


Assignment Case	
DS using CH6	
Periode Berlaku Semester Genap 2023/2024 Valid on Even Year 2023/2024	Software Laboratory Center Assistant Recruitment 24-2

Note: Please focus on the main logic and main feature!

(Splash screen and design are not scored)

Soal

Case

HarmonyHUB

HarmonyHUB is a sophisticated music collection library designed to organize and manage music seamlessly. Utilizing **Hashtable Datastructure**, it efficiently retrieves music data. This logic ensures rapid access to your music library, allowing you to quickly search, browse and enjoy your favorite tracks, **With HarmonyHUB** you can easily create music collection, and maintain a harmonious digital music collection tailored to your preferences.

➤ Hash Table

- The **hash table** have a **max size** of **10**
- For the **hash function**, add all the **ASCII** of the **music title** and then **multiply it by the last digit of your trainee number** and finally **MOD it by max size**. Example:

```

Music title: seasons
Trainee number: T999
Hash: (115 + 101 + 97 + 115 + 111 + 110 + 115) * 9 MOD 10 = 6
s: 115
e: 101
a: 97
s: 115
o: 111
n: 110
s: 115

```

Figure 1. Hash function example

- When more than one data has the same hash, **use chaining for the collision handling**, For the chaining method **use push mid** according to the **music rating**. Example:

```
Music title: seasons
Rating: 4.0
Hash: 6

Music title: something
Rating: 3.0
Hash: 6

Music title: go
Rating 5.0
Hash: 6

HASH TABLE
=====
1.
2.
3.
4.
5.
6. something -> seasons -> go

the chaining is sorted by rating, because it use push mid by rating
```

Figure 2. Chaining example

➤ **Home Page (Menu 1)**

- This menu contains of 5 options, which are **Add Music**, **View Music Collection**, **Search Music** and **Remove Music** and **Exit**.
- **Prompt** user to **input chosen menu**. **Validate** the input must **between 1 and 5 inclusively**.

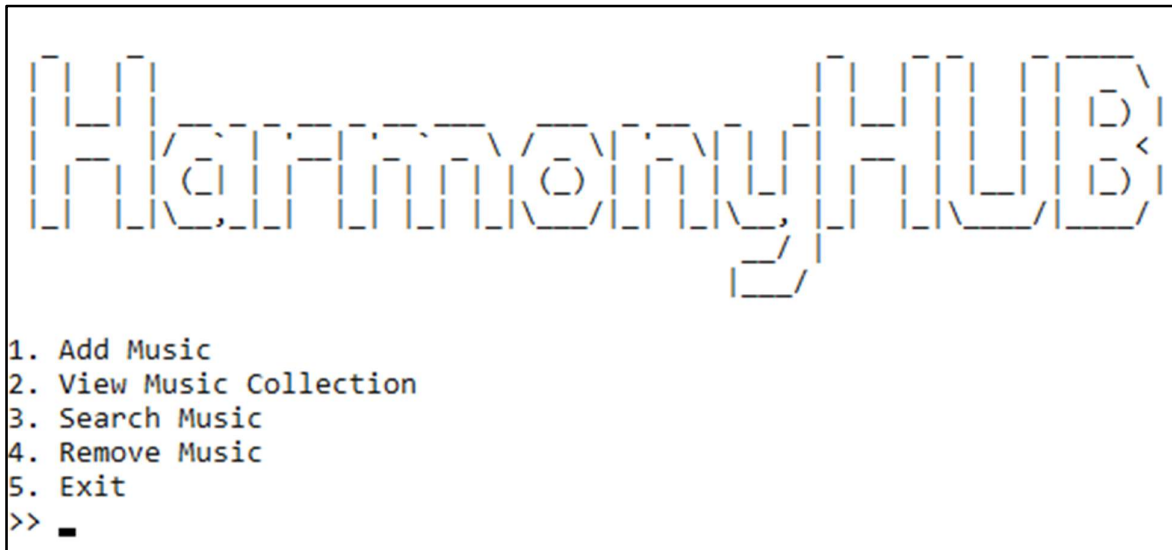


Figure 3. Home Page

➤ If the user chooses **Add Music** (Option 1), then:

