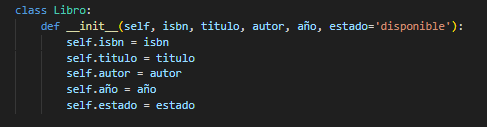
Gestor de Biblioteca

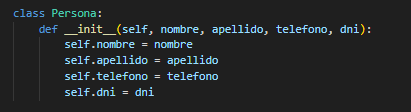
El programa desarrollado tiene como objetivo simular un gestor de biblioteca. Incluye diversas funcionalidades, como la gestión de libros, donde es posible agregar, actualizar y eliminar títulos; la gestión de clientes, que permite registrar nuevos usuarios y visualizar la lista de clientes registrados; y la gestión de préstamos, que facilita realizar préstamos y devoluciones de libros. Finalmente, cuenta con un apartado de reportes, en el que se puede consultar los libros prestados, disponibles y los clientes más frecuentes.

# clases y métodos del programa

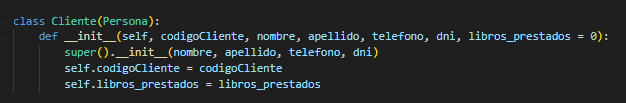
**Clase libro:** la clase se ha creado con los atributos de ISBN, titulo, autor, año y estado. El ultimo atributo denominado “estado” siempre se va agregar con el dato de “disponible” para cada objeto de la clase libro.



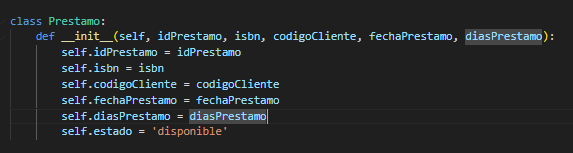
**Clase persona:** La clase se ha creado con los atributos de nombre, apellido, teléfono y DNI.



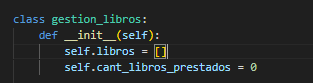
**Clase cliente:** La clase cliente cuenta con dos atributos únicos los cuales son codigoCliente y librosPrestados, donde este segundo atributo siempre registrara al objeto con el valor de 0. Los atributos de nombre, apellido, teléfono y DNI, son atributos que va heredar de la clase de Persona.



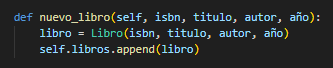
**Clase préstamo:** la clase fue creada con los atributos de idPresamo, isbn, codigoCliente, fechaPrestamo, diasPrestamo y un atributo adicional denominado estado.



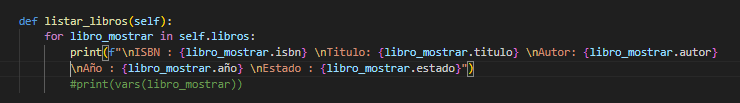
**clase gestión de libros:** Esta clase cuenta con dos atributos, el primero es denominado “libros” este comienza como una lista vacía y el segundo es denominado “cant\_libros\_prestados” que sirve para conocer la cantidad de libros prestados.



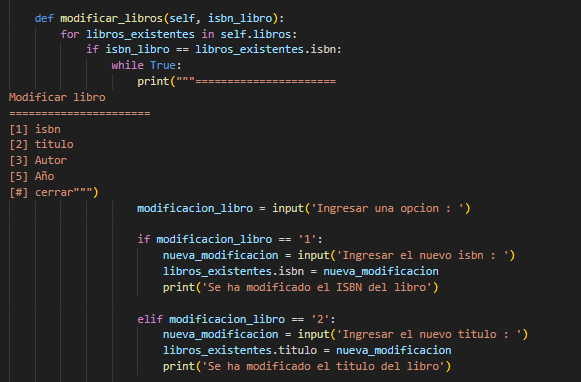
**método nuevo libro:** Registra un nuevo libro creándolo en un objeto con la clase libro, el objeto se almacena dentro de la lista denominada libros.



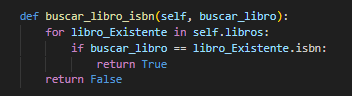
**Método listar libro:** Recorre un bucle for que tiene como rango la lista libros, con la finalidad de mostrar cada objeto creado de la clase libro como si de una lista se tratase.



