I Logotipo

Descripción generada automáticamente con confianza baja

**Servicio Nacional de Aprendizaje – SENA**

**Centro de Diseño y Metrología**

**InformeBiblioteca**

**Aprendiz:**

**Juan Esteban Castro Escamilla**

**No Ficha 2848530-A**

# Tabla Contenido

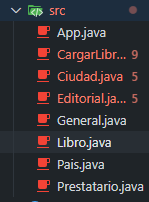
[**Tabla Contenido 2**](#_es6cyuysmcm8)

[**Biblioteca Agregación y Composición JAVA 3**](#_f3jt0t2tem82)

# **Biblioteca Agregación y Composición JAVA**

## **Composicion:**

(Bilbioteca se compone principalmente de estas clases: CargarLibro, Ciudad, Editorial, General, Libro, Pais, Prestatarios)

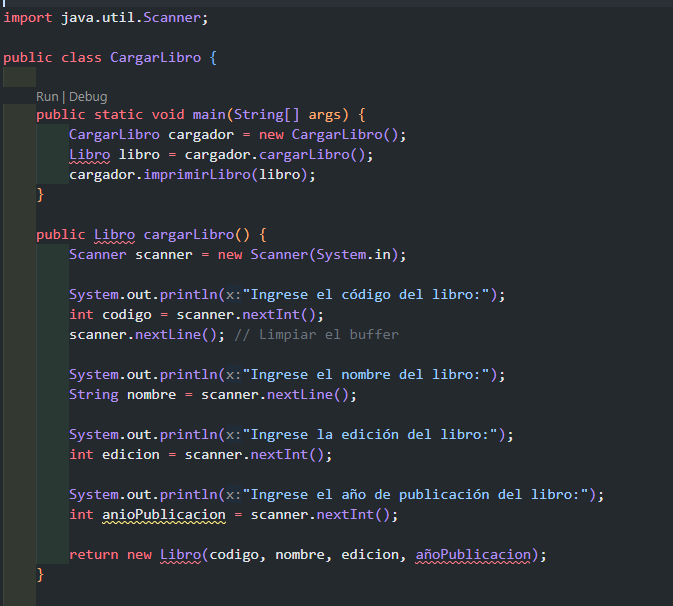


### Clase CargarLibro:

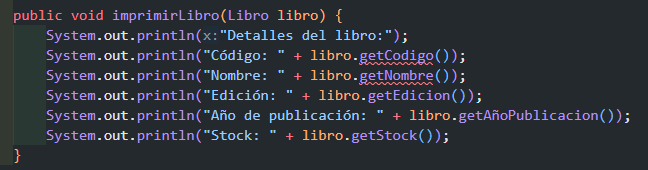
(Se utiliza la utilidad Scanner para la asignación de los datos)

(Este clase CargarLibro contiene una función para hacer la respectiva carga del mismo y el libro ya cargado y la impresión de este)

(Contamos con la función de CargarLibro la cual compone a un scanner en el cual, dentro de este scanner se guardaran los datos que se le pida por pantalla o terminal al usuario, en este caso ingresar el código, nombre, edición y año de publicación del libro y asi mismo retornar dichos valores)



(Finalmente concluye con la función imprimirLibro la cual dentro de esta estara el System out es decir aquellos que permiten mostrar por pantalla o terminal los datos que se asignaron previamente es decir, los detalles del libro)

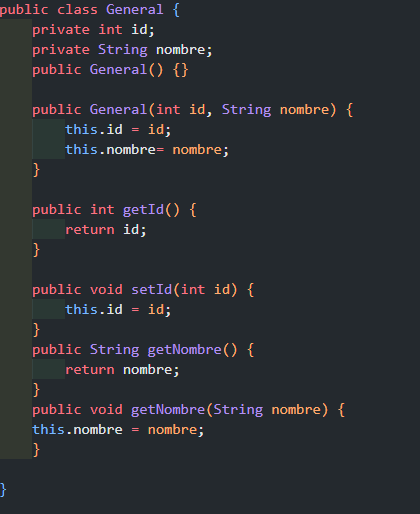


### Clase General:

(Ahora se crea la clase General la cual contendrá alguno datos del prestatario tales como su nombre y un id para identificarlo dentro del sistema)

(Identificamos un constructor para obtener precisamente el id y nombre de los atributos previamente asignados)

(Creamos los respectivos Getter and Setters para cada uno)

