

Database and Programming
Project SQLite 1-7
2017.04.12



Lillebaelt Academy of
Professional Higher Education

Author:
Milan Kristof Vince:
mila1025@edu.eal.dk

April, 16, 2017

Table of Contents

1. Introduction	3
1/b Why should a Network specialist know about SQLite?.....	3
2. Download the Code	4
3 Command Line Shell For SQLite.....	5
3.1 Starting sqlite3, creating a database and how to exit the program.	6
4 Entering SQL.....	6
4.1 CREATE a table, INSERT and SELECT	6
4.2 Some data must be inserted	7
4.3 Let's format the table	7
5. Special dot command in SQLite3.....	7
6. Writing results to a file	9
7. Recovery.....	10

1. Introduction

The document contains the first seven exercise of the Project SQLite assignment and their solution. The audience is my lecturer and any of my classmates who is interested in to check. SQLite is a relational database management system. In contrast to many other database management systems, SQLite is not a client–server database engine. It has an embedded SQL database engine. A complete SQL database with multiple tables, indices, triggers, and views, is contained in a single disk file.

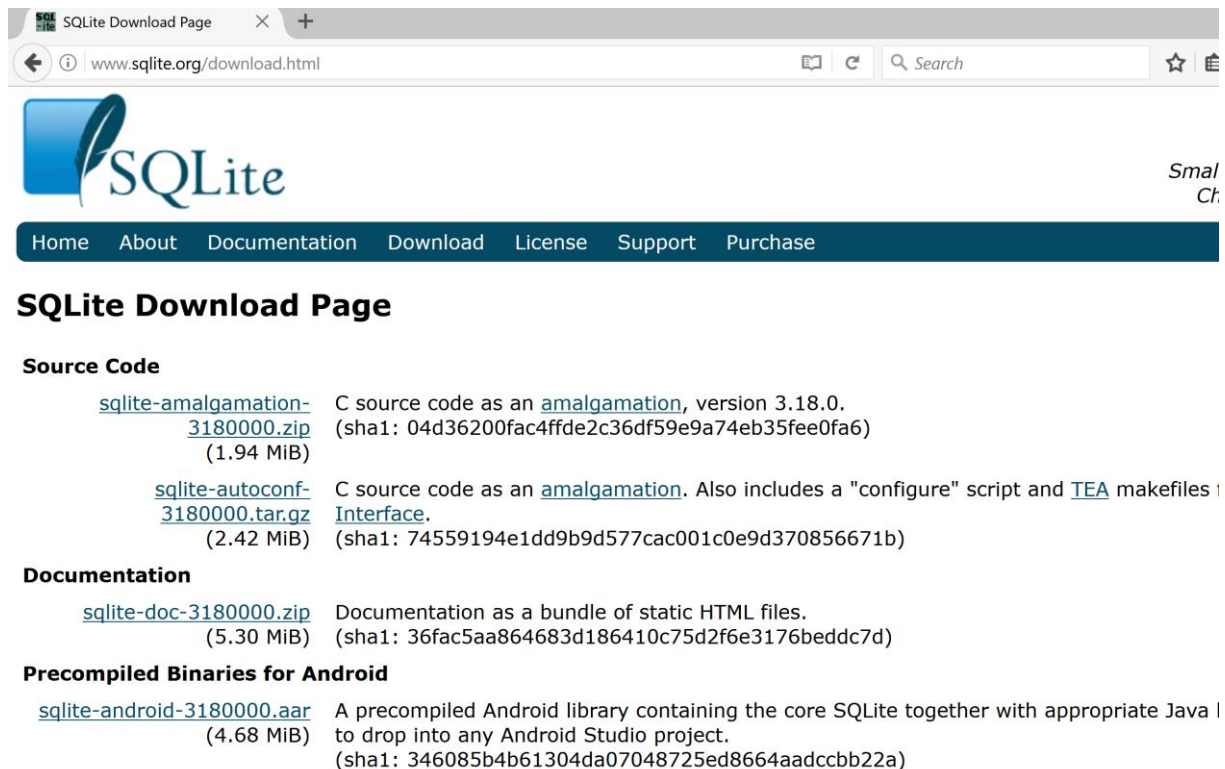
1/b Why should a Network specialist know about SQLite?

