Assignment Analysis3-1819

A Public Library System

Introduction

This assignment consists of the design and implementation of a Public Library System (PLS) with which a public library, lending out paper books to customers, can be managed. It goes without saying that the functionality required is only minimal and comes nowhere near a real PLS. Moreover, normally such a system would involve a database, but here we limit ourselves to storing data in object containers such as lists and dictionaries in RAM and files. No graphical user interface to the system is required. The system may be used by directly calling methods and accessing attributes of the classes defined in the system. The whole issue of security is also left out of scope of this assignment.

The library has paper books, a catalog of books, a loan administration, customers and a librarian (bibliothecaris). New customers can be added to the administration. New books (that means: the identification details of paper book items) can be added to the catalog. Customers can search for books and loan books. The library can be filled with books and customers from data files. All the data in the PLS can be backupped on file.

The assignment

The design consists of a UML Class Diagram. This should at least cover classes such as Book, Author, BookItem (a paper copy of a book), Customer, Librarian (if needed!), Person, Catalog, LoanAdministration, LoanItem. Especially the relationships between the classes should be shown in the diagram. Only the most essential attributes and methods per class have to be expressed. You may add a text document with additional explanations for the design.

The implementation should consist of a Python program which is clearly an implementation of the design. Additional classes, attributes and methods (not made explicit in the design) are allowed. Only with explicit confirmation by the teacher is it allowed to implement in a different language, such as C#. Php is explicitly not allowed.

The system must feature the following functions:

- Searching for a book. This shows if there are book items in the library present for this book and if they are available. The search criteria must cover various keys, such as title, author, ISBN, etc, and combinations of keys.
- Making a book loan for an available book item.
- Adding new customers.
- Adding new books.
- Making a backup of the data in the system (catalog of books and book items, data in the loan administration, list of customers, etc.)

- Recovering the library from a backup.

What you should hand in

The delivery to be handed in must consist of one zip-file. The zip-file must contain:

- A Readme.txt file with names and numbers of the team (preferably 2, but at most 3 students) and, if needed, explanations how to run the system.
- A single pdf-file with the Class Diagram and any explanations that go with it.
- A directory with the code files, containing a main file (called PLS.py) that can access the other files in the same directory.
- A user manual detailing the functions of the system, both for customers and for the librarian, and how to call them, with examples.
- Do not include any bulky data files or Python system files in the delivery. Only a small configuration data file is allowed.

The code must only use standard library modules. Any exceptions to this must be explained in the Readme.txt file. The code must run error-free. The code may only write to temporary storage directories on the local machine (such as C:\temp).

For assessment of the delivery, the system will be tested for the following functions:
- filling the system with books from a file of books in JSON format (file with books). You may add a fake ISBN number to book items and generate the number of book items per book.
- filling the system with customers from a file of people (file).
- adding a book item.
- adding a customer.
- searching a book.
- making a book loan.
- making a backup of the system in JSON format.
- restoring the system from a backup.

You can submit to the Google classroom of your class or to the email-address supplied by your teacher.

Deadline

Deadline for submission is Friday 26th of april, 23:59, that is the end of week 10 of period 3. There will be a second occasion to submit in the fourth period.