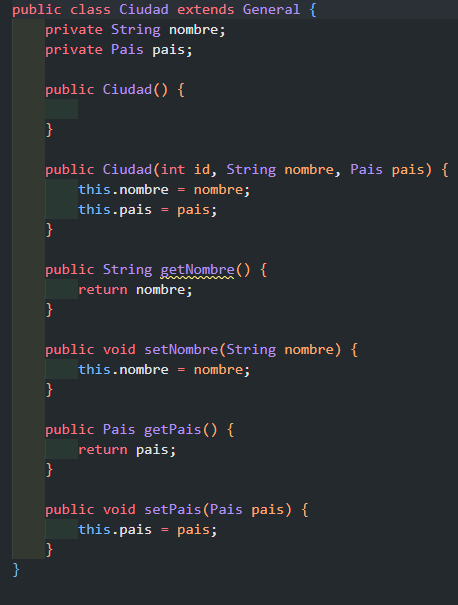


### Clase Ciudad:

(Creación de la clase Ciudad la cual hereda los atributos de la CLASS GENERAL así mismo esta clase contendrá un nombre y el respectivo país)

(Contiene un constructor vacío y otro con parámetros los cuales se especifican con el this para obtener precisamente dichos atributos)

(Claramente se crea los Getter And Setter para cada uno de los atributos creados dentro de la clase)



### Clase Pais:

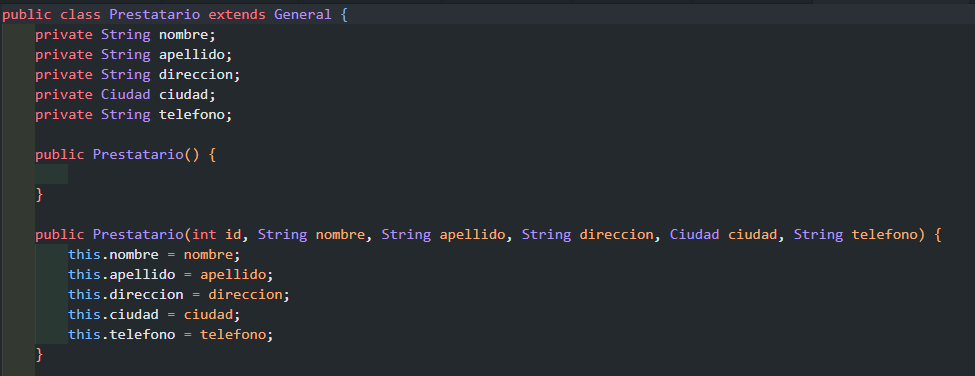
(Creación respectiva del País la cual tendrá el nombre de dicho país y sus Setter And Getter, realmente aquí no hay tanta explicación ya que no contiene nada extraño nuevo a parte de que hereda con el extends la clase General)

### 

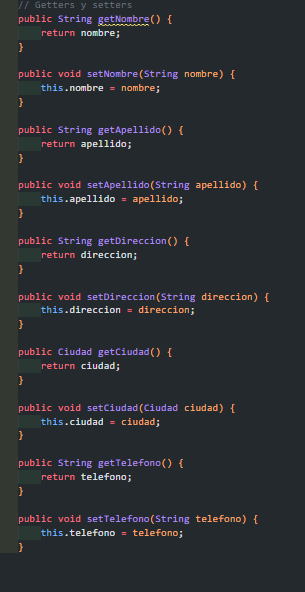
### Clase Prestatario:

(Creación de la clase Prestatarios es decir dentro de esta clase se contendrá la información de la persona\_Usuario que pedirá un libro en específico)

(Hereda los los atributos de general y se crean los respectivos constructores)



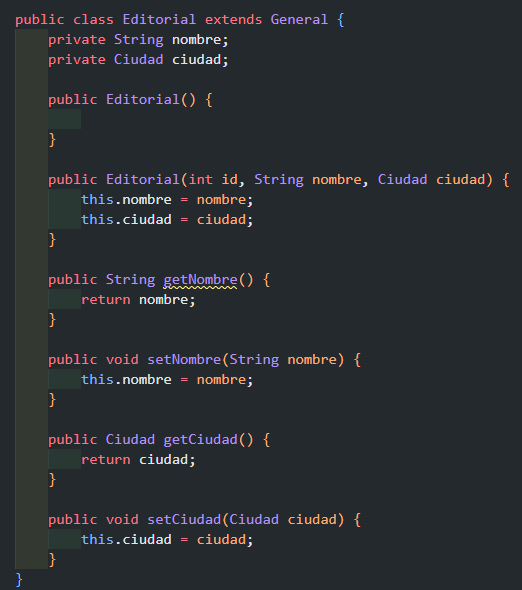
(Sus respectivos Getter And Setter)

****

### Clase Editorial:

(Creamos una clase para la respectiva Editorial con su nombre de la misma y ciudad)

