

In C SQLite Datenbank einbinden

Beitrag von n4seweis » 17.03.2021 10:37:01

Vom einfachen Programm zum fertigen Debian-Paket, Fragen rund um Programmiersprachen, Scripting und Lizenzierung.

[n4seweis](#)

Beiträge: [14](#)

Registriert: 06.03.2021 16:01:10

[In C SQLite Datenbank einbinden](#)

Hey,

versuche grade ne SQL Datenbank in C einzubinden und kriege dabei diese Fehlermeldung:

Fehlermeldungen:

Code: [Alles auswählen](#)

```
"undefined reference to ||=== Build: Debug in database (compiler: GNU GCC Compiler) ===|
/usr/bin/ld: obj/Debug/sqlite3.o||in function `pthreadMutexAlloc':|
/home/n4seweis/programming/learnC/k16/sqlite3.c|26691|undefined reference to `pthread_mutexat
/usr/bin/ld: /home/n4seweis/programming/learnC/k16/sqlite3.c|26692|undefined reference to `pt
/usr/bin/ld: /home/n4seweis/programming/learnC/k16/sqlite3.c|26694|undefined reference to `pt
/usr/bin/ld: obj/Debug/sqlite3.o||in function `pthreadMutexTry':|
/home/n4seweis/programming/learnC/k16/sqlite3.c|26836|undefined reference to `pthread_mutex_t
/usr/bin/ld: obj/Debug/sqlite3.o||in function `sqlite3ThreadCreate':|
/home/n4seweis/programming/learnC/k16/sqlite3.c|30744|undefined reference to `pthread_create'
/usr/bin/ld: obj/Debug/sqlite3.o||in function `sqlite3ThreadJoin':|
/home/n4seweis/programming/learnC/k16/sqlite3.c|30764|undefined reference to `pthread_join'|
/usr/bin/ld: obj/Debug/sqlite3.o||in function `unixDlOpen':|
/home/n4seweis/programming/learnC/k16/sqlite3.c|40398|undefined reference to `dlopen'|
/usr/bin/ld: obj/Debug/sqlite3.o||in function `unixDLError':|
/home/n4seweis/programming/learnC/k16/sqlite3.c|40412|undefined reference to `dlerror'|
/usr/bin/ld: obj/Debug/sqlite3.o||in function `unixDlSym':|
/home/n4seweis/programming/learnC/k16/sqlite3.c|40438|undefined reference to `dlsym'|
/usr/bin/ld: obj/Debug/sqlite3.o||in function `unixDlClose':|
/home/n4seweis/programming/learnC/k16/sqlite3.c|40443|undefined reference to `dlclose'|
||error: ld returned 1 exit status|
||=== Build failed: 11 error(s), 0 warning(s) (0 minute(s), 0 second(s)) ===|
```

C Programm:

Code: [Alles auswählen](#)

```
#include <stdio.h>
#include <stdlib.h>
#include "sqlite3.h"

int main()
{
    sqlite3 *db;
```

```

int rc;
rc=sqlite3_open("test.db",&db);

if(rc !=0){

printf("Datenbanken konnte nicht geöffnet werden %s \n",sqlite3_errmsg(db));
exit (EXIT_FAILURE);
} else{

    printf("Datenbanken erfolgreich geöffnet");

}

sqlite3_close(db);
return 0;
}

```

Habe die Header und C datei von der website eingebunden und eine Datenbank im entsprechenden Verzeichnis erstellt.

Zuletzt geändert von [Meillo](#) am 17.03.2021 14:21:51, insgesamt 1-mal geändert.

Grund: *Meillo: Titel angepasst s/SQL Lite/SQLite/*



[heisenberg](#)

Beiträge: [2314](#)

Registriert: 04.06.2015 01:17:27

[SQLite in C verwenden](#)

[Beitrag](#) von [heisenberg](#) » 17.03.2021 11:11:58

Hi,

Grundsätzlich ist der Ablauf beim nutzen von externen Bibliotheksfunktionen in C so:

- Relevantes Devel-Paket installieren(Im Fall von Debian als Betriebssystem heisst das sqlite3-dev). (Alternativ kann man den Quellcode von sqlite3 auch direkt herunterladen und verwenden. Das ist aber weniger komfortabel.)
- Header Datei einbinden(Deine Zeile `#include"sqlite3.h"` ist falsch). Mit Doppelpunktzeichen eingebundene Headerdatei müssen mit absolutem/relativem Pfad angegeben werden. Und es liegt bestimmt keine sqlite3.h in Deinem Verzeichnis wo auch Dein Programm ist, oder? (Ansonsten fehlt ein Leerzeichen nach include). Hier ist es hilfreich auch nochmal genauer nach zu lesen, wie man Dateien in C einbindet(Suchbegriffe für die Suchmaschine: "C" "Header" "Dateien" "einbinden").
- Beim kompilieren die relevante Lib mit angeben

