ECE 122: Introduction to Programming for ECE- Spring 2023

Project 2: My BookStore
(Basic Introduction to OOP)

Due Date: **Deadline: see website, class policy and Moodle for submission**

Description

The goal of the project is to design and implement a basic bookstore library. As administrator, you will be able to inquire the current inventory, and add or remove items. As user, you will be able to buy items by creating a shopping cart. Each book entry in the library can be identified by its title, its author, its reference, and its price. A sample media library comprised of 10 items is given below:

Reference	Title	Author	Price
GV5N32M9	A Brief History of Time	S. Hawking	$10.17
TR3FL0EW	The Alchemist	P. Coelho	$6.99
F2O9PIE9	Thus Spoke Zarathustra	F. Nietzsche	$7.81
R399CED1	Jonathan Livingston Seagull	R. Bach	$6.97
6Y9OPL87	The Time Machine	H. G. Wells	$5.95
5T3RRO90	Introduction to Programming in Python	R. Sedgewick	$69.99
3W2TB162	Atoms of Silence	H. Reeves	$28.02
TU2RL012	2001: A Space Odyssey	A. C. Clarke	$8.99
JI2PL986	20,000 Leagues under the Sea	J. Verne	$5.99
VC5CE249	Les Miserables	V. Hugo	$9.98

Submission/Grading Proposal

The project must include three files:

1. `Book.py` file that contains the class Book (to complete).
2. `Inventory.py` file that contains the class Inventory using the class Book (to complete).
3. `project2.py` file containing the main program using the class Inventory (to complete).

All three files must be submitted on Moodle. This project will be graded out of 100 points:

- You can do the project alone or by group of 2 max. If you do it in team, add your two names in the header of the file `project2.py`, and submit only **`*one*`** file per team (anyone from the two team members).
- Your program should implement all basic functionality/Tasks and run correctly (90 points).
- Overall programming style: program should be written in a clear way with all the proper comments. All functions should have a header doc-string as well (5 points).
- Pre-submission up to at least Option-3 (5 points). The program does not have to run correctly for pre-submission. Only the act of pre-submission is graded and not the content of your submission. If needed, pre-submission can be used to check if you are effectively making steady progress.

Preliminary-1 [10pts]

The class `Book` contains a main method, and it means it can be executed directly. Complete the class Book by providing an appropriate constructor (`__init__`) (where you will need to define multiple attributes) and a `__str__` method. The output of your code should look like:

```
Title: A Brief History of Time; Author: Stephen Hawking; (Ref: GV5N32M9; Price: $10.17)
```

Preliminary-2 [10pts]

At the first execution of the program `project2.py`, the output includes a menu containing multiple options:

```
Welcome to BestMedia
====================
-------------------------------------------------------
1-List Inventory; 2-Info Inventory; 3-Search Inventory;
4-Add Item; 5-Remove Item; 6-Inflation; 7-Shop; 8-Check-out

Enter Command:
Goodbye!
```

In the example above, no option was selected by the user (just pressing Enter), the program ended up stopping with a “Goodbye!” message.
How to proceed? Take inspiration from project1 and implement the basic “while loop” skeleton structure for this project2 file where you will be able to select multiple options from the user input command. Implement the simple option to leave the program (as shown above) for the time being.

All the other options will be reviewed below. The project is designed to be incremental, you can then debug, test and run your code after each new task/option is implemented. All the Tasks/options can be completed in any order (with some exceptions). Do not forget to comment your code (including function/method docstring). Make sure you obtain the **exact same output** for the **exact same input** for all the examples (this includes syntax, blank spaces, and skipping blank lines). Your program will also be tested with different inputs by the graders.

