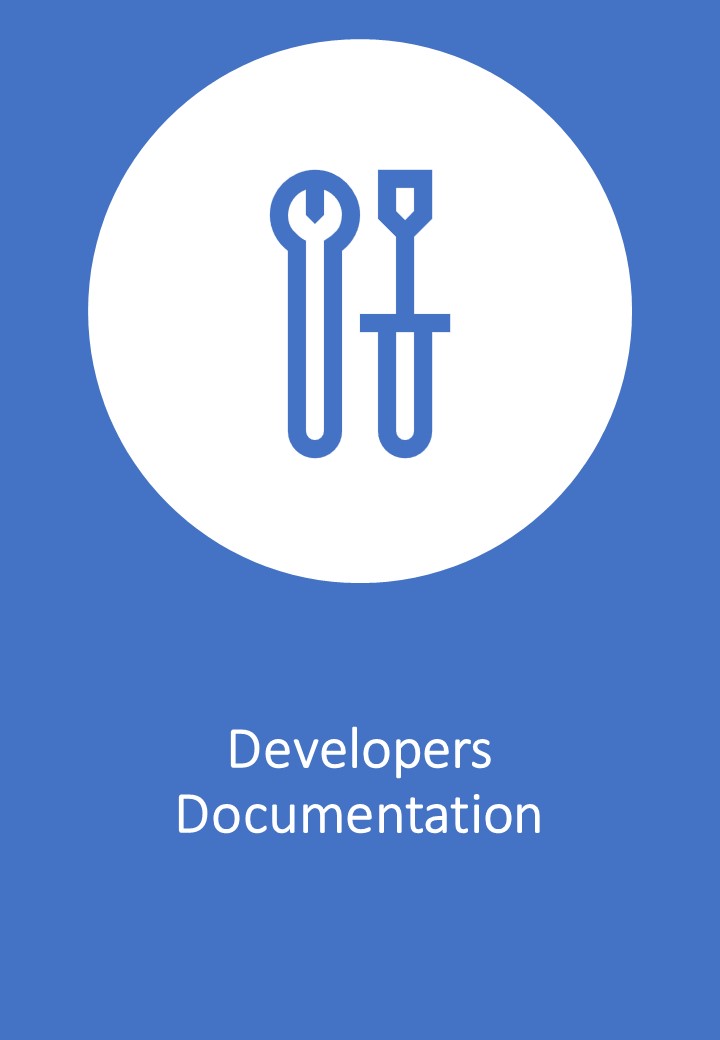
****

**SONGS**

**Overview:**

The program is designed to store data of songs in a database. For each entry the program can store the following data: title of the song, name of the performer (singer or group), title of the album it was released on, year of release, genre of the music (rock, pop, hip-hop, jazz, classical, etc.), length of song (in minutes and seconds).

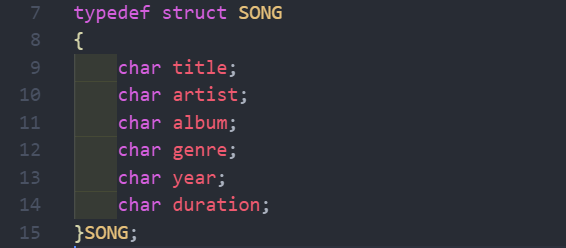
**Scope:**

The program is be capable of following functions:

* load existing database into memory.
* add new song entry to the database.
* save the database into a file.
* display all songs of a user selected artist.
* display all details of the songs of a user selected album.
* list all songs that were released in a user selected year.
* list all songs of a user selected genre.
* output the whole database.

**Explanation of the Solution:**

The program relies heavily on File Handling, Dynamic Memory Allocations, Data Structures and String Manipulations. For the ease of readability and accessibility, the code has been divided into several modules, and each extensive task has its own function.

 For the input of the information, we rely upon this Data Structure:

It allows use to receive input as a string, or as an integer for the release year and song duration inputs.