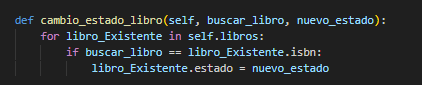
**Método modificar libro:** Este método primero va buscar la existencia de un libro a través de un bucle for y una sentencia condicional, donde si el libro es encontrado se abrirá un submenú en el cual se van a poder realizar los cambios en el libro seleccionado.



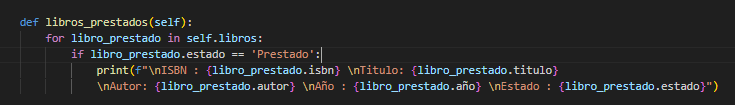
**Método buscar libro isbn:** Se utiliza para poder encontrar la existencia de un libro a través de su dato de isbn, para ello recorre un bucle for con la lista de libros, luego con una sentencia condicional compara si el isbn ingresado coincide con algún isbn registrado, y en caso de encontrar el libro retornara un valor booleano.



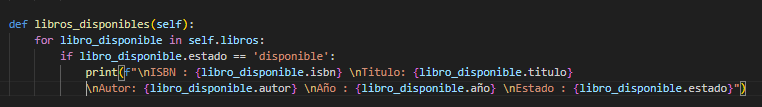
**Método cambiar estado de libro:** Se utiliza al argumento de buscar libro, el cual contiene el dato del isbn, este dato lo va comparar con la variable de libro\_existente del bucle for, ambas variables se van a comparar en una sentencia condicional donde sí coinciden utilizara al argumento de “nuevo\_estado” para actualizar el estado de un libro.



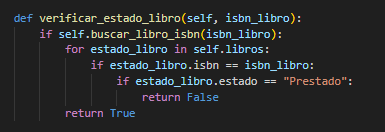
**Método libros prestados:** recorre un bucle for con la lista de libros, donde en una sentencia condicional se va comparar el estado del libro. si el estado del libro es “prestado” entonces se mostrará en consola.



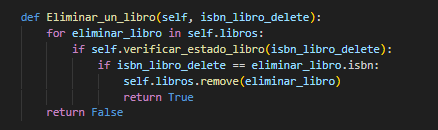
**Método libros disponibles:** recorre un bucle for con la lista de libros, donde en una sentencia condicional se va comparar el estado del libro. si el estado del libro es “disponible” entonces se mostrará en consola.



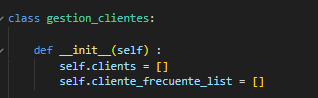
**Método verificar estado de un libro** El método verifica si el libro ha sido prestado. primero con una sentencia condicionan se va buscar la existencia del libro con el método de buscar\_libro\_isbn, segundo, tras encontrar el libro recorre un bucle for donde va utilizar el valor de isbn ingresado por el usuario y que ya fue verificado, dentro del bucle hay una sentencia condicional el cual va buscar el libro con el isbn, tercero entrara a una sentencia condicional donde si el estado del libro es “prestado” va retornar el valor de False, de no ser así retornara el valor de True.



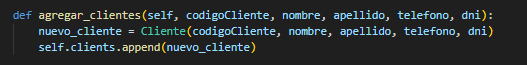
**Método Eliminar libro:** el método tiene como finalidad eliminar un libro utilizando el isbn ingresado por el usuario, el libro y todos los datos del libro se van a remover de la lista de libros.



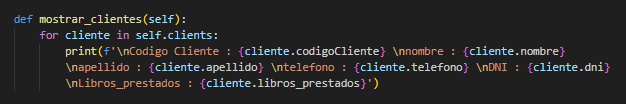
**Clase gestión clientes:** La clase cuenta con dos atributos los cuales comienzan siendo listas vacías, estos son: clients y cliente\_frecuente\_list.



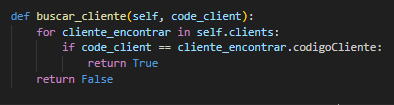
**Método agregar clientes:** A través de este método se crean objetos de la clase cliente que son almacenados en una variable para luego esta variable sea agregada a la lista de clients.



