COMP 3100 – Web Programming
Assignment 1
Winter 2021 (Weeks 1 and 2)

* This assignment has the objective of training you for your final project. This application has several limitations to be used as a real-life application. At this stage I just want you to practice your JavaScript and Node.js skills.
* Please document your code to assist us in marking your assignment. A poor job in documentation will impact in your grade.
* The bonus challenge can only be moved between your assignments. For example, if you score 105 in this one, and 95 in assignment 2, you will end up with 100 in assignment 1 and 100 in assignment 2.

In this assignment you should create an API to handle data from a Public Library. Your API should handle two entities with the following information:

- Book.
    - **id**: The book's unique identifier.
    - **name**: The book's name.
    - **authors**: The authors' names.
    - **year**: The book's publication year.
    - **publisher**: The institution that published the book.
- Loan.
    - **id**: The loan's unique identifier.
    - **book id**: A reference to the book's id that was on loan.
    - **date**: The date of the loan.
    - **client name**: The name of the client who borrowed the book.
    - **was returned**: The information if the book was returned.
    - **date of return**: The date that the book was returned.

* You will receive 10 marks if your objects are correctly designed.

There are several requisites that your implementation should maintain and they are listed below.

- The data should be stored and handled in your application as a JSON.
- The information should be saved in a text file (.json format).
- When you start your server, the system should be able to load the data in the file(s) managing the books and loans (This is different from the example seen in Week 2, where we delete the file every time we restart the service).
- All get operations should send an outcome a JSON object.

* If you follow all these requirements, you will receive 15 marks.

Your API must provide the following operations.

- Add a new book.
- Retrieve the information of a book by id.
- List all books, providing the book id and its name.
- List all books by a given year with all its information.
- Verify if a book is available.
- Add a loan.
- List all loans that were finished (the book was returned).
- List all loans that are open (the book was not returned).
- Update a loan. This operation should control if the book was returned or not.

* You will receive 5 marks for each operation correctly implemented in this section, totaling 45 marks.

You should write your own tests for all these operations.

* You will receive 3.3 marks for each operation correctly tested in this section, totaling 30 marks.

Bonus challenge (5 marks): Validate all the fields for the entities before performing your I/O operations. You may want to use the validator package for this task.