Option-1- [15pts]

Let us first see what is happening when option 1 is selected.

```
Welcome to BestMedia
====================
-------------------------------------------------------
1-List Inventory; 2-Info Inventory; 3-Search Inventory;
4-Add Item; 5-Remove Item; 6-Inflation; 7-Shop; 8-Check-out

Enter Command: 1
1 - Title: A Brief History of Time; Author: S. Hawking; (Ref: GV5N32M9; Price: $10.17)
2 - Title: The Alchemist; Author: P. Coelho; (Ref: TR3FL0EW; Price: $6.99)
3 - Title: Thus Spoke Zarathustra; Author: F. Nietzsche; (Ref: F2O9PIE9; Price: $7.81)
4 - Title: Jonathan Livingston Seagull; Author: R. Bach; (Ref: R399CED1; Price: $6.97)
5 - Title: The Time Machine; Author: H. G. Wells; (Ref: 6Y9OPL87; Price: $5.95)
6 - Title: Introduction to Programming in Python; Author: R. Sedgewick; (Ref: 5T3RRO90; Price: $69.99)
7 - Title: Atoms of Silence; Author: H. Reeves; (Ref: 3W2TB162; Price: $28.02)
8 - Title: 2001: A Space Odyssey; Author: A. C. Clarke; (Ref: TU2RL012; Price: $8.99)
9 - Title: 20,000 Leagues under the Sea; Author: J. Verne; (Ref: JI2PL986; Price: $5.99)
10 - Title: Les Miserables; Author: V. Hugo; (Ref: VC5CE249; Price: $9.98)

-------------------------------------------------------
1-List Inventory; 2-Info Inventory; 3-Search Inventory;
4-Add Item; 5-Remove Item; 6-Inflation; 7-Shop; 8-Check-out

Enter Command:
```

Here an inventory of the book store is listed (with their ID in front from 1 to 10). We can also see the references, title, author, and price of each item. At the end the menu selection is printed again and the program is waiting for you to select another option.

How to proceed?

1. In the `Inventory.py` file, you need to create a constructor with the following **requirements:** no input argument, and only a single attribute which is initialized as an empty list.
   You will then create a method `initialize` that is going to fill up your list of items with Book objects representing all the books present in your inventory. Ideally we would like to read all the inventory from a file (so we could easily consider 1000s of items if needed) but we will do that later in the semester. Here, you will need to fill up by hand (hard coded) all the attributes of the objects for our selected 10 items presented at the beginning of the project (a bit long but you can cc-paste title, etc. from this pdf file).

2. In the `project2.py` file, you will need to first instantiate a new empty Inventory (create an object of type "Inventory"). You will then need to call the method `initialize` before entering your while loop. If option 1 selected by the user, you will need to display the current Inventory using a simple `print` instruction. It means that you will need to add a method `__str__` to your Inventory class that returns a string. Hint: when constructing a string you could add instruction like `\n` to skip lines. Requirement: your `str` method for the Inventory must take advantage of `str` method defined in the Book class.

Option-2- [5pts]

Let us now see what is happening when option 2 is selected.

```
Enter Command: 2
#Items: 10
Total price $160.86
Most expensive item at $69.99
-------------------------------------------------------
1-List Inventory; 2-Info Inventory; 3-Search Inventory;
4-Add Item; 5-Remove Item; 6-Inflation; 7-Shop; 8-Check-out

Enter Command:
```

The program is displaying some info about the inventory, the number of books, the total value of all the book items, and the price of the most expensive book.

How to proceed?

1. In the `project2.py` file: implement the option 2 that contains a call to a method `info`
2. In the `Inventory.py` file: the method `info` (no argument and no return) that displays the information as presented in the output example. You will probably need a for loop that scans through all the books. To find the maximum price you could implement the linear search algorithm seen in class.

Option-3- [10pts]

Let us now see what is happening when option 3 is selected (here we do it three times):

```
Enter Command: 3
Enter a title keyword: 2001
8 - Title: 2001: A Space Odyssey; Author: A. C. Clarke; (Ref: TU2RL012; Price: $8.99)
-------------------------------------------------------
1-List Inventory; 2-Info Inventory; 3-Search Inventory;
4-Add Item; 5-Remove Item; 6-Inflation; 7-Shop; 8-Check-out

Enter Command: 3
Enter a title keyword: Time
1 - Title: A Brief History of Time; Author: S. Hawking; (Ref: GV5N32M9; Price: $10.17)
5 - Title: The Time Machine; Author: H. G. Wells; (Ref: 6Y9OPL87; Price: $5.95)
-------------------------------------------------------
1-List Inventory; 2-Info Inventory; 3-Search Inventory;
4-Add Item; 5-Remove Item; 6-Inflation; 7-Shop; 8-Check-out

Enter Command: 3
Enter a title keyword: Java
No book found!
-------------------------------------------------------
1-List Inventory; 2-Info Inventory; 3-Search Inventory;
4-Add Item; 5-Remove Item; 6-Inflation; 7-Shop; 8-Check-out
```

```
Enter Command:
```