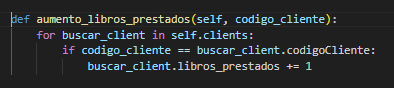
**Método mostrar clientes:** El método recorre un bucle for el cual va mostrar los datos de cada cliente en la consola que se han agregado en la lista clients



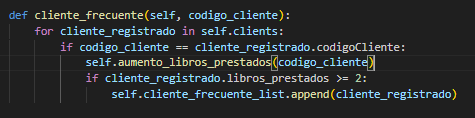
**Método buscar cliente:** el método se utiliza para poder buscar un cliente a través de su código de cliente, donde sí se encuentra el valor que se retorna es True y si no se encuentra el valor que se retorna es False.



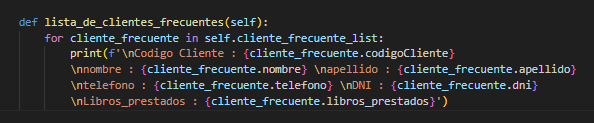
**Método aumento de libros prestados:** utilizando el código de cliente se busca al cliente a través de un bucle for y una sentencia condicional donde se comparan los datos, al encontrar al cliente se le va aumentar en 1 la cantidad de su atributo denominado libros prestados, que sirve para conocer cuántos libros a prestado de la biblioteca.



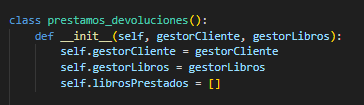
**Método cliente frecuente:** Utilizando el argumento de código\_cliente se va buscar al cliente a través de un bucle for y una sentencia condicional donde se realiza una comparación, al encontrar al cliente se llama al método aumentar\_libros\_prestados para aumentar el valor del atributo de libros\_prestados, después, con una sentencia condicional se verifica si el atributo de libros\_prestados es mayor o igual a 2, de ser así se agrega el objeto de cliente a la lista de clientes frecuentes.



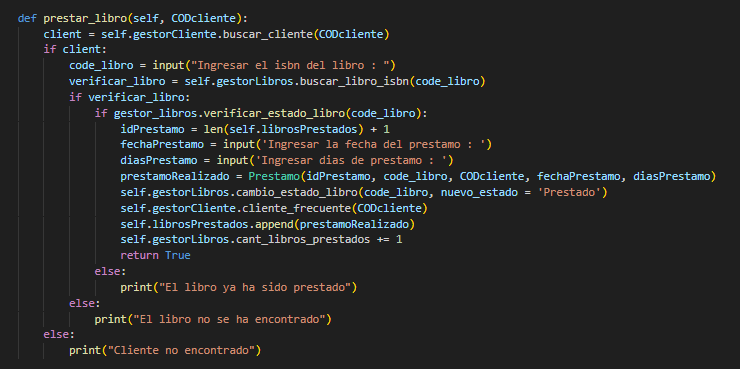
**Método lista de clientes frecuentes:** el método va mostrar los datos de los clientes que se han agregado como frecuentes.



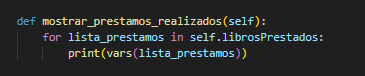
**clase préstamos y devoluciones:** La clase fue creada con 3 atributos los cuales son gestor cliente y gestor libros, estos dos atributos toman como referencia a las dos clases anteriores las cuales van a permitir que se puedan acceder a los atributos y métodos de esas clases dentro de esta misma clase. También cuenta con un tercer atributo el cual es denominado como libros prestados e inicia como una lista vacía, aquí se van a almacenar la información de los libros prestados.



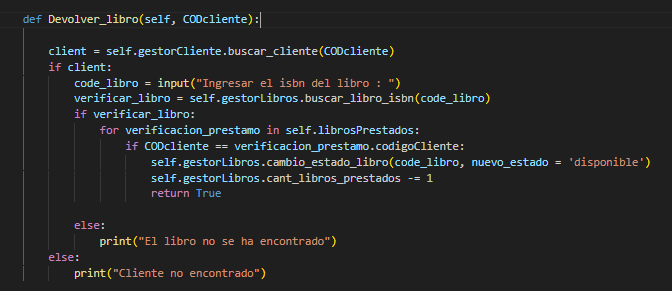
**Método prestar libro:** Este método cuenta con la finalidad de realizar el préstamo de un libro a un usuario, creando un objeto de la clase de préstamos y almacenando la información en la lista de librosPrestados. comienza llamando al método de la clase de gestor cliente denominado buscar cliente, almacena la información booleana dentro de una variable que será utilizada en una sentencia condicional, al confirmar la existencia del usuario realiza lo mismo, pero con el isbn del libro, esto para verificar la existencia de un libro. Cuando ya se cumplan ambas condiciones solicitara 2 datos extra que serán necesarios para formar el objeto de la clase prestamos, la información de la clase se almacenara en una variable y se va a agregar a la lista llamada librosPrestados, luego de esto realizara el cambio del estado del libro seleccionado, utilizando el método de la clase libro, también cambiara el dato de cliente frecuente utilizando el método de la clase cliente y aumentara en uno la información en libros prestados. finalmente retornara el valor booleano de True para confirmar que se realizó el préstamo correctamente.



