|  |  |
| --- | --- |
| **Assignment Case** |  |
| COMP6708016 Object Oriented Programming |
| **Computer Science** | **E223-COMP6708016-DJ04-01** |
| ***Valid on*** *Even Semester Year 2021/2022* | **Revision 00** |

## Soal

*Case*

**Book’s Manager**

Book’s Manager is a simple program for library in school. The school need it to help with their book listing management. You to create a simple program using **JAVA Programming Language** with **Object Oriented Programming** concepts such as **Encapsulation**.**Composition**, and **Aggregation**. Here are the details of the simulation:

A picture containing application

Description automatically generated

**Figure 1. Menu**

1. **View Books (Menu 1)**

* When user choose this menu, **validate** if the list is **empty, show error message**.

**A picture containing text

Description automatically generated**

**Figure 2. Error Message**

* Else, show all Books and their attributes in the list.

**Table

Description automatically generated**

**Figure 3. Book Collection**

1. **Insert Book (Menu 2)**

* When user choose this menu, the program will ask user to input title. **Validate** the title must be **between** **5 – 10 characters** (**inclusive**).

**A screenshot of a computer

Description automatically generated with low confidence**

**Figure 2. Input Book Title**

* After that, the user will choose the book type. **Validate** Book type must be either “**Novel**” or “**Comic**” (**case** **sensitive**).

Text

Description automatically generated

**Figure 3. Input Book Type**

* Lastly, the user will choose the **number** **of** **chapters** for the book. Validate **number** **of** **chapters** must be **between 1 to 10 (inclusive)**.

Text

Description automatically generated

**Figure 4. Input Number of Chapters**

* Then, the program will **calculate** **Price** using following conditions:

|  |  |
| --- | --- |
| Book Type | Price |
| Novel | **100000 +** **numberOfChapters \* 1000** |
| Comic | **50000 + numberOfChapters \* 1000** |

* Then, the program will **generate** an **ID** for the book. The ID must satisfy the following format:

**BKXXX**

X = Random number between 0 – 9

Example: BK878

* Then, **insert** the **Book** to a **list** (**ArrayList** / **Vector** / **Array**) and return to **main menu**.

1. **Delete Book (Menu 3)**

* When user choose this menu, **validate** if the list is **empty, show error message**.
* However if the list is **not empty**, user must choose **which Book to remove** from the list. **Validate** the chosen number must be **between 1** and **the total amount of Book in the list.**

**Table

Description automatically generated**

**Figure 5. Choose Book to Remove**

**Table

Description automatically generated**

**Figure 6. Book List after Deletion**

1. **Exit (Menu 4)**

* When user choose this menu, **exit the program**.

**If you need any assistance, kindly ask your assistants for help.**