**Book recommendation system**

**Developed by TEAMENT:**

**Gwendal HOLLOCOU**

**Tao SAINT PAUL AMOURDAM**



**Years 2021-2022**

*Introduction*

The aim of the project “book recommendation system” is to create a numeric tool that will suggest books to readers based on their profiles and previous reading.

Users can register the database, enter what books they have read and get new books recommended for them.

The recommendation algorithm is based on a mathematic formula that computes the similarity between readers.

There is no GUI (graphic user interface), we use the terminal as support.

And because we know terminal is not always easy to understand, we made it explicit and clear so that anyone can use the “book recommendation system”.

Firstly, we decided to put ourself 3 main goals that will be our red line:

* We wanted to have **1 file by part**. So, there are 3 python files (.py) that contain the 3 parts of the project (readers part, books part, recommendation part). And one main.py python file, managing the run of the three other ones.
* We also wanted to have a **clear interface**. At first, we wanted to make a GUI but we realized that we had done a huge work on making the terminal interface shapely. So, we decided to continue the terminal interface and make it even more clear.
* And finally, we wanted the user to have the possibility to **back or exit** the program at any time.

*Exit & main system*

In order to be able to exit the program, all along the program, **every function with an input returns a variable that we called “power”.**

It is what the user will input that will determine the power returned by the function ; if the user just execute the functions, they will always return 1,

but in case they try to exit they will return 0.

If the function returns a power equal to 1 the program keeps going.

Otherwise, if power returned is equal to 0, all the functions that called this function return power = 0 as well until the main program that check if power == 0. If it is, it displays “Hope you enjoyed! Bye!” and the program ends.

We started developing by the **main file**.

This file, (after importing every other files) starts by running the main program (red framed part), and this while power == 0 (blue framed part), meaning that the main.py will keep running until the user try to exit.

This program permits the user to choose between the functions of the 3 parts he wants to access.

He has to type 1 to access part 1 functions, 2 for part 2, etc.

Since we are at the root of the program, the user can’t back here.

If he tries, it displays "you cannot go back from here".

Une image contenant texte

Description générée automatiquement

*Parts 1, 2 & 3*

These files are ruled by menus, each of them has one. They kind of work the same way, here is for example part 2’s menu:

You can see we print every choices the user has, then get what he chose to do.

Every menus has the two same “if” blocks :

* One to handle the different functions based on the input given by the user (the blue framed one),
* A second one to handle the returned power by the function that was run (the red framed one).

As we just saw in page 3, **return 1 will keep running the main.py file, while return 0 will exit the program.**

Une image contenant texte

Description générée automatiquement

*What’s inside ?*

Now that we understood how the program is ruled and managed, we will look at the program in a more concrete view : what does it actually do.