Practicum Final Exam – Odd Semester Year 2021/2022		
Subject COMP6178003 – Introduction to Programming		
Class : BA18/BA80/BB18/BQ11/BR11	**Start Date**	**: 13 January 2022**
Lecture : D1828 – Hendra, S.Kom., M.T. D 5151 – Ikhtiar Faahakhododo, S.Kom., M.TI D 6434 – Lie Maximilianus Maria Kolbe, S.Kom., M.T.I.	**Start Time**	**: 07:20**
	End Date	**: 13 January 2022**
	End Time	**: 09:20**

PERATURAN UJIAN:

Exam Regulations:

- Mahasiswa tidak diperbolehkan berdiskusi dan/atau bekerja sama dengan peserta ujian lainnya

  Student is not allowed to discuss and/or work together with other exam participants

- Mahasiswa tidak diperbolehkan untuk membuka dan menyalin dari **BUKU** atau **CATATAN**, **VIDEO** dari pengajar (recording kelas, VBL, Youtube, dsb) dan **REFERENSI** lainnya

  Student isn't allowed to open and copy from any resources such as notes, videos (class recording, VBL, Youtube, etc) and other references

- Mahasiswa tidak diperbolehkan membuka dan menyalin jawaban dari internet (google, stackoverflow, dsb)

  Student isn't allowed to open and copy answer from the internet (google, stackoverflow, etc)

- Asisten **BERHAK** memberi nilai 0 **(NOL)** bagi peserta ujian yang melakukan segala bentuk kecurangan

  Assistant is able to give 0 (ZERO) score for exam participant who does any cheating actions

- Kumpulkan jawaban tepat pada waktunya, apabila terlambat mengumpulkan maka jawaban tidak akan dikoreksi dan nilai mahasiswa adalah 0

  Submit the answer on time, if not, then the answer will not be checked, and the students will receive 0 (ZERO)

- Bila Anda tidak membaca peraturan ini, maka Anda dianggap telah membaca dan menyetujuinya

  If you have missed to read these regulations, so you are considered to have read and agreed on it

BINUS UNIVERSITY
JANGAN NYONTEK
APAPUN CARA/ALASANNYA, NYONTEK ☛ DROP OUT
SK REKTOR No. 1426/SK/PTTKK-UBIN/VII/2016

SOFTWARE YANG DIGUNAKAN:

Software will be used:

- Eclipse 2020.6
- Java 8

FILE YANG DIKUMPULKAN:

File must be collected:

- JAVA
- CLASS

PERHATIAN!

Attention!

- Bagi yang mengerjakan tidak sesuai dengan soal, maka akan diberikan nilai **NOL (0)**

  *For those who do not work in accordance with the exam case will be marked as **ZERO (0)***

- Bagi yang mengerjakan tidak sesuai dengan software dan versi yang telah ditetapkan, maka akan tetap dikoreksi dengan software dan versi yang telah ditetapkan

  For those who do not work in accordance with the software and specific version will be corrected by the predefined software and version

- Kompres semua jawaban yang akan diunggah. Pastikan format pengumpulan nama file dan ekstensi sesuai dengan format berikut: **[NIM]-[NAMA].zip**

  *Compress all file that will be uploaded. Make sure the format for collecting file name and extension according to the following format: **[NIM]-[NAME].zip***

Important Notes :

1. Do not code only on the **main** function, **each menu must be separated into a separate function**
2. You are **allowed** to use any sorting algorithm aside from **.sort()** method from **Collections Class**. It's recommended to use **Bubble Sort Algorithm**

Soal

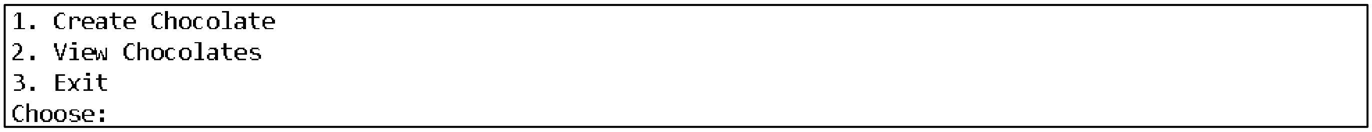
Case

Chocolate Factory

Chocolate Factory is a small factory that just got established in 2021. Ever since middle of 2021, the factory has some trouble managing goods and menus. The owner of the factory asked you as their most capable programmer to create an application to help the factory. The application has the following requirements :

The application consists of **3 menus** :

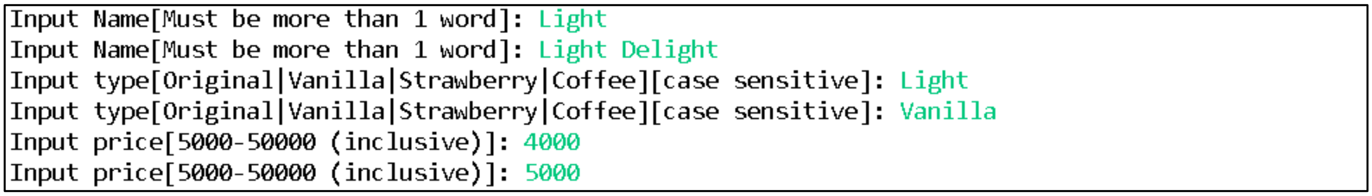
1. **Create Chocolate**
2. **View Chocolate**
3. **Exit**



```
1. Create Chocolate
2. View Chocolates
3. Exit
Choose:
```