Suche Dir doch nochmal ein Tutorial für "C" "Programmierung" "Sqlite3" in Deiner bevorzugten Suchmaschine und lerne, wie es verwendet wird. Ich habe dieses Tutorial verwendet: <https://zetcode.com/db/sqlitec/>

Anmerkung: Die Software heisst nicht SQL oder SQL Lite, sondern SQLite. Es hat mich teilweise etwas verwirrt und ich habe mich gefragt ob es nicht vielleicht doch um MySQL

geht, was etwas dezent anderes ist. (SQLite ist eine Bibliothek zum Zugriff auf eine dateibasierte Datenbank. MySQL ist ein Datenbankserver bzw. Datenbankmanagementsystem). "SQL" ist die Abfragesprache, die von beiden Systemen mit unterschiedlichem Funktionsumfang unterstützt wird.

Grüße,
h.



[Meillo](#)

Moderator

Beiträge: [7642](#)

Registriert: 21.06.2005 14:55:06

Wohnort: Balmora

Kontakt:

[Re: In C SQL Lite Datenbank einbinden](#)

[Beitrag](#) von [Meillo](#) » 17.03.2021 11:56:36

Ich habe deinen Code, n4seweis, bei mir lokal abgespeichert als `code.c`.

(Edit: Eigentlich sollte der Include von sqlite3.h in spitzen Klammern gesetzt sein weil du eine System-Datei includest. Aber wenn es in Double-Quotes steht und im Arbeitsverzeichnis nichts gefunden wird, werden auch System-Header versucht. Wenn du in dem Verzeichnis keine passenden Header-Dateien hast verhaelt sich die Variante mit "" gleich wie die mit <>.)

Dann habe ich folgendes Paket installiert: [libsqlite3-dev](#).

Zusaetzliche Dateien habe ich nicht runtergeladen.

Anschliessend habe ich mit diesem Befehl erfolgreich kompiliert:

(Das letzte Argument ist der Name deiner Code-Datei, abzueglich der Dateiendung. Das geht so nur wenn dur nur eine Datei hast.)

Shell-Session der Aktion:

Code: [Alles auswählen](#)

```
:-Q ls
code.c

:-Q make LDFLAGS=-lsqlite3 code
cc -lsqlite3 code.c -o code

:-Q ls
code* code.c

:-Q
```

Use ed once in a while!

[JTH](#)

Moderator

Beiträge: [1874](#)

Registriert: 13.08.2008 17:01:41

Wohnort: Berlin

[Re: SQLite in C verwenden](#)

[Beitrag](#) von [JTH](#) » 17.03.2021 12:10:55

[heisenberg](#) hat geschrieben: ↑

17.03.2021 11:11:58

Mit Doppelanführungszeichen eingebunde Headerdatei müssen mit absolutem/relativem Pfad angegeben werden.

Wenn ich da mal korrigierend zwischenspringen darf: Das stimmt nicht. In beiden Formen,

```
#include ""
```

und

```
#include <>
```

, kann rein der Name des Headers angegeben werden und wird er ja üblicherweise auch nur. Der Unterschied ist dann nur, in welchen und in welcher Reihenfolge in den vorgegebenen Include-Verzeichnissen nach dem Header gesucht wird, wie Meillo geschrieben hat.

[Manchmal](#) bekannt als **Just (another) Terminal Hacker**.



[heisenberg](#)

Beiträge: [2314](#)

Registriert: 04.06.2015 01:17:27

[Re: SQLite in C verwenden](#)

[Beitrag](#) von [heisenberg](#) » 17.03.2021 12:52:40

@n4seweis: Im Zweifelsfall hier lieber auf die anderen hören. Meine C-Kompetenzen sind eher gering.



[Meillo](#)

Moderator

Beiträge: [7642](#)

Registriert: 21.06.2005 14:55:06

Wohnort: Balmora

Kontaktdaten:

[Re: In C SQL Lite Datenbank einbinden](#)

Beitrag von [Meillo](#) » 17.03.2021 13:21:36

Ich finde es gut, wenn verschiedene Personen auch mit verschiedenen Wissensstaenden in dem Thema antworten. Oft kann man im Voraus nicht wissen, was dem Frangenden am besten hilft. Es gibt ja auch verschiedene Denk- und Herangehensweisen. heisenberg hat mehr Hilfe zur Selbsthilfe gegeben, waehrend ich nur mal gezeigt habe, dass der Code grundsaeztlich funktioniert. Jetzt koennte mal noch jemand vorbei kommen und die konkreten Fehlermeldungen erklaeern. Und so weben wir Stueck fuer Stueck am Wissensnetz. 😊

Bitte gerne auch weiterhin posten, wenn man nicht der Obercrack in einem Thema ist. Hier wird nun so mancher Mitleser etwas ueber den Unterschied zwischen "" und <> bei CPP-Includes gelernt haben -- ein netter Nebeneffekt, obwohl es mit der eigentlichen Frage nichts zu tun hatte. 🍷

Use ed once in a while!