- As Network specialist you will come across users using SQLite in their applications.
- netSQLite is a client/server solution running over TCP/IP with SQLite3 at its core. As a Network specialist you may come across and have to assist troubleshoot this traffic.

2. Download the Code

”Show and describe how to get and install SQLite.”

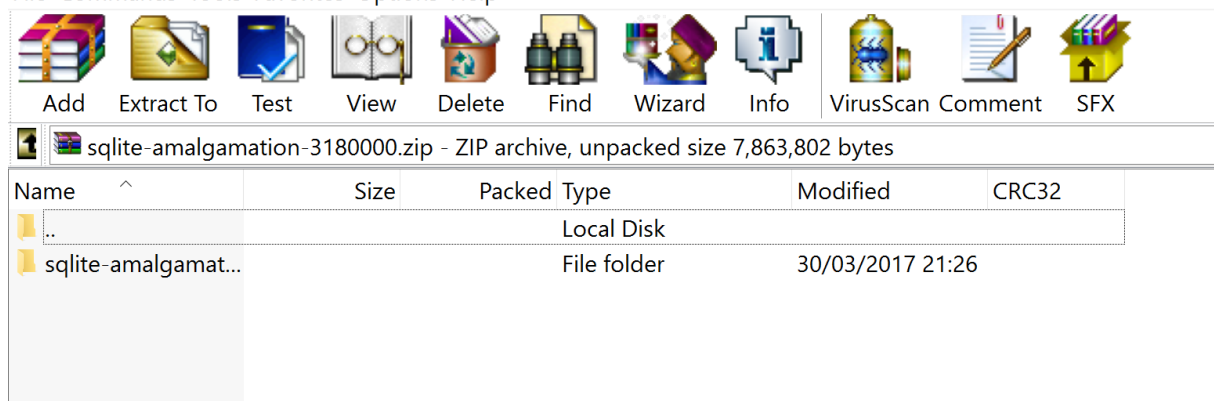
- Let's download one of the prebuilt binaries for your machine, or get a copy of the sources. Both are available at the official website of SQLite in download section:
<http://www.sqlite.org/download.html>



The screenshot shows the SQLite Download Page in a web browser. The page has a dark blue header with the SQLite logo and navigation links: Home, About, Documentation, Download, License, Support, and Purchase. Below the header, the page title is "SQLite Download Page". Under the "Source Code" section, there are two download options: "sqlite-amalgamation-3180000.zip" (1.94 MiB) and "sqlite-autoconf-3180000.tar.gz" (2.42 MiB). Both are described as C source code as an amalgamation, version 3.18.0, with their respective SHA1 hashes. The "Documentation" section offers "sqlite-doc-3180000.zip" (5.30 MiB) as a bundle of static HTML files. The "Precompiled Binaries for Android" section provides "sqlite-android-3180000.aar" (4.68 MiB) as a precompiled Android library containing the core SQLite together with appropriate Java code to drop into any Android Studio project. The SHA1 hash for the AAR file is also provided.

sqlite-amalgamation-3180000.zip - WinRAR (evaluation copy)

File Commands Tools Favorites Options Help



The WinRAR interface shows the contents of the "sqlite-amalgamation-3180000.zip" file. The file is a ZIP archive with an unpacked size of 7,863,802 bytes. The interface displays a list of files and folders extracted from the archive. The list includes a folder named "sqlite-amalgamat..." and a file named "sqlite-amalgamat...". The table below shows the details of the extracted files and folders.