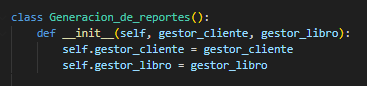
**Método mostrar prestamos Realizados:** El método consiste en mostrar los datos de cada objeto almacenado dentro de la lista de librosPrestados, esto lo va realizar de manera de una lista visual en la consola.



**Método devolver libro:** similar al método de prestar libro, este método primero confirma la existencia de un usuario y del libro prestado, luego utilizando el bucle for para busca con el código cliente a un cliente registrado en la lista de librosPrestados, al confirmar la existencia del cliente cambia el estado del libro a disponible y disminuye el valor del atributo de cant\_libros\_prestados en uno, finalmente regresa el valor booleano del True para confirmar la devolución del libro.



**Clases generación de reporte:** la clase cuenta con los atributos de gestor clientes y gestor libros, ambos atributos hacen referencia a las clases de gestor libros y gestor clientes, y permitirá el uso de los métodos y atributos de esas clases dentro de esta misma clase.



**Método Libros prestados:** Se llama al método de la clase de libros prestados en cual mostrara en consola todos los libros que tengan como estado “prestado”.



**Método clientes frecuentes:** Se llama al método de lista\_de\_clientes\_frecuentes de la clase de clientes para poder mostrar en la consola una lista de todos los clientes que se han registrado como frecuentes.



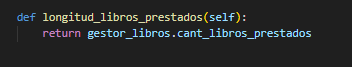
**Método Libros Disponibles:** Se llama al método de la clase de libros disponibles en cual mostrara en consola todos los libros que tengan como estado “disponible”.



**Método longitud de clientes frecuentes:** este método llama a la lista de clientes frecuentes para poder obtener su longitud en un dato numérico.

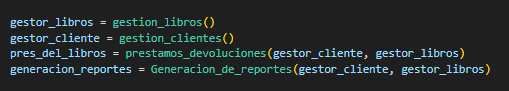


**Método longitud de libros prestados:** este método llama al atributo de libros prestados para poder obtener su dato numérico.

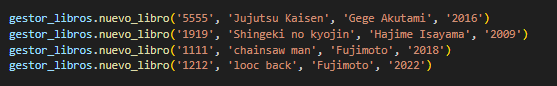


## Creación de objetos e ingreso de datos

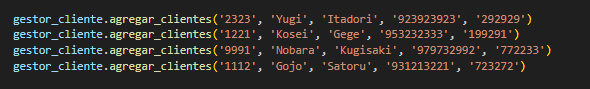
**Creación de objetos:** se crea un objeto para clase el cual va a permitir el uso de los métodos durante el funcionamiento del programa. El objeto de la clase libro y clientes, serán utilizados como argumentos para la clase de prestamos\_devoluviones y generación\_de\_reportes.



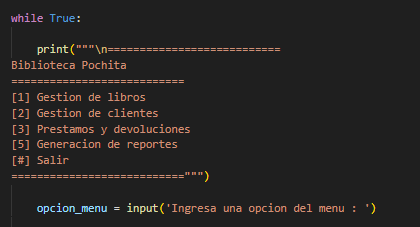
**Ingreso posterior de libros:** a través del objeto de la clase de gestión libro, se realiza el ingreso de datos de cuatro libros.



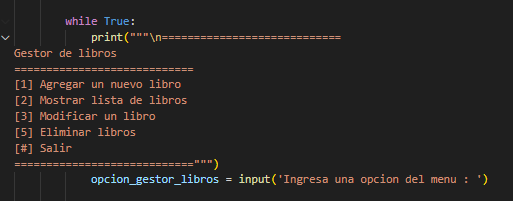
**Ingreso posterior de clientes:** a través del método de agregar\_cliente de la clase de gestión cliente, se realiza el ingreso de clientes antes de que inicie el programa.