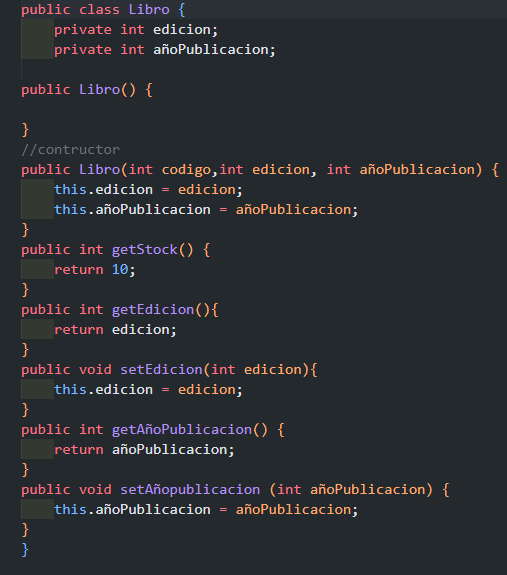
(Asi mismo sus constructores y setter and getter que ya se ha visto previamente)

****

### Clase Libro:

(Creación de la clase Libro para obtener la edición de este y su año de publicación)

(Creamos los respectivos constructores y Setter and Getters, también agregamos un Get extra para obtener el Stock o cantidad disponible para la prestación de dicho libro)

****

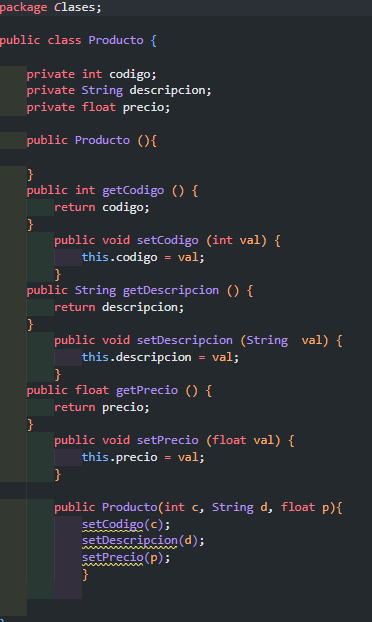
## **Agregación Biblioteca:**

(Biblioteca a parte de las composiciones principales que esta contiene, también debe tener o conseguir daños extras que ayudarán en el buen funcionamiento del sistema de la biblioteca para esto, se ha creado más clases que agregaran y funcionarán junto a lo que se compone de una biblioteca)

(El libro se puede catalogar como producto, por ende, tendrá sus respectivos atributos como producto, así mismo, este tendrá una factura, comprobante, cliente y fecha para poder guardar dichos datos en el sistema y saber con precisión acerca de la prestación de dicho libro o libros)

### Clase Producto:

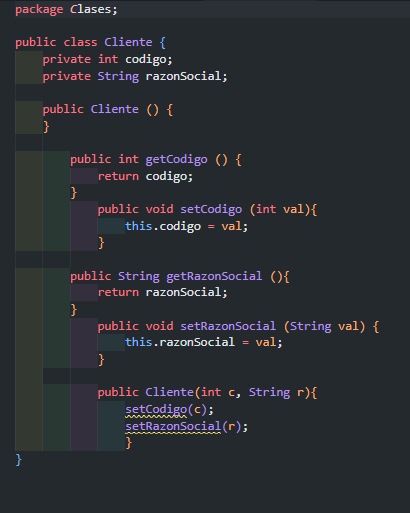
(Creacion de la clase producto la con contiene el codigo, descripcion y precio de este, así mismo sus respectivos constructores y sus getter and setter)



### Clase Cliente:

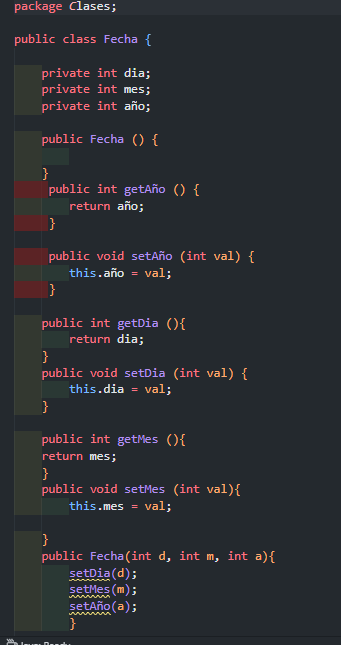
(Encontramos una clase para los diferentes clientes que se podrán encontrar en la biblioteca y necesitamos guardar dicha información, para ello, se realizó el atributo código y razón social)

