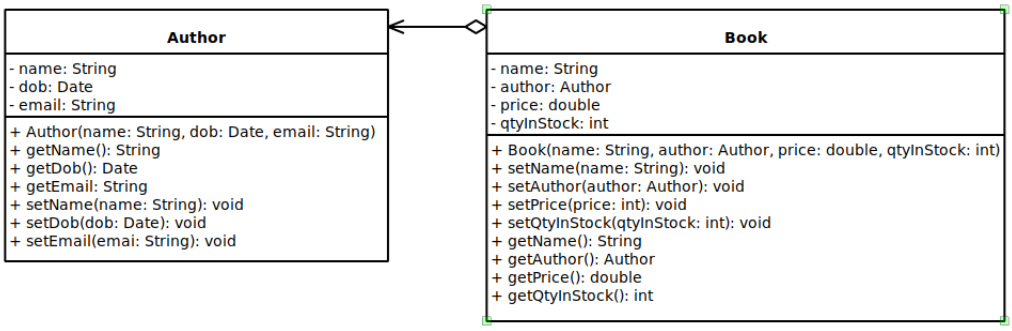
CSC 230: Elementary Data Structures and Algorithms

Fall 2022

Assignment 4

- General programming guidelines:
  - Create a separate NetBeans project with the name `QuestionXX`, where `XX` is the question number.
  - Do not forget necessary javadoc comments before classes and methods.
  - You should use *single line* or *multiline* comments, **if it is required**. Do not put unnecessary comments (Do not state the obvious!!!).
  - Use meaningful identifier.
  - Don't forget to check the parameters of your methods and throw appropriate exceptions as necessary.
  - **Creating correct NetBeans projects, zipping your final assignment folder, and testing it before uploading are your responsibilities. If any of these steps fails, you will receive a grade of zero.**
- Academic integrity policy
  - You are **not** allowed to use any online resources EXCEPT Revel, class lecture notes and Java documentation.
  - All programs/ code must be your own work.
  - You should be able to clearly explain every line of your code, if instructor requests you to do so.
  - Any violation of these policies will be considered as plagiarism and dealt accordingly.

Question 1 (50 points): Create a NetBeans project with the name `Question01`. Inside `question01` package, create a public `Author` and `Book` classes according to the following UML diagram. Make sure to choose appropriate access modifiers and throw exceptions as necessary. Add necessary javadoc and inline comments.



You can use the following sample code inside the `main` method to test your classes. Sample output is also provided.

```
// Date constructor will give you a deprecated warning but do not worry about it now.
Author sipser = new Author("Michael Siper", new Date("10/14/1964"), "sipser@gmail.com");
Book itc = new Book("Introduction to the Theory of Computation", sipser, 112.50, 23);
Book foc = new Book("Fundamental of Computer Science", sipser, 117.49, 14);

// Use of getters
System.out.println(itc.getName());
System.out.println(itc.getAuthor().getName());
System.out.println(itc.getAuthor().getDob());
System.out.println(itc.getAuthor().getEmail());
System.out.println(itc.getPrice());
System.out.println(itc.getQtyInStock());
```

```
System.out.println();

// Use of getters
System.out.println(foc.getName());
System.out.println(foc.getAuthor().getName());
System.out.println(foc.getAuthor().getDob());
System.out.println(foc.getAuthor().getEmail());
System.out.println(foc.getPrice());
System.out.println(foc.getQtyInStock());

System.out.println();

// Some sample setter usage and their output

itc.setQtyInStock(100);
foc.setQtyInStock(45);

System.out.println(itc.getQtyInStock());
System.out.println(foc.getQtyInStock());
```

Here is the expected output.

```
Introduction to the Theory of Computation
Michael Siper
Wed Oct 14 00:00:00 EDT 1964
sipser@gmail.com
112.5
23

Fundamental of Computer Science
Michael Siper
Wed Oct 14 00:00:00 EDT 1964
sipser@gmail.com
117.49
14

100
45
```

Note: We will find better ways to print the content using `toString` method but we need to talk about inheritance before that.

Question 2 – Asymptotic Analysis (50 points) This question needs a written answer. Do not forget to put your answer inside your assignment folder with above programming questions before you zip and upload.

```
private static boolean horizontal(char[][] array) {
        for (int i = 0; i < array.length; i++) {
            for (int j = 0; j < array[0].length - 3; j++) {
                boolean c1 = array[i][j + 3] == array[i][j + 2] + 1;
                boolean c2 = array[i][j + 2] == array[i][j + 1] + 1;
                boolean c3 = array[i][j + 1] == array[i][j] + 1;

                if (c1 && c2 && c3) {
                    return true;
                }
            }
        }

        return false;
}
```

Above method may be used to identify horizontal patterns in assignment-1: Question 2. What is the running time of this algorithm (Note that I am not specifying any case-best or worst)? *Justify your answer.*

Note:

- Assume that the array has n rows and m columns. Your answer will contain n and m.
- Your goal should be to setup an upper bound using O and a lower bound using Ω. If you can use Θ, that is what we should look for.