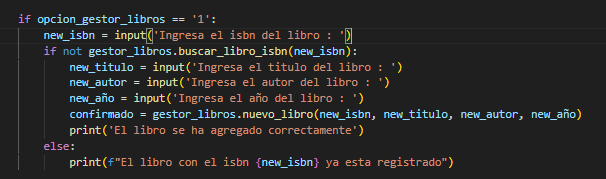
**Menú principal:** el programa mostrara el menú principal con el cual se van a poder acceder a los demás submenús según la opción que se seleccione, las opciones disponibles son: gestión de libros, gestión de clientes, préstamos y devoluciones, y Generación de reportes, la última opción es para finalizar el programa. se puede acceder a cada submenú accediendo desde un dato obtenido en consola.



**Menú gestión de libros:** Se iniciare un nuevo bucle el cual va mostrar indefinidamente el menú de gestión libros hasta que se cierre el bucle:



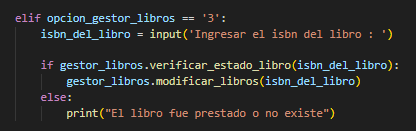
* **Agregar nuevo libro:** solicitara los datos necesarios para poder agregar un nuevo libro, donde primero va buscar la existencia de un libro con el isbn en caso de encontrar el isbn se cancela el nuevo ingreso.



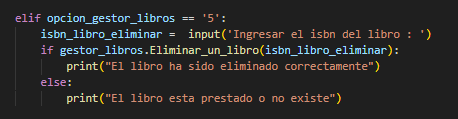
* **Mostrar lista de libros:** mostrara los libros registrados en la consola



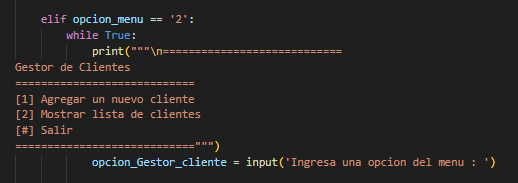
* **Modificar un libro:** Al confirmar la existencia del libro con el isbn se abrirá un submenú que actualizará los datos del libro.



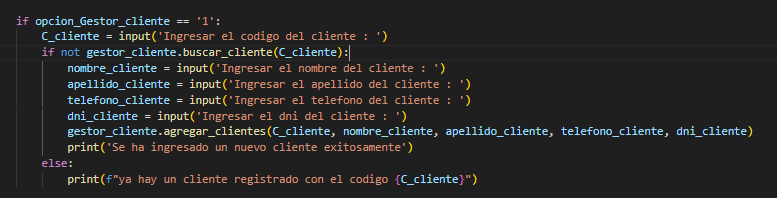
* **Eliminar libro:** Al confirmar la disponibilidad de un libro se procederá a eliminar el libro de la lista de libros.



**Menú gestión clientes:** Se iniciare un nuevo bucle el cual va mostrar indefinidamente el menú de gestión clientes hasta que se cierre el bucle:



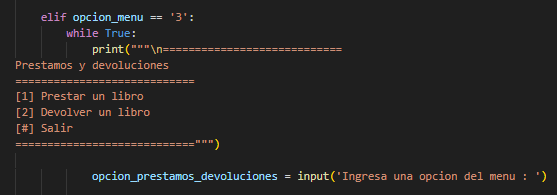
* **Agregar nuevo cliente:** se solicitará una serie de datos que serán utilizados para la poder agregar a un nuevo cliente a la lista de clientes. primero verificará la existencia del cliente a través del código de cliente, donde sí se encuentra registrado se cancelará el nuevo ingreso.



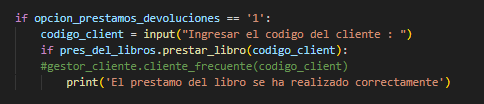
* **Mostrar lista de clientes:** Mostrara una lista de clientes registrados en la consola.