[n4seweis](#)

Beiträge: [14](#)

Registriert: 06.03.2021 16:01:10

[Re: In C SQL Lite Datenbank einbinden](#)

Beitrag von [n4seweis](#) » 17.03.2021 13:24:22

So habs jetzt um zu compilieren muss das so aussehen:

sowas kann man halt auch nicht in lehrbücher schreiben ... weil wer benutzt schon Linux 😞

Danke euch allen für die Hilfe

[eggy](#)

Beiträge: [3319](#)

Registriert: 10.05.2008 11:23:50

[Re: In C SQL Lite Datenbank einbinden](#)

Beitrag von [eggy](#) » 17.03.2021 13:37:39

@Meillo: verwirr uns die armen C-Anfänger mit dem ollen make doch nicht so , die haben doch schon genug mit den fiesen Pointern in ihren Kellern zu kämpfen 🍷

Warum der Ausflug nach make? Abgesehn davon, dass Alles mit Make besser wird, der durchschnittliche C Anfänger hat sich damit noch nie beschäftigt. Und kennt, wenn überhaupt, maximal nur den "Dreisatz". Dass dann auch noch implizite Targets genutzt sind, macht die Sache für den Uneingeweihten vermutlich auch nicht verständlicher. Und das

aufgerufene cc bringt ne weitere Indirektion mit, die es möglicherweise erschwert Problemen auf den Grund zu gehen: cc ist nen Symlink auf /etc/alternatives/cc was wiederum nen Symlink auf den gcc, clang oder weitere Sysmlinks ist.

@n4seweis : beim nächsten Mal auch den benutzen Compilerbefehl gleich mitposten. Wichtig wäre in Bezug auf includes noch zu erwähnen, dass die Reihenfolge der Parameter beim gcc bei einigen Parametern wichtig ist, bei anderen nicht. Manchmal gibt man Suchpfade an und der findet die Dateien aufgrund der Reihenfolge der Parameter trotzdem nicht, dann nochmal prüfen, ob man nicht doch noch mal was undefiniert hat.

Du kannst auch noch "-o zielname" dranschreiben, damit bekommt das Binary nen sinnvollen Namen.

Das Ganze hat übrigens Nichts mit Linux oder nicht-Linux zu tun, das ist einfach wie die meisten C Compiler funktionieren: du sagst "lieber compiler" "da da da und da such nach den Sachen" "das das das das und das sind meine Quelltext" "und da wir Code von anderen benutzen, brauchen wir die Übergabepunkte und zwar die die und die" "und heißen soll das ganze dann so' ... in der Regel wird das bei Grafischen Programmierungsumgebungen hinter bunten Knöpfen versteckt, ist aber auch nichts anderes als der entsprechende Compileraufruf auf der Shell. (Und den kann man der Faulheit halber in nem Makefile verstecken, aber dazu demnächst mehr in diesem Theater - aka Suchmaschine - oder nochmal nachfragen)



[Meillo](#)

Moderator

Beiträge: [7642](#)

Registriert: 21.06.2005 14:55:06

Wohnort: Balmora

Kontakt:

[Re: In C SQL Lite Datenbank einbinden](#)

Beitrag von [Meillo](#) » 17.03.2021 14:11:44

[eggy](#) hat geschrieben: ↑

17.03.2021 13:37:39

@Meillo: verwirrt uns die armen C-Anfänger mit dem ollen make doch nicht so , die haben doch schon genug mit den fiesen Pointern in ihren Kellern zu kämpfen



Warum der Ausflug nach make?

Gute Frage! Die Antwort darauf ist wohl: Weil ich nicht darueber nachgedacht habe und einfach nur das gepostet habe was ich bei mir gemacht habe.

Abgesehen davon, dass Alles mit Make besser wird,

... make ist zwar schrecklich, aber du hast ja so recht!

der durchschnittliche C Anfänger hat sich damit noch nie beschäftigt. Und kennt, wenn überhaupt, maximal nur den "Dreisatz". Dass dann auch noch implizite

Targets genutzt sind, macht die Sache für den Uneingeweihten vermutlich auch nicht verständlicher. Und das aufgerufene cc bringt ne weitere Indirektion mit, die es möglicherweise erschwert Problemen auf den Grund zu gehen: cc ist nen Symlink auf /etc/alternatives/cc was wiederum nen Symlink auf den gcc, clang oder weitere Sysmlinks ist.

Dein Einwand, dass da eine Menge im Hintergrund passiert, ist berechtigt. Ich habe mich halt daran gewoehnt, weil es im einfachsten Fall mit make halt sooo bequem ist:

... und schon fertig. 😊 (Das Ergebnis heisst dann auch nicht `a.out`, sondern `foo'.)

Aber ich stimme dir zu, hier haette ich besser den gcc-Befehl hingeschrieben, anstatt make zu verwenden.

Use ed once in a while!