The program is asking the user to enter a particular keyword to search in the title of the book. Once found, it will display all particular books containing the keyword (with their ID) (or returned "No book found!" if the book is not in the inventory).

How to proceed?

1. In the `project2.py` file: implement option 3 that asks the user to enter a keyword, and it should include a call to a method `search` with the keyword as argument. The method is not returning anything so all the printing must be done within the method.

2. In the `Inventory.py` file: the method `search` that scan the inventory and print the books when a title contains the keyword. As a reminder you can easily search in Python within a string using the operator `in`.

Option-4- [10pts]

Let us now see what is happening when option 4 is selected. Here we select option 4, then option 1, and option 2.

```

Enter Command: 4
Enter Title: Hamlet
Enter Author Name: W. Shakespeare
Enter Price: 4.59
Enter Reference: J45K99EE
-------------------------------------------------------
1-List Inventory; 2-Info Inventory; 3-Search Inventory;
4-Add Item; 5-Remove Item; 6-Inflation; 7-Shop; 8-Check-out

Enter Command: 1
1-Title: A Brief History of Time; Author: S. Hawking; (Ref: GV5N32M9; Price: $10.17)
2-Title: The Alchemist; Author: P. Coelho; (Ref: TR3FL0EW; Price: $6.99)
3-Title: Thus Spoke Zarathustra; Author: F. Nietzsche; (Ref: F2O9PIE9; Price: $7.81)
4-Title: Jonathan Livingston Seagull; Author: R. Bach; (Ref: R399CED1; Price: $6.97)
5-Title: The Time Machine; Author: H. G. Wells; (Ref: 6Y9OPL87; Price: $5.95)
6-Title: Introduction to Programming in Python; Author: R. Sedgewick; (Ref: 5T3RRO90; Price: $69.99)
7-Title: Atoms of Silence; Author: H. Reeves; (Ref: 3W2TB162; Price: $28.02)
8-Title: 2001: A Space Odyssey; Author: A. C. Clarke; (Ref: TU2RL012; Price: $8.99)
9-Title: 20,000 Leagues under the Sea; Author: J. Verne; (Ref: JI2PL986; Price: $5.99)
10-Title: Les Miserables; Author: V. Hugo; (Ref: VC5CE249; Price: $9.98)
11-Title: Hamlet; Author: W. Shakespeare; (Ref: J45K99EE; Price: $4.59)

-------------------------------------------------------
1-List Inventory; 2-Info Inventory; 3-Search Inventory;
4-Add Item; 5-Remove Item; 6-Inflation; 7-Shop; 8-Check-out

Enter Command: 2
#Items: 11
Total price $165.45000000000002
Most expensive item at $69.99
-------------------------------------------------------
```

```
1-List Inventory; 2-Info Inventory; 3-Search Inventory;
4-Add Item; 5-Remove Item; 6-Inflation; 7-Shop; 8-Check-out

Enter Command:
```

How to proceed? You will consider adding a method `add` to the class Inventory and a call to this `add` method in your main program. This method should have no argument and no return, so all of user entries must be processed within the method.