**Explanation of the Modules & Functions:**

The code has been divided into 2 main modules:

* main.c file for calling the functions
* header.h file for function implementations and definitions

Here is a list of all the functions being used in *“header.h”* file:

* CLEAR\_SCREEN
* FILE\_CHECK\_FUNCTION
* MAIN\_MENU\_FUNCTION
* MAIN\_A\_FUNCTION
* MAIN\_A\_1\_FUNCTION
* MAIN\_A\_1\_SEARCH\_FUNCTION
* MAIN\_A\_2\_FUNCTION
* MAIN\_A\_3\_FUNCTION
* MAIN\_B\_FUNCTION

**CLEAR\_SCREEN:**

The function uses a simple *system(“cls”)* library function to clear the output on consoled to improve readability and remove unnecessary outputs from previous functions.

**FILE\_CHECK\_FUNCTION:**

This function uses file handling to create a pointer that points to the Database file with “r” (read) parameter. If it fails to open the file, the pointer returns NULL, which tells us that the database file doesn’t exist. This function is used in the very start to check whether we have a database or does the user need to create one.

**MAIN\_MENU\_FUNCTION:**

This function prints out the main menu for the user and saves the choice into an integer and returns that value back to main function where we can decide what to do with that choice. Its parameter is *“int choice”.*

**MAIN\_A\_FUNCTION:**

If the user chooses the option 1 (Load the Existing Database), this function prints out another menu for the user and saves the choice into an integer and returns that value back to main function where we can decide what to do with that choice. Its parameter is *“int choice”.* The menu asks whether the user would like to search the database, add a new entry to it, or output the whole database. In case the user wants to exit the program at any point, the function allows to do with another choice.

**MAIN\_A\_FUNCTION:**

If the user chooses the option 1 (Load the Existing Database), this function prints out another menu for the user and saves the choice into an integer and returns that value back to main function where we can decide what to do with that choice. Its parameter is *“int choice”.* The menu asks whether the user would like to search the database, add a new entry to it, or output the whole database. In case the user wants to exit the program at any point, the function allows to do with another choice.

**MAIN\_A\_1\_FUNCTION:**

If the user chooses the option 1 (Search the Database), this function prints out another menu for the user and saves the choice into an integer and returns that value back to main function where we can decide what to do with that choice. Its parameter is *“int choice”.* The menu asks whether the user would like to search the song with the title, artist, genre, year, or album name. In case the user wants to exit the program at any point, the function allows to do with another choice.

**MAIN\_A\_1\_SEARCH\_FUNCTION:**

This function makes use of 2 parameters: *“int choice”*  which helps us decide which kind of parameters of the song database do we need to search for, and *“int QUERY\_CEHCK”*  which helps us to decide whether search was successful or not.

Note: *We used another library function i.e. fflush(stdin) here because the previous \n from scanf was still in the memory and it was causing some issues regarding the next scanf that we had to use.*

The function uses file handling to open the database file in read mode and uses dynamic memory allocation to read the contents of the file in a dynamic array. After taking the search parameter (song name, or artist name etc.), we read the database file line by line and check whether the query matches the data that we have. After carrying out the search, we determine whether the search was successful or not by changing the value or *“int QUERY\_CHECK”* and returning it to main function.

**MAIN\_A\_2\_FUNCTION:**

This function is for the second choice of menu i.e. Adding a new entry to the database. The database file is opened with the parameter “a+” which helps us to append the incoming data at the end of the existing data in the database file. The function also checks whether the file exists or not. The user is asked to input the song information and its stored in the file. To avoid the hassle of searching and matching upper-case and lower-case letters, the functions makes use of *“tolower(c)”* to keep all the data in lower-case letters. The function exits after closing the file pointer.

**MAIN\_A\_3\_FUNCTION:**

This function is for the third choice of the main menu i.e. Output the database. The database file is opened with the parameter “r” (Read only). The function also checks whether the file exists or not. Using a dynamic memory allocated array to read the file was totally unnecessary here so we just read the file until the EOF (End Of File), and output each character to the console. The function exits after closing the file pointer.

**MAIN\_B\_FUNCTION:**

This function is for the second choice of the main menu i.e. Create a new database. The database file is opened with the parameter “r” (Read only). First, we check whether the database exists or not. If it does exist, we don’t create another file because having two database files would be a conflicting situation. If the file doesn’t exist, we use “a+” parameters of file handling and take the song information from the user and store it into the file and close the file pointer. Now that we have initiated a database, the user can go ahead and add new entries.