Figure 1. main menu

- If user choose **menu 1** ("**Create Chocolate**"), the application will :
  - Prompt user to input **Chocolate Name**. Validate the input must be **more than 1 words**
  - Prompt user to input **Chocolate Type**. Validate the input must be **between** "**Original**", "**Vanilla**", "**Strawberry**", and "**Coffee**" (**case sensitive**)
  - Prompt user to input **Chocolate Price**. Validate the input must be **between 5000** and **50000** (**inclusive**)



```
Input Name[Must be more than 1 word]: Light
Input Name[Must be more than 1 word]: Light Delight
Input type[Original|Vanilla|Strawberry|Coffee][case sensitive]: Light
Input type[Original|Vanilla|Strawberry|Coffee][case sensitive]: Vanilla
Input price[5000-50000 (inclusive)]: 4000
Input price[5000-50000 (inclusive)]: 5000
```

Figure 2. insert chocolate input

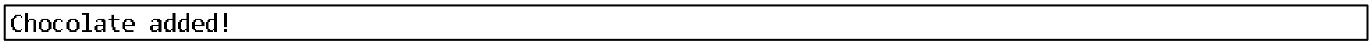
- **Generate Chocolate ID** based on the following formula

Chocolate ID = XXYYY

X = random between A – Z

Y = random between 0-9

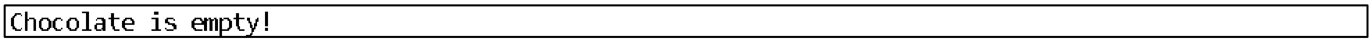
- **Insert** chocolate data into the **list**, and **display** following message



```
Chocolate added!
```

Figure 3. success insert message

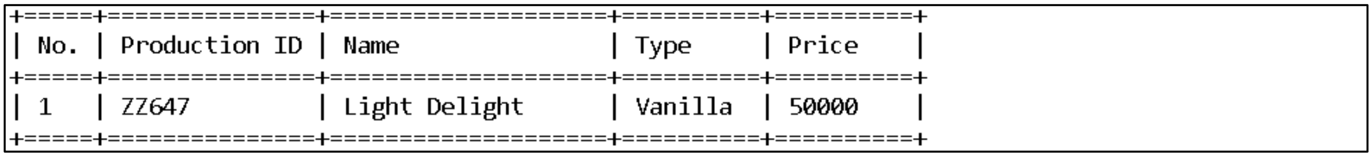
- If user choose **menu 2** ("**View Chocolates**"), the application will :
  - **Display** submenu consisting of **3 menus** :
    - **Update Chocolate**
    - **Delete Chocolate**
    - **Go back to main menu**
  - Prompt user to input **Action Choice**. Validate the input must be **between 1** and **3** (**inclusive**)
  - If user choose **submenu 1** ("**Update Chocolate**"), the application will :
    - **Check** for chocolate in the **list**. If there are **no chocolate**, **display** following message and **redirect** to **submenu**



```
Chocolate is empty!
```

Figure 4. empty chocolate message

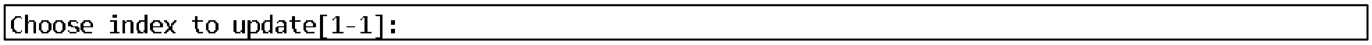
- Otherwise,
  - **Display** all chocolate from list **sorted by Chocolate Name** in **ascending order**



```
+=====+===============+===================+=========+==========+
| No. | Production ID | Name              | Type    | Price    |
+=====+===============+===================+=========+==========+
| 1   | ZZ647         | Light Delight     | Vanilla | 50000    |
+=====+===============+===================+=========+==========+
```

Figure 5. view chocolates

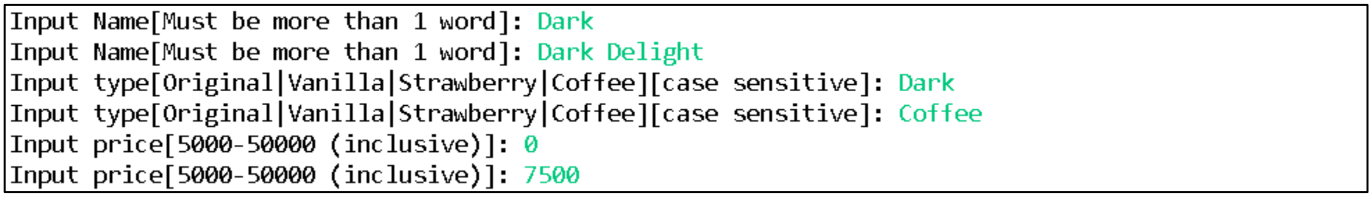
- ❖ Prompt user to input **Update Index**. Validate the input must be **between 1** and **current size of** list (**inclusive**)



```
Choose index to update[1-1]:
```