Option-5- [5pts]

Let us now see what is happening when option 5 is selected. Here we select option 5, then option 1, and option 2.

```
Enter Command: 5
which item do you want to delete: 5
-------------------------------------------------------
1-List Inventory; 2-Info Inventory; 3-Search Inventory;
4-Add Item; 5-Remove Item; 6-Inflation; 7-Shop; 8-Check-out

Enter Command: 1
1-Title: A Brief History of Time; Author: S. Hawking; (Ref: GV5N32M9; Price: $10.17)
2-Title: The Alchemist; Author: P. Coelho; (Ref: TR3FL0EW; Price: $6.99)
3-Title: Thus Spoke Zarathustra; Author: F. Nietzsche; (Ref: F209PIE9; Price: $7.81)
4-Title: Jonathan Livingston Seagull; Author: R. Bach; (Ref: R399CED1; Price: $6.97)
5-Title: Introduction to Programming in Python; Author: R. Sedgewick; (Ref: 5T3RR090; Price: $69.99)
6-Title: Atoms of Silence; Author: H. Reeves; (Ref: 3W2TB162; Price: $28.02)
7-Title: 2001: A Space Odyssey; Author: A. C. Clarke; (Ref: TU2RL012; Price: $8.99)
8-Title: 20,000 Leagues under the Sea; Author: J. Verne; (Ref: JI2PL986; Price: $5.99)
9-Title: Les Miserables; Author: V. Hugo; (Ref: VC5CE249; Price: $9.98)

-------------------------------------------------------
1-List Inventory; 2-Info Inventory; 3-Search Inventory;
4-Add Item; 5-Remove Item; 6-Inflation; 7-Shop; 8-Check-out

Enter Command: 2
#Items: 9
Total price $154.91
Most expensive item at $69.99
-------------------------------------------------------
1-List Inventory; 2-Info Inventory; 3-Search Inventory;
4-Add Item; 5-Remove Item; 6-Inflation; 7-Shop; 8-Check-out

Enter Command:
```

How to proceed? You must ask the user to enter a particular ID and the corresponding book will be removed from the inventory. Hint: You could simply use the `del` function.

Option-6- [10pts]

Inflation is a big issue in today's economy... here what would happen to our inventory with a 18% inflation.

```
Enter Command: 6
Enter Inflation %: 18
-------------------------------------------------------
1-List Inventory; 2-Info Inventory; 3-Search Inventory;
4-Add Item; 5-Remove Item; 6-Inflation; 7-Shop; 8-Check-out

Enter Command: 1
1-Title: A Brief History of Time; Author: S. Hawking; (Ref: GV5N32M9; Price: $12.000599999999999)
2-Title: The Alchemist; Author: P. Coelho; (Ref: TR3FL0EW; Price: $8.2482)
3-Title: Thus Spoke Zarathustra; Author: F. Nietzsche; (Ref: F209PIE9; Price: $9.2158)
4-Title: Jonathan Livingston Seagull; Author: R. Bach; (Ref: R399CED1; Price: $8.224599999999999)
5-Title: The Time Machine; Author: H. G. Wells; (Ref: 6Y9OPL87; Price: $7.021)
6-Title: Introduction to Programming in Python; Author: R. Sedgewick; (Ref: 5T3RRO90; Price: $82.58819999999999)
7-Title: Atoms of Silence; Author: H. Reeves; (Ref: 3W2TB162; Price: $33.0636)
8-Title: 2001: A Space Odyssey; Author: A. C. Clarke; (Ref: TU2RL012; Price: $10.6082)
9-Title: 20,000 Leagues under the Sea; Author: J. Verne; (Ref: JI2PL986; Price: $7.0682)
10-Title: Les Miserables; Author: V. Hugo; (Ref: VC5CE249; Price: $11.7764)

-------------------------------------------------------
1-List Inventory; 2-Info Inventory; 3-Search Inventory;
4-Add Item; 5-Remove Item; 6-Inflation; 7-Shop; 8-Check-out

Enter Command: 2
#Items: 10
Total price $189.8148
Most expensive item at $82.58819999999999
-------------------------------------------------------
1-List Inventory; 2-Info Inventory; 3-Search Inventory;
4-Add Item; 5-Remove Item; 6-Inflation; 7-Shop; 8-Check-out

Enter Command:
```

As you can see all the prices went up!

How to proceed? Create a method `adjust_price` for the class Inventory that accepts as single argument (the rate of inflation), modify all the prices in-place (so no return statement). Call this method from your main code as soon as you hit Option 6.

