HomeWork C++ Language

Ali ZAINOUL
ali.zainoul.az@gmail.com

May 14, 2023

General informations

Particular attention will be paid to:

- your understanding of the question and what is being asked
- your code quality and **compilation of your program**.
- work must be returned **before 29 may at midnight (no exceptions)**.
- Exceptation: 1 directory for the problem, zipped into one zip, named like this: **firstName_lastName**_Cpp.zip, every directory/file must contain the source files (.cpp) and/or the header files (.hpp).

- **Compilation of a single file:** Assume we are into Project directory, then open a command prompt and type in:
  g++ -std=c++11 -o myExecutable main.cpp && ./myExecutable
- **Compilation of multiple files:** Assume we are into Project directory, and there is several files to be executed before the main; then open a command prompt and type in:
  g++ -std=c++11 -o myExecutable src/*.cpp main.cpp && ./myExecutable

  Where: ***.cpp** : represents all the src files .cpp contained in the src directory.

We recall that:

- pwd: prints the actual directory where you are
- cd myDirectory: change directory into myDirectory
- mkdir myDirectory: create a directory named myDirectory
- touch myFile: creating a file named myFile

- ls: listing all the files and directories present into your actual directory (you can know where you are by typing pwd)
- &&: stands for the binary operator applied into several commands. E.g: command1 && command2 is equivalent to: command 1 then command2.

Example: helloWorld.cpp source code

```
#include <iostream>

int main() {
    cout << "Hello, world!" << '\n';
    return 0;
}
```

Compile with: g++ -std=c++11 -o myExe helloWorld.cpp && ./myExe

C++ Language Problem

1 Problem Description

- You are tasked with creating a library management system in C++. The system should allow users to add, remove, and view books in a library. Each book should have a unique identifier, title, author, and genre.
- You should create a base class called `Book` and three derived classes: `FictionBook`, `NonFictionBook`, and `ReferenceBook`. Each derived class should have additional attributes that are specific to its genre. For example, a `FictionBook` might have a `bool` attribute that indicates whether it is a romance novel or a mystery, while a `ReferenceBook` might have a `std::vector<std::string>` attribute that lists the topics covered in the book.
- You should also create a `Library` class that manages a collection of `Book` objects. The `Library` class should have the following methods:
  - `void addBook(Book* book)`: adds a book to the library.
  - `void removeBook(std::string id)`: removes a book from the library with the given identifier.
  - `void viewAllBooks() const`: prints out information about all books in the library.

You should create a main function that creates a `Library` object, adds several `Book` objects to it, and then calls the `viewAllBooks()` method to display information about all the books in the library.

2 Requirements

- You should implement the classes and functions described above in C++.
- You should use appropriate data structures to manage the collection of `Book` objects in the `Library` class.
- You should include appropriate error handling to ensure that books are not added with duplicate identifiers and that books are not removed with invalid identifiers.
- You should include appropriate comments and documentation in your code.

3 Constraints

- You may not use any external libraries or frameworks.
- You should ensure that your implementation is memory safe and free of memory leaks.

4 Example Output

Here is an example output of the `viewAllBooks()` method:

ID: 12345 | Title: To Kill a Mockingbird | Author: Harper Lee | Genre: Fiction | Subgenre: Classic
ID: 67890 | Title: The Elements of Style | Author: William Strunk Jr. | Genre: Non-Fiction | Subgenre: Writing Guide
ID: 24680 | Title: The C++ Programming Language | Author: Bjarne Stroustrup | Genre: Reference | Subgenre: Programming Language
ID: 13579 | Title: Effective C++ | Author: Scott Meyers | Genre: Reference | Subgenre: Programming Language
ID: 86420 | Title: C++ Primer | Author: Stanley B. Lippman, Josée Lajoie, and Barbara E. Moo | Genre: Textbook | Subgenre: Programming Language

Good luck ♥