Name	Size	Packed	Type	Modified	CRC32
..			Local Disk		
sqlite-amalgamat...			File folder	30/03/2017 21:26	

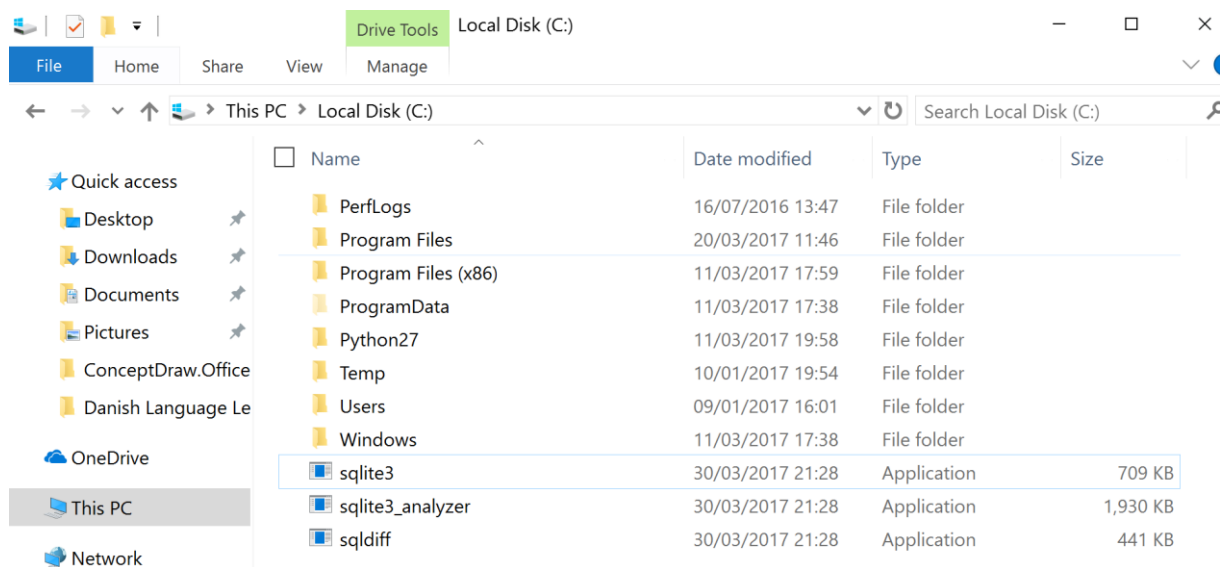
- Place the amalgamation in one easy accessible place.

mila1025@edu.eal.dk

3 Command Line Shell For SQLite

” Describe what the command-line utility named sqlite3, or sqlite.exe is.”

- The SQLite library includes a simple command - line utility named sqlite3, or sqlite3.exe on windows, that allows the user to manually enter and execute SQL commands against an SQLite database.
- In this example the amalgamation unpacked to the root directory. C://

