden en que aparecen a las variables **strTítulo, strEscritor, strAño y strEditorial.**



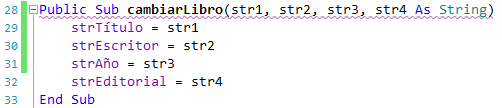
### Mostrar libro

Esta subrutina mostrará con el comando “Log” las propiedades del libro indicado.



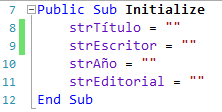
### Cambiar libro

Este método aceptará 4 parámetros para cambiar las propiedades del libro en el mismo orden que en el método “insertarLibro”:



### La rutina “Initialize”

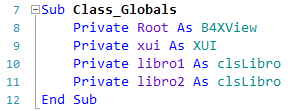
Esta rutina se usa para dar valores iniciales a las variables o bien para realizar alguna otra acción al crear un objeto de la clase:



### Cómo usar la Clase

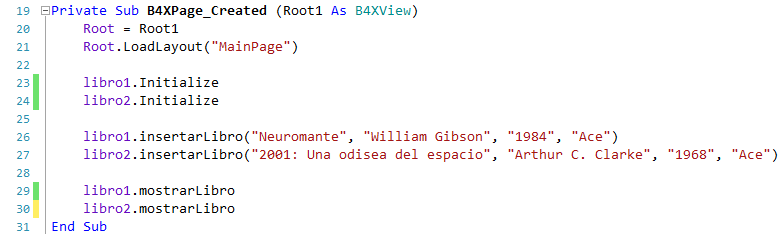
Volvemos a la pestaña **B4XMainPage** para usar la clase **clsLibro**.

1. Primero creamos los objetos de la clase clsLibro.

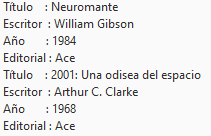


donde libro1 y libro2 son dos objetos de la clase clsLibro con todas las propiedades y métodos discutidos anteriormente.

### Uso de los métodos



El primer método que se usará en los objetos es el método **Initialize**.



**Recuerda**

Initialize **no es un método opcional**. Es el primer método que hay que invocar antes de usar un objeto.



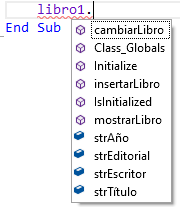
El método **insertarLibro** introduce los valores de los dos libros en las propiedades de los objetos.

El método **mostrarLibro** muestra las propiedades de cada libro.

**Recuerda**

Cada propiedad se puede consultar escribiendo el nombre del objeto, seguido de un punto y después el nombre de la propiedad o método. Algunos métodos necesitan parámetros y otros no.



Cada vez que escribas el nombre de un objeto y pulses la tecla con el punto, B4X mostrará una ventana con todas las propiedades y métodos de la clase que hay disponibles. El método **IsInitialized** comprueba si el objeto está inicializado y existe en todas las clase que has creado.

## Ejercicios

1. Siguiendo el primer ejemplo del tema, implementa la clase estudiante en B4J con las siguientes propiedades:

* Identificador escolar
* Nombre
* Apellidos
* Clase
* Teléfono
* Correo electrónico

Y los métodos:

* Nuevo estudiante
* Mostrar estudiante
* Cambiar estudiante
* Cambiar teléfono

1. Supongamos que un profesor sólo imparte una materia en un colegio. Implementa la clase Curso con las siguientes propiedades:
   1. Tema
   2. Clase
   3. Horas
   4. Profesor

Y los métodos:

1. Nuevo Curso
2. Cambiar Horas
3. Cambiar Profesor
4. Mostrar Curso
5. Una tienda vende ordenadores. De cada uno guarda lo siguiente:
   1. El tipo (sobremesa, portátil)
   2. El modelo
   3. El precio
   4. Su CPU (i3, i5, i7, i9)

Crea una clase que implemente un ordenador con las propiedades anteriores y los métodos nuevoOrdenador, mostrarOrdenador, cambiarCPU y cambiarPrecio. Comprueba que los valores introducidos en CPU y en Tipo son los que aparecen entre paréntesis.