Karl Boeh

ICS 265

4/19/2022

Term Project – SQLITE3 C Library

I decided to make a program which shows some applications of the C SQLITE3 library. I wrote the program using a modular programming style. Besides some of the text prompting the user to hit ENTER and describing what the program is doing, the program is broken up into separate functions that all serve a different purpose. All of my function prototypes and declarations are properly commented, and I will run through the .c source file giving a quick explanation of the functions before running/presenting my program to the class. I should be able to accomplish all of this within the five minutes allotted for the presentation.

First, the program has an introduction which will give the user a basic overview of what the program will do. Next, I wrote 2 separate functions which both work the same and each one creates a database file. If there is an error creating the .db file or if there is already an identical .db file that exists, the program will give an error but continue running. Next, the program has 2 functions which create tables to store records that will be inserted later in the program. After the two tables are created, one named CYBERSECURITY and one named COMPUTERSCIENCE, records will be entered into these tables using SQL statements. CYBERSECURITY and COMPUTERSCIENCE will have records filled for ID PRIMARY KEY, which is required for database entries, COURSENAME, DEPARTMENT, COURSENUMBER, and CREDITS for all the major core required classes for these two degrees. I purposely entered incorrect records in the CYBERSECURITY table which I will update later with SQL statements. The next two functions will display all the records in the terminal that were just entered in the last step. I used some publicly available code from <https://www.tutorialspoint.com/sqlite/sqlite_c_cpp.htm> to help with the callback function and the sqlite3\_exec() function. After the records from the two tables are printed to the screen, there is a function which uses SQL statements to update the incorrect records from the CYBERSECURITY table. After these entries are updated the last two functions search for all of the ICS classes in each table. Only 2 classes from the cybersecurity field are ICS classes. All 8 classes in the computer science major are ICS classes. Only the classes where DEPARTMENT = ICS will print to the screen. This is the end of the program.

To get this program to compile and run you will have to take certain steps that are different than running the C programs we have been writing in this class. I was writing this program in Code::Blocks IDE and I needed to create a static library from the .c source file downloaded from sqlite.org and turn it into a static library. After I made the static library, I had to link the .a file to my main.c file to get the SQLITE3 library to work. I expect you will be running this on a Linux machine so I will give instructions for how to get this program to compile and run on Linux.

First, the SQLITE3 library can be downloaded by running “sudo apt-get install sqlite3”

Second, this command may need to be run “sudo apt-get install sqlite3 libsqlite3-dev” I had to run this to get the SQLITE3 library to work on my linux machine

Last, when compiling the .c file, this command will need to be run “gcc karl\_boeh\_main.c -l sqlite3”. Without adding the “-l sqlite3”, the .c program will not compile.

After this, the output file can be run. When prompted, press ENTER to move through the program.

When researching the SQLITE3 C library, there were some websites that I found extremely helpful. The first was <https://www.tutorialspoint.com/sqlite/sqlite_c_cpp.htm> and the publicly available code that I used in my program from this site is:

static int callback(void \*data, int argc, char \*\*argv, char \*\*azColName){

int i;

fprintf(stderr, "%s: ", (const char\*)data);

for(i = 0; i<argc; i++){

printf("%s = %s\n", azColName[i], argv[i] ? argv[i] : "NULL");

}

printf("\n");

return 0;

}

This function is used as an argument for the sqlite3\_exec() function and makes it possible to execute the SQL statements.

The other website that I found helpful with this project was the sqlite.org website. I also used Kahn Academy for some basic help learning how to write the SQL statements.