**Diplomatura\_TP**

[Repositorio](https://github.com/RomanAdriel/BibliotecaNivelIntermedio) del TP de la diplomatura en Python

Integrantes:

* Román Adriel Díaz
* Gerardo Agustin Cabrera

TP: Biblioteca

Alcance:

Diseñar un CRUD de biblioteca que permita a un usuario consultar, agregar, actualizar o borrar un libro de la base de datos de libros. Para cada evento, se mostrará en la terminal un mensaje que indique el resultado de la operación.

El formato del programa sigue el patrón MVC. Cada módulo se encuentra documentado con su respectivo archivo de documentación HTML generado con pydoc.

# La tabla de libros contendrá los siguientes datos:

* idlibros (autoincremental) (int)
* isbn (varchar(45))
* titulo (varchar(250))
* autor (varchar(250))
* ano\_publicacion (int)
* editorial (varchar(45))
* idioma (varchar(45))
* formato (varchar(45))
* stock (int)
* creado\_por (varchar(250))
* hora\_creacion (timestamp)
* modificado\_por (varchar(250))
* hora\_modificacion (timestamp)

# Interfaz con el usuario del programa:

* Login (solo ingreso de mail y validación de formato(regex))
  + Menu principal
    - Alta de libro
    - Modificación de libro
    - Eliminación de libro
    - Consulta de libro
    - Cambiar de cuenta
    - Salir
  + Notificación de resultado de operación

# Reseña del programa:

El programa inicializa en el módulo ***controlador***, levantando la base de datos (en caso de existir, hace un DROP y la crea) y luego llamando a los métodos de interacción con el usuario.

El usuario va a tener en el menú la posibilidad de realizar las actividades típicas de un CRUD, recibiendo un feedback por parte del programa en estas acciones.

Como agregado, el usuario puede "cambiar de cuenta" durante la ejecución del programa. Lo que hace esta opción es básicamente cambiar el mail y el usuario por uno nuevo, cosa que puede verse reflejada cuando se hace una modificación o inserción de algún libro y en el mensaje de despedida al elegir "Salir".

Una vez el programa deje de ejecutarse, la información permanecerá en base de datos hasta la próxima ejecución en que nuevamente se hará un DROP. Por otra parte, el usuario podrá descargar un archivo .csv con la información al momento de haber sido consultada, que se guardará en la carpeta "archivos" dentro del proyecto.

# Happy Path:

1. Inicializar el programa
2. Ingresar un correo (tiene 3 intentos para hacerlo correctamente, en caso contrario el programa terminará)
3. En el menú, dar de alta un libro
4. Cambiar de cuenta con un mail válido (nuevamente, tendrá 3 intentos. En caso contrario seguirá con la misma cuenta)
5. Consultar el libro ingresado
6. Modificar el libro (utilizar nombres de columna válidos)
7. Volver a consultar el libro (se podrá ver que el timestamp de la creación difiere de la modificación, así como los usuarios)
8. Salir