Figure 6. index update input

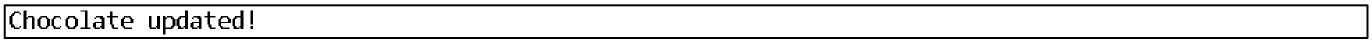
- ❖ Prompt user to input **Chocolate Name**. Validate the input must be **more than 1 word**
- ❖ Prompt user to input **Chocolate Type**. Validate the input must be **between** "**Original**", "**Vanilla**", "**Strawberry**", and "**Coffee**" (**case sensitive**)
- ❖ Prompt user to input **Chocolate Price**. Validate the input must be **between 5000** and **50000** (**inclusive**)



```
Input Name[Must be more than 1 word]: Dark
Input Name[Must be more than 1 word]: Dark Delight
Input type[Original|Vanilla|Strawberry|Coffee][case sensitive]: Dark
Input type[Original|Vanilla|Strawberry|Coffee][case sensitive]: Coffee
Input price[5000-50000 (inclusive)]: 0
Input price[5000-50000 (inclusive)]: 7500
```

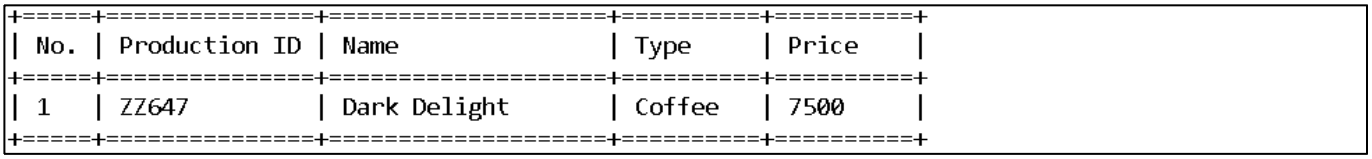
Figure 7. update chocolate input

- ❖ **Update** selected chocolate, **display** following message and **redirect** to **submenu**



```
Chocolate updated!
```

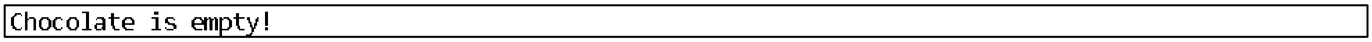
Figure 8. success update message



```
+=====+===============+===================+=========+==========+
| No. | Production ID | Name              | Type    | Price    |
+=====+===============+===================+=========+==========+
| 1   | ZZ647         | Dark Delight      | Coffee  | 7500     |
+=====+===============+===================+=========+==========+
```

Figure 9. chocolate list after update

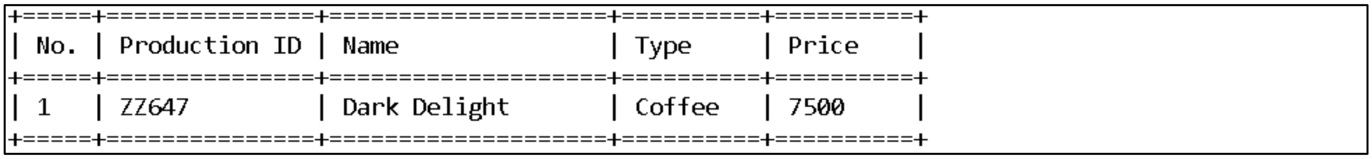
- ▪ If user choose **submenu 2** ("**Delete Chocolate**"), the application will :
  - ➢ **Check** for chocolate in the **list**. If there are **no chocolate**, **display** following message and **redirect** to **submenu**



```
Chocolate is empty!
```

Figure 10. empty chocolate message

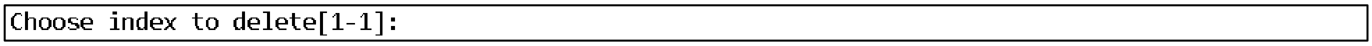
- Otherwise,
  - **Display** all chocolate from list **sorted by Chocolate Name** in **ascending order**



```
+=====+===============+====================+==========+==========+
| No. | Production ID | Name               | Type     | Price    |
+=====+===============+====================+==========+==========+
| 1   | ZZ647         | Dark Delight       | Coffee   | 7500     |
+=====+===============+====================+==========+==========+
```

Figure 11. view chocolates

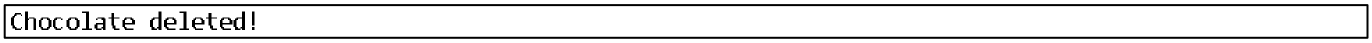
  - Prompt user to input **Delete Index**. Validate the input must be **between 1** and **current size of** list (**inclusive**)



```
Choose index to delete[1-1]:
```

Figure 12. index delete input

  - **Remove** selected chocolate from the **list**, **display** following message, and **redirect** to **submenu**



```
Chocolate deleted!
```

Figure 13. success delete message

- If user choose **submenu 3** ("**Go back to main menu**"), the application will :
  - **Redirect** to **main menu**

- If user choose **menu 3** ("**Exit**"), the application will **exit**