**Assignment 1**

**Book Inventory**

Before attempting this project, be sure you have completed all the reading assignments, non-graded exercises, discussions, and assignments to date.

***There is only one java file for all the coding of this assignment allowed.***

**Design and implement Java program as follows:**

(1) There will be a Book Java class with following attributes: id, title, and price.

1. The id is five digits (integer).
2. Check id for length: error message.
3. Check id for data type: error message.
4. Prompt is redisplayed if there is an error in id entry.
5. Title must allow for multiple words.
6. The data type for price is either float or double.
7. Array is populated only if all above elements are correctly entered.
8. Confirmation message is displayed.

(2) There will be an Inventory Java class which stores and manages a list of Book objects. An ArrayList must be created to hold the various orders.

(3) Inventory class will provide the following functionality:

a. **Add:** Prompts user for book data [five-digit id, multi word title, and price] and adds to the inventory list in array if all entries are correct. If the id does not meet the requirements, an error message will be displayed to the console, and the program will redisplay the prompt for an id. The new book is added to the ArrayList if all requirements are met.

b. **Remove.** Prompts user for book id, finds the id in the inventory ArrayList and removes it. A message is displayed with specific information about the book being removed. If the book matching the id is not in the inventory, remove request will fail and an error message will be displayed to the console. The program will redisplay the prompt for a correct id.

c. **Find:** Prompts user for book id, finds the id in the inventory list and displays all the data for the book (id, title, and price) horizontally. If the book matching the id is not in the inventory, find request will fail and an error message will be displayed to the console. The program will redisplay the prompt for a correct id.

d. **Display**: Display all the book information in the inventory list from the ArrayList for each book to the console horizontally for each entry.

(4) There is a main () method at the end of the program.

* 1. An inventory ArrayList is declared.
  2. The method begins with instantiation of the Inventory class.
  3. A menu is displayed either in this method or called to another method.
  4. The menu will use numbers for the selections and consist of the four options [Add, Remove, Find, and Display] and an exit selection numbered 9. It is within a loop.
  5. The switch structure is used for calling to the appropriate functionality (methods) and includes an error selection if correct numbers not chosen.
  6. The main () should be considered as a driver, that is, only the minimum number of code lines are in it. Most of the coding is done in methods that are outside of the main () method.
  7. The switch should not be considered process coding – no conditionals – but calls to methods in the program.

**Style and Documentation:**

Make sure your Java program is using the recommended style such as:

* Javadoc comment up front with your name as author, date, and brief purpose of the program.
* Comments for variables and blocks of code to describe major functionality (Refer to the Documentation of Programs found in Week 1).
* Meaningful variable names and prompts – no single character that is meaningless.
* Class names are written in upper CamelCase
* Constants are written in All Capitals
* Use proper spacing and empty lines to make code more user-friendly, readable.

**Capture execution:**

You should capture and label screen captures associated with compiling your code and running a passing and failing test cases for each functionality.

**Sample run 1:**

MENU

1: Add book

2: Remove book

3: Find book

4: Display all books

9: Exit program

Enter your selection: 4

The inventory has no books.

MENU

1: Add book

2: Remove book

3: Find book

4: Display all books

9: Exit program

Enter your selection: 9

Thank you for using the program. Goodbye!

**Sample run 2:**

MENU

1: Add book

2: Remove book

3: Find book

4: Display all books

9: Exit program

Enter your selection: 1

What is the book id (5 integers)? 12345

What is the book title? The End of Civilization

What is the book price (double value)? 19.98

MENU

1: Add book

2: Remove book

3: Find book

4: Display all books

9: Exit Program

Enter your selection: 4

Book id 12345 Title The End of Civilization Price 19.98

MENU

1: Add book

2: Remove book

3: Find book

4: Display all books

9: Exit program

Enter your selection: 9

Thank you for using the program. Goodbye!

**Assignment Deliverables:**

Deliverables include a **single Java program** (.java) and a **single Word (or PDF) document**. ***No zipped files are acceptable***. The Java and Word/PDF files should be named appropriately for the assignment (as indicated in the *CMIS242 Submissions Expectations* document posted in Week 1): **CMIS242ASG1[name].java, CMIS242ASG1[name].docx, or CMIS242ASG1[name].pdf.**

Though you might construct the assignment with multiple classes, the final submission should be a single Java program. The process of combining multiple classes into a single Java program is found in Week 1 Content under the Practice Exercise thread and titled HandleCandlesONE Java program. Submit the single Java program as the attachment in the Assignment 1 folder.

The word (or PDF) document should include a Test Plan with multiple test cases. Each test case should be referenced to a screen capture showing the successful compiling and running of each of the test cases. One YouTube site for how to generate screen shot/capture is [(32) 💻 How to Screenshot on Laptop or PC with Windows - YouTube](https://www.youtube.com/watch?v=gSn5JtS53BI) [Go to 6:51 in video where Space X information is detailed.] Each screen capture should be properly labeled clearly indicated what the screen capture represents. It is suggested that the test table include these columns:

* Test Case #
* Menu Selection
* Under input book attributes the three items of id number, title, and price
* The output would include prefix, id number, title, and price.
* Processing Command(s)
* Messages for errors and other information to the user

Submit your files to Assignment 1 folder submission area no later than the due date listed in your online classroom.

**Grading Rubric:**

The following grading rubric will be used to determine your grade: Note: Elements Included in the Grading of the Program document contains additional grading elements.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Criteria | **Level 3** | **Level 2** | **Level 1** | **Score** |
|  | **20 points** | **13 points** | **5 points or less** | **/20** |
| Criterion 1 Add | * Prompts * Messages for confirmation and errors * id length, data type, duplicate checked * Title words * Price float or double * Code efficient | Incomplete implementation of the requirements | Missing or significantly incorrect implementation |  |
|  | **20 points** | **13 points** | **5 points or less** | **/20** |
| Criterion 2 Remove | * Use of book id * Messages for confirmation and errors * Redisplay prompt if id not in list * Code efficient | Incomplete implementation of the requirements | Missing or significantly incorrect implementation |  |
|  | **20 points** | **13 points** | **5 points or less** | **/20** |
| Criterion 3 Find | * Use of book id * Messages for confirmation and errors * Redisplay prompt of if id not in list * Code efficient | Incomplete implementation of the requirements | Missing or significantly incorrect implementation |  |
|  | **15 points** | **10 points** | **5 points or less** | **/15** |
| Criterion 4 Display | * Use Menu Selection * Display horizontal for each book * Clean and simple display | Incomplete implementation of the requirements | Missing or significantly incorrect implementation |  |
|  | **10 points** | **5 points** | **0 points** | **/10** |
| Criterion 5  Test Plan, Documentation, Format | * Test table * Screen shots referencing scenarios * Documentation followed: both line and block comments * Spacing and blank lines | Incomplete implementation of the requirements | Missing or significantly incorrect implementation |  |
|  | **15 points** | **10 points** | **5 points or less** | **/15** |
| Criterion 6  The main () method | * ArrayList * Instantiation of Inventory class * No reference to the name of the program * Menu – five selections * Switch * Driver (minimum code) | Incomplete implementation of the requirements | Missing or significantly incorrect implementation |  |
| **Total** |  |  |  | **/ 100** |