

```
// Name: Rodrigo Ignacio Rojas Garcia
// Course Number: ECE 2230
// Section: 001
// Semester: Spring 2017
// Assignment Number: 2

// Libraries Declaration Section
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "guitar.h"
#include "guitardb.h"
#include "list.h"

// Prints the commands of the program
void menu(void)
{
    // Asks the user to enter a command and prints those commands
    printf("*****\n");
    printf("Please enter any of these commands:\n");
    printf("ADD - Add a guitar to the Database\n");
    printf("DELETE - Delete a Guitar from the Database\n");
    printf("LOOKUP - Find Guitar in Database and Print It\n");
    printf("LIST - List all Guitar Current in Database\n");
    printf("HELP - Print this help message\n");
    printf("QUIT - Exit Program\n");
    printf("*****\n\n");
}

int main(int argc, char *argv[])
{
    // Variable Declaration Section
    char line[MAXCHARACTERS];
    char command[MAXCHARACTERS];
    char error = -5;
    int addition_result = -2;
    guitardb_t list;

    // Prints the commands of the program
    menu();

    // Allocates dynamic memory for a guitar list which will first use a function i
n guitardb.c which will use a function in list.c to allocate memory for the list
    list = guitardb_init();

    while (1)
    {
        error = -5;
        printf("> ");
        // Allows user to enter command
        memset(line, '\0', sizeof(line));
        memset(command, '\0', sizeof(command));
        fgets(line, MAXCHARACTERS, stdin);
        sscanf(line, "%s %s", command, &error);

        // If user enters command ADD, it will prompt the user to enter an id numbe
r followed by allocated memory for an item and then calling guitardb_add
        // which will add the item to the list only if it is not already stores on
the list
        if (strcmp("ADD", command) == 0 && error == -5)
        {
            // Variable declaration Section
            guitar_t item;
            int id_number;
            printf("Enter the following data for the guitar:\n");
            printf("Enter ID (ID must be >= 0): ");
            fgets(line, MAXCHARACTERS, stdin);
```

```
                sscanf(line, "%d", &id_number);
                item = guitar_init();
                guitar_setid(item, id_number);
                // Function will add the guitar characteristics to the data base and li
st
                addition_result = guitardb_add(list, item);

                if (addition_result != 0)
                {
                    printf("Guitar with ID %d is already in database\n", guitar_getid(i
tem));
                    guitar_free(item);
                }
            }

            // If user enters command DELETE, it will prompt the user to enter an id nu
mber, then the guitardb_delete function is called which will
            // return the address of the item based on the id number passed if it exist
s on the list, if so, it will free the memory allocated for the
            // item, if no item found, lets the user know that it is not on the databas
e
            else if (strcmp("DELETE", command) == 0 && error == -5)
            {
                int id_number;
                guitar_t item;
                printf("Enter the ID of the Guitar to Delete (ID must be >= 0): ");
                fgets(line, MAXCHARACTERS, stdin);
                sscanf(line, "%d", &id_number);
                item = guitardb_delete(list, id_number);
                if (item != NULL)
                {
                    guitar_free(item);
                    printf("Guitar Removed from Database\n");
                }
                else
                {
                    printf("Guitar with ID %d was not in Database\n", id_number);
                }
            }

            // If user enters command LOOKUP, it will prompt the user to enter the id n
umber of the item they want to look for, function guitardb_lookup
            // is called which will serach for an item on the list that has the same id
number, if found, returns the address of the item and prints
            // the characteristics of the item, if not found, it lets the user know tha
t there is no item with that id number on the database
            else if (strcmp("LOOKUP", command) == 0 && error == -5)
            {
                guitar_t item;
                int id_number;
                printf("Enter the ID of the Guitar to Lookup (ID must be >= 0): ");
                fgets(line, MAXCHARACTERS, stdin);
                sscanf(line, "%d", &id_number);
                item = guitardb_lookup(list, id_number);
                if (item != NULL)
                {
                    guitar_print(item);
                }
                else
                {
                    printf("Guitar with ID %d was not in Databse\n", id_number);
                }
            }

            // If user enters command LIST, it will call the guitardb_report function w
```

```
high will print all items on the database
    else if (strcmp("LIST", command) == 0 && error == -5)
    {
        printf("Contents of the Database are:\n");
        guitardb_report(list);
    }

    // If user enters command HELP, it will print the commands of the program
    else if (strcmp("HELP", command) == 0 && error == -5)
    {
        menu();
        printf("\n");
    }

    // If user enters command QUIT, function guitardb_finalize will be called f
reeing all memory used in the program and then
    // exiting as well
    else if (strcmp("QUIT", command) == 0 && error == -5)
    {
        guitardb_finalize(list);
        exit(0);
    }

    // If user enters an unkown command, this message will display and user wil
l be asked to enter a new command
    else
    {
        printf("Option \"%s\" is not recognized...\n", command);
    }
}
return 0;
}
```