(Contiene de igual manera sus constructores y setter and getter)



### Clase Fecha:

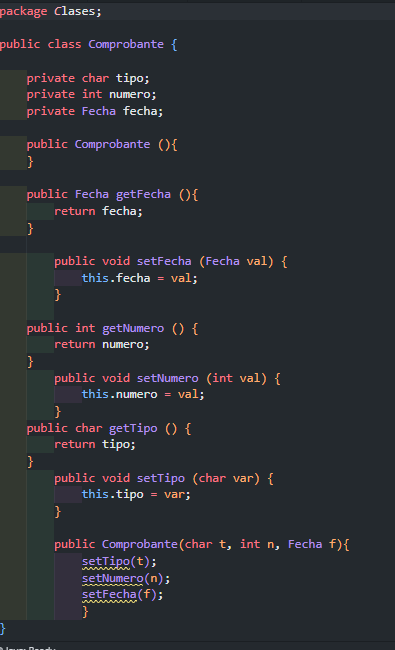
(Contiene esta clase Fecha el día, mes y año exacto con todo los respectivo de una clase)



### Clase Comprobante:

(Contenemos los datos del comprobante, en este caso, su tipo, número y fecha de la realización de este)

(A si mismo sus respectivos constructores y set and get)

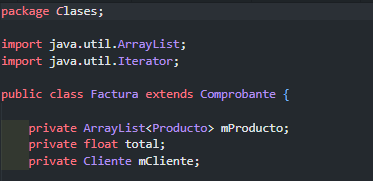


### Clase Factura:

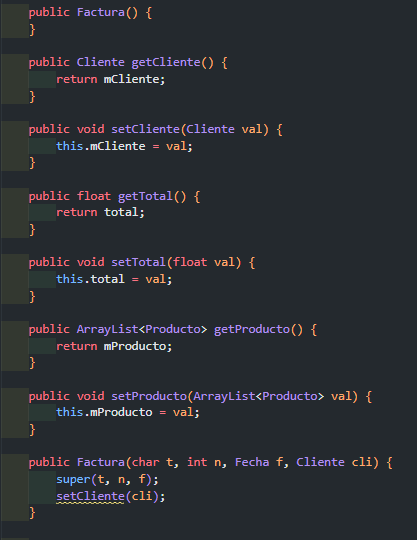
(Encontramos las utilidades ArrayList e Iterator que se utilizarán a continuación)

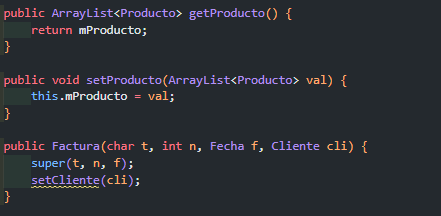
(Clase Factura contendrá atributos heredados del Comprobante y asi mismo contendrá un Arreglo el cual contendrá el Producto y su nombre será mProducto para su utilización)

(Así mismo tendrá un total y un objeto que se ha sacado de la clase Cliente y se llamará mCliente)



(Sus respectivos constructores y getter and setter)

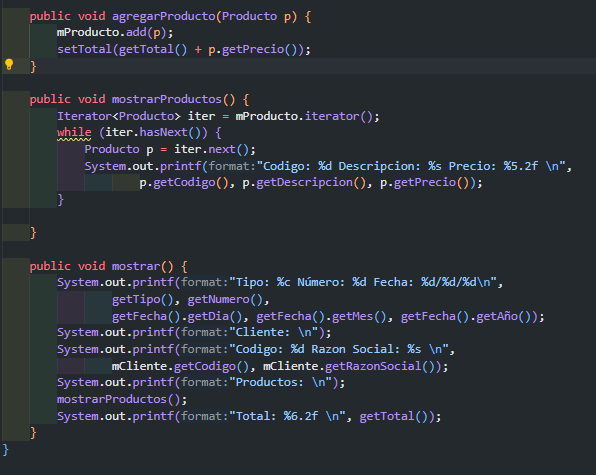




(Creamos 3 funciones, una para agregar el producto dentro de arreglo y así mismo calcular el precio total)

(otra para mostrar el producto con la función iterator y asi mismo un bucle while que hasta que que termine de recorrer cada uno de los productos con su detalle de codigo, descripcion y precio)

(Finalmente la 3 función mostrata cada uno de los datos asignados previamente de las diferentes clases)



(Se hace la respectiva prueba en nuestro main Java donde vamos a crear un objeto de cada unas de las clases creadas y vamos a agregar diferentes valores y así mismo utilizar las funciones para que estos se agreguen y se muestren como debe de ser)