Option-7- [10pts]

It is time to shop! at this point we would like to take the role of a potential buyer. Let us choose Option 7 three times below.

```
Enter Command: 7
which item do you want to buy? 5
"The Time Machine" added to shopping cart!
-------------------------------------------------------
1-List Inventory; 2-Info Inventory; 3-Search Inventory;
4-Add Item; 5-Remove Item; 6-Inflation; 7-Shop; 8-Check-out

Enter Command: 7
which item do you want to buy? 1
"A Brief History of Time" added to shopping cart!
-------------------------------------------------------
1-List Inventory; 2-Info Inventory; 3-Search Inventory;
4-Add Item; 5-Remove Item; 6-Inflation; 7-Shop; 8-Check-out

Enter Command: 7
which item do you want to buy? 5
"The Time Machine" added to shopping cart!
-------------------------------------------------------
1-List Inventory; 2-Info Inventory; 3-Search Inventory;
4-Add Item; 5-Remove Item; 6-Inflation; 7-Shop; 8-Check-out

Enter Command:
```

As you can see, you are allowed to buy the same items multiple times.

How to proceed?

1. In the `project2.py` file: implement option 7 that asks the user to enter the book ID that you want to buy, confirm the purchase by displaying the title of the book (between quotes) you are adding to your shopping cart (as shown above). We are going to manage the shopping cart as another Inventory. It means that (at the beginning of the main program) you need to instantiate another object (for example named `cart`) as a new empty Inventory. At this point, `cart` becomes a new instance of the class Inventory and you can call any of its related methods or attributes. Once the user select the book you want to buy, you would need to add this book into your cart. You will need to modify the method `add` to include an optional argument that accepts an object of type Book.

2. In the `Inventory.py` file: the method `add` must now include an optional (Book) argument (you can use `None` as default). If the argument is provided (like in the case of the shopping cart), you can just append this book to your list of books; if it is not provided you can just ask the user to enter the book date (like you did in Option 4). To summarize: the method `add` will be used without argument in Option 4 (when associated with your original inventory), but will be called using one argument when associated with your cart.

Option-8- [15pts]

Time to check out! This is a follow-up from Option 7 (the example below is also the continuation of the example given in Option 7)

```
Enter Command: 8
Current shopping cart:
1-Title: The Time Machine; Author: H. G. Wells; (Ref: 6Y9OPL87; Price: $5.95)
2-Title: A Brief History of Time; Author: S. Hawking; (Ref: GV5N32M9; Price: $10.17)
3-Title: The Time Machine; Author: H. G. Wells; (Ref: 6Y9OPL87; Price: $5.95)

#Items: 3
Total price $22.07
Most expensive item at $10.17
Enter your promotion code if any: Voyage
Updated shopping cart:
1-Title: The Time Machine; Author: H. G. Wells; (Ref: 6Y9OPL87; Price: $5.3698749999999995)
2-Title: A Brief History of Time; Author: S. Hawking; (Ref: GV5N32M9; Price: $9.6615)
3-Title: The Time Machine; Author: H. G. Wells; (Ref: 6Y9OPL87; Price: $5.3698749999999995)

#Items: 3
Total price $20.40125
Most expensive item at $9.6615
Enter you credit card number: 5555-5555-5555
Purchase done!..Enjoy you new books
-------------------------------------------------------
1-List Inventory; 2-Info Inventory; 3-Search Inventory;
4-Add Item; 5-Remove Item; 6-Inflation; 7-Shop; 8-Check-out

Enter Command:
```

How to proceed? Some tips and requirement:

1. When listing your shopping cart: just use `print(cart)`.
2. You will also need to call the method `info`
3. When asking for a promotion code, you will consider 2 codes: "Voyage" that gives you -5% and "Parfait" that gives you -10%. If no code is provided, your shopping cart is not updated (and the updated list and info should not be printed).
4. When you apply a valid promotion, you need to use the `adjust_price` method written in Option 6 to adjust the price of your cart inventory. In that case the rate that you include as argument becomes negative (opposite of inflation).
5. There is no check on the credit card number you enter (could be anything, any format, any numbers/letter, does not matter).
6. Before leaving Option 8 make sure to re-instantiate your shopping cart (so it is empty again)
7. Before entering Option 8, make sure your shopping cart is not empty, if not your program must print `"Your cart is Empty!"` and leave the Option.