|  |  |
| --- | --- |
| **Assignment Case** |  |
| COMP6708016 Object Oriented Programming |
| **Computer Science** | **E223-COMP6708016-JK01-01** |
| ***Valid on*** *Even Semester Year 2021/2022* | **Revision 00** |

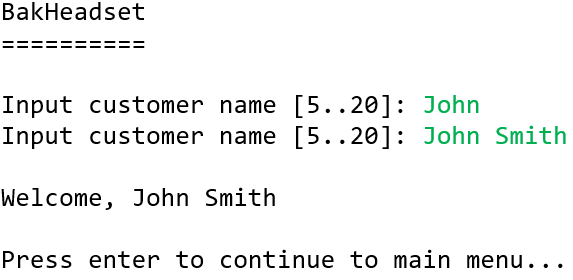
## Soal

*Case*

**BakHeadset**

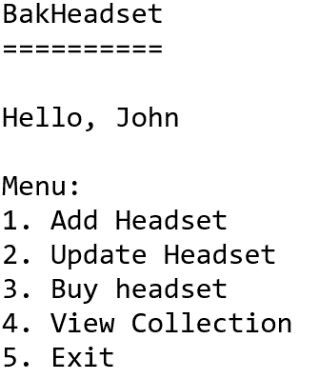
**BakHeadset** is a new company that sell headset. Since there’s increasing demands, the owner of **BakHeadset** asked you to create a program to help manage their product transactions. You must create the program using **JAVA Programming Language** with **Object Oriented Programming Concept** such as **Encapsulation, Composition, and Aggregation**. The requirements for the application are:

* First, the program will ask for **customer** **name**. Validate the **length** must be **between 5 and 20 characters long**.

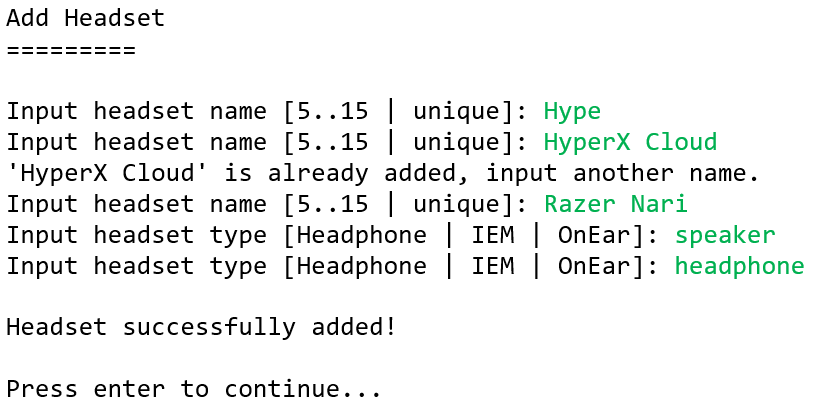


* Then, the user will be redirected to **main** **menu**. Show the **customer’s** **name** with the menus. The application consists of **5 menus**:

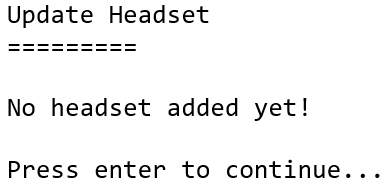
1. **Add Headset**
2. **Update Headset**
3. **Buy headset**
4. **View Collection**
5. **Exit**

****

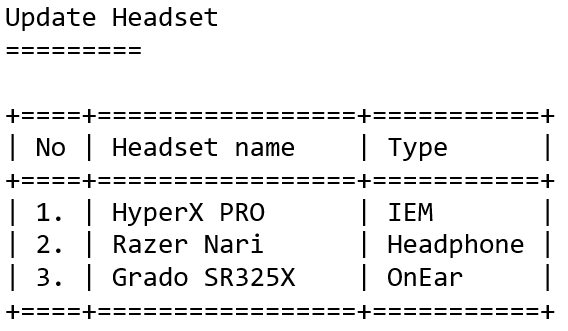
* If user choses menu **1** (“**Add Headset**”), then the program will:
  + Ask the user to input **headset** **name**. Validate the **length** must be **between 5 and 15 characters long**. Also, make sure the headset name is **unique**.
  + Ask the user to input **headset** **type**. Validate the **value** must be **either “Headphone”, “IEM”, or “OnEar” (case insensitive)**.
  + The program will **save** the headset information into a **list** (**ArrayList / Vector / Array**). Then, return to **main menu**.

****

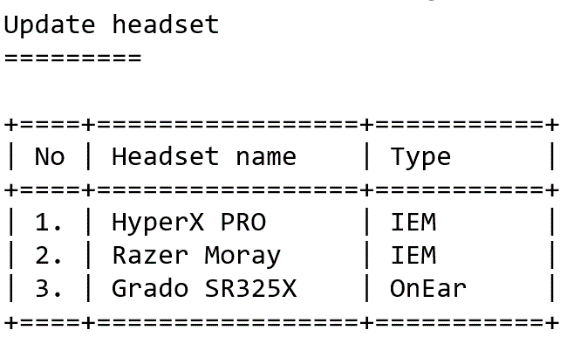
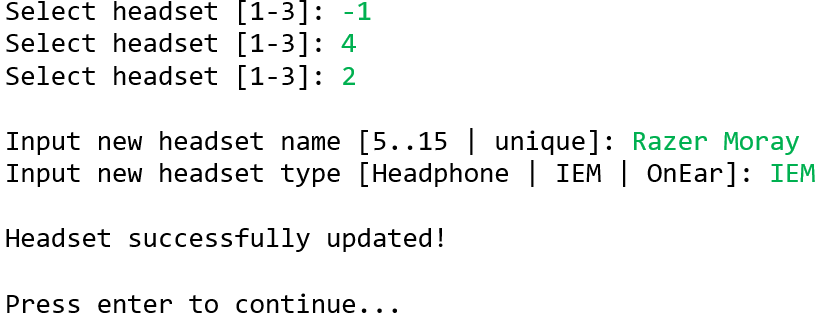
* If user choses menu **2** (“**Update Headset**”).
  + If there’s no headset added in the **list** yet. **Show** this **message** and return to **main menu**.