- **Prompt** user to input **music title**. Validate music title must be more than **5 characters**

Insert music title [more than 5 characters]:

Figure 4. Input music title

- **Prompt** user to input **artist name**. **Validate** artist name **must be between 6 and 20 characters inclusively**

Insert artist name [minimum 6 character maximum 20 character]:

Figure 5. Input artist name

- **Prompt** user to input **music genre**. **Validate** music genre **must be Pop, Soul, or Rock (case sensitive)**

Insert genre of the music [Pop | Soul | Rock]:

Figure 6. Input music genre

- **Prompt** user to input **music rating**. **Validate** music rating **must be between 1.0 and 5.0 inclusively**

Insert rating of the music [minimum 1 maximum 5.0]:

Figure 7. input music rating

- **If all criteria are met, store music data in the hash table, and display a success message**

```
New music successfully added!  
Press enter to continue...
```

Figure 8. Success message

➤ If the user chooses **View Music Collection** (Option 2), then:

- **Display** all music collection.

```
View Music Collection  
=====
```

Title	Artist Name	Genre	Rating
something	hendra	Pop	3.0/5.0
seasons	hendra	Soul	4.0/5.0
seasons	daniel	Rock	5.0/5.0

```
=====
```

Press enter to continue...

Figure 9. View music collection

- If there is **no music currently display error message**

```
There is no music at this moment...  
Press enter to continue
```

Figure 10. View music error message

➤ If the user chooses **Search Music** (Option 3), then:

- **Prompt** user to input **music title** they want to search, **Validate** if user input **0** it will **redirect user back to main menu**

```
Insert music name you want to search [0 to go back]:
```

Figure 11. Search music

- **If** there is no music in the hash table, **display error message**

```
There is no music at this moment...  
Press enter to continue
```

Figure 12. No music error message

- **If** the music is not found, **diaply error message**

```

Search Music
=====
Insert music name you want to search [0 to go back]: love

Music with title "love" not found!
Press enter to continue...

```

Figure 13. Music not found error message

- If music is **found**, **Display** search result

```

Music found!
=====
| Title                | Artist Name          | Genre    | Rating  |
=====
| something            | hendra               | Pop      | 3.0/5.0 |
=====

Press enter to continue...

```

Figure 14. Search result

- If there are more than 1 music with same title, **Display** all of the search results

```

Insert music name you want to search [0 to go back]: seasons

Music found!
=====
| Title                | Artist Name          | Genre    | Rating  |
=====
| seasons              | hendra               | Soul     | 4.0/5.0 |
| seasons              | daniel               | Rock     | 5.0/5.0 |
=====

Press enter to continue...

```

Figure 15. Search result 2

- If the user chooses **Remove Music** (Option 4), then:

- **Display** all music collection, If there is **no music** currently display error message

```

There is no music at this moment...
Press enter to continue

```

Figure 16. No music error message

- **Prompt** user to input **music title** they want to delete

```
Remove Music
=====
| Title           | Artist Name       | Genre  | Rating |
=====
| something       | hendra            | Pop    | 3.0/5.0 |
| seasons        | hendra            | Soul   | 4.0/5.0 |
| seasons        | daniel            | Rock   | 5.0/5.0 |
=====

Insert music title you want to delete [0 to go back]:
```

Figure 17. Remove music

- If music is not found, display error message

```
Insert music title you want to delete [0 to go back]: Love

Music with title "Love" not found!
Press enter to continue...
```

Figure 18. Music not found error message

- If music is **found** and there is **only 1 music found**, delete it immediately.
- If music is **found** and there are **more than 1 music found** (More than 1 music with same title), **display** all music **found**, and **prompt** user to input **music index or number** they want to delete. **Validate** music number **must be between 1 and total music found**

```
2 Music found!
=====
| No | Title           | Artist Name       | Genre  | Rating |
=====
| 1  | seasons        | hendra            | Soul   | 4.0/5.0 |
| 2  | seasons        | daniel            | Rock   | 5.0/5.0 |
=====

Select music number you want to delete [1 - 2]: _
```

Figure 14. Input music number

➤ If the user chooses **Exit** (Option 5), then:

- **Stop** the program

Please run the EXE file to see the sample program.