The project Bookworm is a collection of 5 separate classes: Entertainment, Entertainment Book, Entertainment Movie, Entertainment Play and a final Bookworm Exception class

These classes are in their individual .py files and the main class: Entertainment gets initialized and called through the main.py file which is a simple def program that initializes the Entertainment class.

The entertainment class has a call to the main function within the program which requests user input entry of either book, movie or quit.
The program will quit only on the entry of word: quit

As soon as the user selects from a book or movie, the function: entertainment_main gets called
Which in turn calls either the Entertainment Book or Entertainment Movie class depending on user input of a book or movie. If the user enters a value other than book or movie (non-case-sensitive) the user will be prompted to enter again.

When the user selects Book - they will be asked if they want to check out fiction or non-fiction books. Again case insensitive using a tuple to select values, but the values have to be one of the two.
Then they will be asked to enter a title and then asked if they want to enter a published date - both non-mandatory fields.

The program then calls the books google api to find 10 relevant books related to the entered values.

The user will then be prompted to select 1 of the books or asked if they want to continue checking out book.
If they select to continue checking out books, the entire job will run from selection of fiction or non-fiction.

Once the user selects the book they find interesting, this will be entered in a booklist.json file - which will be used to check against a movie list json file to print related movies if any.
The book details will also get pushed to a booklist text file to be displayed at the end along with any related movies.

The Entertainment Movie class will similarly print 10 movies and let you select 1 movie - and store it in a json file to get the relevant books.

The Entertainment Play class will make both these modules interact with each other in the sense that if a book has been selected - the program will use the value in booklist.json file against the movies json to get any related movies and input into the booklist text file.
Which will display both the book and related movies.

Similarly if a movie has been selected earlier by the user the Play class will search for related books using the google api and plug into the movielist text file, which will print the data in a structured way.

Steps to run the program:

1) Open terminal window
2) Navigate to the folder Project1_Bookworm location
3) Python main.py like:
   Sonals-MacBook-Air:Project1_Bookworm sonalthakkar$ python main.py

4) Enter the options desired: book or movie
5) In case book is selected the user will get further choices as described above.
6) For the movie selection a pre-entered json list gets called as described above.
7) The user can go through the selection process as many times as desired.
8) Every time a selection is made for a book or a movie, related choices are saved for the other other (if a book is selected by the user, relevant movies based on the book for example, or in case of a movie selection, related books that might be of interest to the user might get saved)
9) At the end of the user selection a list gets printed with the user selection and relevant choices of the book/movie.
10) The user can quit at any point by typing ‘quit’.