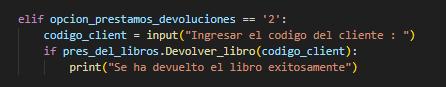
**Menú préstamos y devoluciones:** Se iniciare un nuevo bucle el cual va mostrar indefinidamente el menú de préstamos y devoluciones hasta que se cierre el bucle:



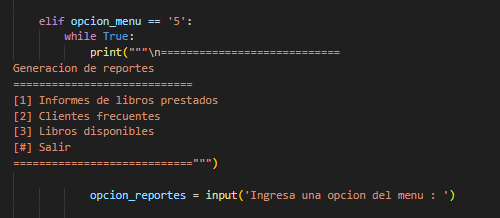
* **Prestar libro:** La opción es con la finalidad de poder prestarle un libro a un usuario, donde principalmente debe de buscar la existencia del usuario a través del código de cliente, donde sí se encuentre se procederá al siguiente ingreso de datos hasta registrar el préstamo.



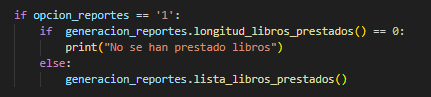
* **Devolver un libro:** La opción es con la finalidad de poder devolver un libro el cual ha sido prestado, donde primero debe de buscar la existencia del cliente a través de su código de cliente, al confirmar que el cliente ha prestado un libro se solicitara el ingreso del dato del libro y se confirmara la devolución.



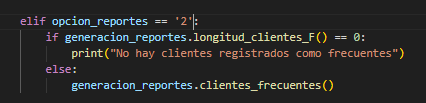
**Menú Generación de reportes:** Se iniciare un nuevo bucle el cual va mostrar indefinidamente el menú de generación de reportes hasta que se cierre el bucle:



* **Informe de libros prestados:** Se solicitará la longitud de los libros prestados donde si la respuesta es cero se mandara un mensaje de no hay registros, pero de no ser así mostrara una lista de los libros prestados en la consola.



* **Clientes frecuentes:** Solicitara el dato de la longitud de clientes frecuentes donde si es cero se mandara un mensaje de no hay clientes frecuentes, pero de no ser así mostrara una lista con los clientes frecuentes registrados.



* **Libros disponibles:** mostrara en la consola una lista con todos los libros que tenga como estado “disponible”.



**Resumen:**

**Clase gestión libros:**

* **Método nuevo libro**: Registra un nuevo libro.
* **Método listar libros**: Muestra una lista de todos los libros registrados.
* **Método modificar libros**: Abre un submenú para modificar los datos de un libro.
* **Método buscar libro isbn**: Busca la existencia de un libro a través de un isbn.
* **Método cambiar estado de libro**: Cambia el estado de un libro
* **Método libro prestado**: Muestra una lista de los libros en la consola que solo tengan como estado “prestado”
* **Método libro disponible**: Muestra una lista de los libros en la consola que solo tengan como estado “disponible”
* **Método verificar estado de libro:** Verifica cual es el estado actual de un libro
* **Método eliminar un libro:** Elimina un libro registrado.

**Clase gestión clientes:**

* **Método agregar clientes:** Registra a un nuevo cliente.
* **Método mostrar clientes:** Muestra en una lista en la consola todos los clientes registrados.
* **Método buscar cliente:** Busca la existencia de un cliente a través del código de cliente.
* **Método aumentar libro prestado:** aumenta en uno el atributo de libros prestados de un cliente.
* **Método cliente frecuente:** Registra a un cliente como frecuente si este tiene más de dos libros prestados.
* **Método lista de clientes frecuentes:** Muestra en la consola una lista de todos los clientes registrados como frecuentes.

**Clase préstamos y devoluciones:**

* **Método prestar libro:** Presta un libro a un usuario registrado en la biblioteca.
* **Método mostrar prestamos realizados:** Muestra una lista en la consola con todos los libros que se han registrado como prestados.
* **Método devolver libro:** Devuelve un libro que se ha prestado por el cliente.

**Clase gestión de reportes:**

* **Método lista de libros prestados:** muestra en la consola una lista con todos los libros que tengan como estado “Prestado”.
* **Método clientes frecuentes:** Muestra una lista en la consola con todos los clientes registrados como frecuentes.
* **Método lista de libros disponibles:** Muestra en la consola una lista con todos los libros que tengan como estado “disponible”.
* **Método longitud de clientes frecuentes:** Obtiene el valor numérico de los clientes registrados como frecuentes.
* **Método longitud de libros prestados:** Obtiene en valor numérico de la cantidad de libros que fueron prestados.