****

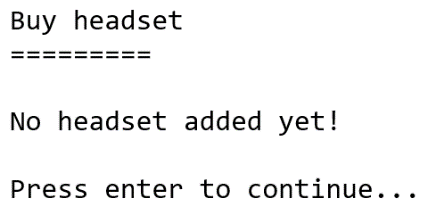
* + **Otherwise**, **show** all headsets’ details.



* + Then, the program will:
    - Ask which **headset no. to be updated**. Validate the **value** must be **between** **1 and** **total headset data**.
    - Ask the user to input **headset** **name**. Validate the **length** must be **between 5 and 15 characters long**. Also, make sure the headset name is **unique**.
    - Ask the user to input **headset** **type**. Validate the **value** must be **either “Headphone”, “IEM”, or “OnEar” (case insensitive)**.
  + The program will **update** the **selected** **headset** with the **new** **information** into. Then, return to **main menu**.



* If user choses menu **3** (“**Buy headset**”).
  + If there’s no headset added in the **list** yet. **Show** this **message** and return to **main menu**.

****

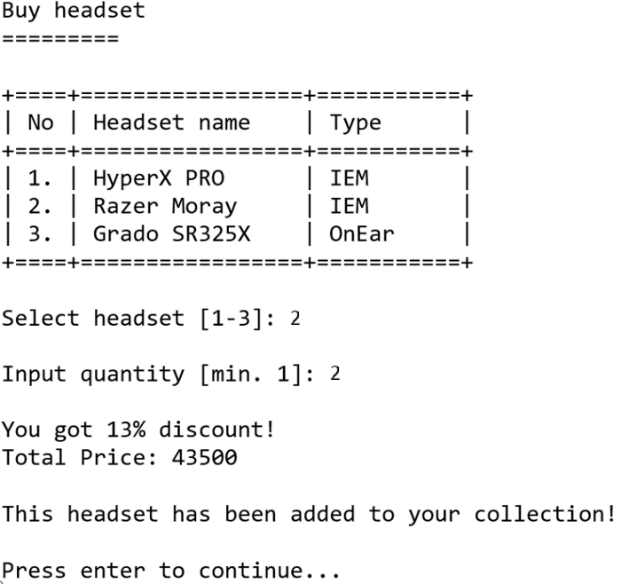
* + **Otherwise**, **show** all headsets’ details. Then, the program will:
    - Ask which **headset no. to be bought**. Validate the **value** must be **between** **1 and** **total headset data**.
    - Ask the user to input **quantity**. Validate the **value** must be **at least 1**.
    - **Determine** the **headset price** based on this table:

|  |  |
| --- | --- |
| Type | Headset Price |
| Headphone | 20000 |
| IEM | 25000 |
| OnEar | 30000 |

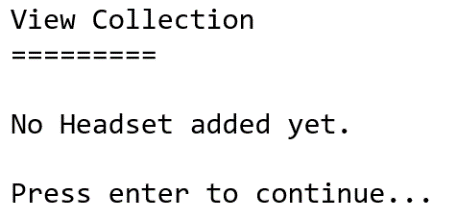
* + - Every customer that buys a headset will also get **discount**. **Randomize the discount** for each customer, **random the value between 10 to 25.**
    - **Calculate** the **total price** based on this formula:

**Total price = (Quantity \* Headset Price) \* ((100 – Discount Amount) / 100)**

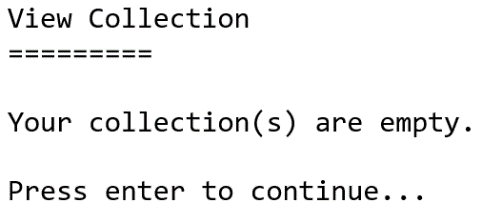
* + Then, the program will **add** the **selected** **headset** into **user’s** **collection** and return to **main** **menu**.



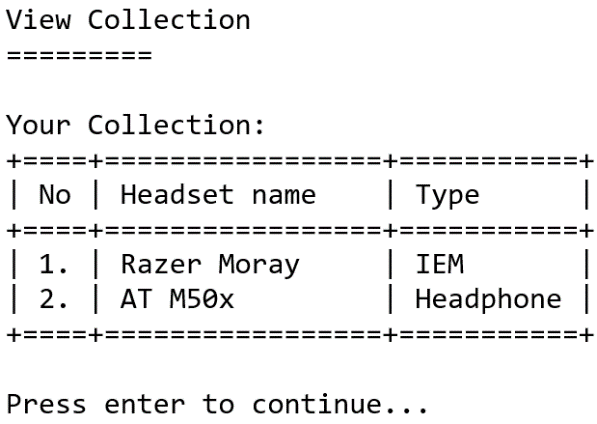
* If user choses menu **4** (“**View Collection**”).
  + If there’s no headset added in the **headset** **list** yet. **Show** this **message** and return to **main menu**.

****

* + **Otherwise**. If the user haven’t **made** **any** **purchases** **yet** (indicating the **collection** **list** is **empty**). **Show** this **message** and return to **main menu**.

****

* + **Otherwise**, **show** all headsets in **user’s** **collection** **list**. Then, return to **main menu**.



* If user choses menu **5** (“**Exit**”), then program will exit.

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