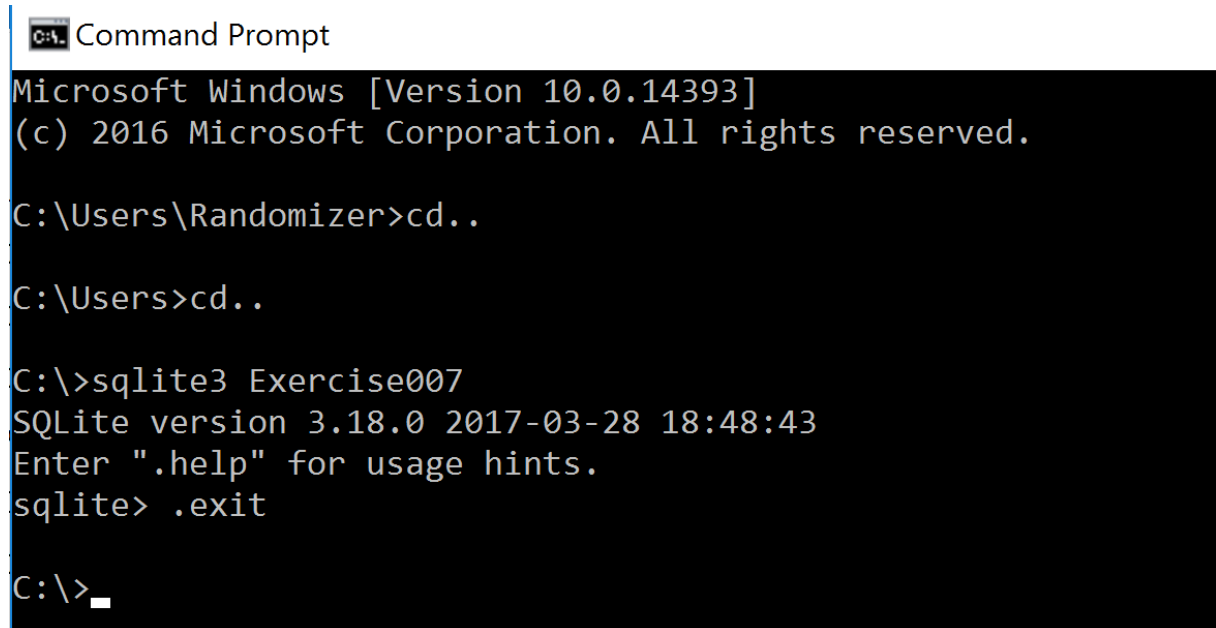


- Having sqlite3 in the root directory is convenient for testing as it saves typing a long directory path. For production, sqlite3 should be place in a project folder. Basically when the user closes the program it takes the database-file to the same folder where sqlite3 is.

3.1 Starting sqlite3, creating a database and how to exit the program.

„ Show and describe how to start sqlite3 and how to create a database ”

- To start the program the user needs to open Command Prompt and type "sqlite3" followed by the name of the file that holds the SQLite Database. If the file does not exist, a new one will be created automatically.
- To close the program type: .exit
- You can also use Ctrl + C to exit sqlite3.



```
C:\> Command Prompt
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Users\Randomizer>cd..

C:\Users>cd..

C:\>sqlite3 Exercise007
SQLite version 3.18.0 2017-03-28 18:48:43
Enter ".help" for usage hints.
sqlite> .exit

C:\>_
```

4 Entering SQL

4.1 CREATE a table, INSERT and SELECT

” Show and describe how to CREATE the shown customerTable and INSERT and SELECT data. ”

- In order to create new table in the database, you should type in the following:

```
CREATE TABLE customerTable
```

4.2 Some data must be inserted

```
INSERT INTO customerTable (idCust, name, email, Address, city)
VALUES
(1,'Per', 'pda@eal.dk', 'Mystreet 1 ', 'Odense'),
(2,'Ole', 'gov@gmail.com', 'Allstreet 2 ', 'Aalborg'),
(3,'Jesper', 'jes@hotmail.com', 'Weststreet 3 ', 'London');
```

- Now you can check your new table and its contents by the following command:
select * from customerTable

```
sqlite> select * from customerTable;
1|Per|pda@eal.dk|Mystreet 1 |Odense
2|Giovanni|gov@gmail.com|Allstreet 2 |Aalborg
3|Jesper|jes@hotmail.com|Weststreet 2 |London
```

4.3 Let's format the table

```
sqlite> .mode column
sqlite> .headers on
```

```
sqlite> .mode column
sqlite> .headers on
sqlite> select * from customerTable;
idCust      name      email      address      city
-----
1           Per      pda@eal.dk  Mystreet 1   Odense
2           Giovanni  gov@gmail.  Allstreet 2  Aalborg
3           Jesper    jes@hotmai  Weststreet   London
```

5. Special dot command in SQLite3

” Show and describe what the special dot commands are. ”

- Most of the time, sqlite3 just reads lines of input and passes them on to the SQLite library for execution. But if an input line begins with a dot ("."), then that line is intercepted and interpreted by the sqlite3 program itself. These "dot commands" are typically used to change the output format of queries, or to execute certain prepackaged query statements.

- To check the list of the available dot commands, you can enter ".help" any time. For example:

```

sqlite> .help
.auth ON|OFF          Show authorizer callbacks
.backup ?DB? FILE      Backup DB (default "main") to FILE
.bail on|off           Stop after hitting an error.  Default OFF
.binary on|off         Turn binary output on or off.  Default OFF
.changes on|off        Show number of rows changed by SQL
.check GLOB            Fail if output since .testcase does not match
.clone NEWDB           Clone data into NEWDB from the existing database
.databases             List names and files of attached databases
.dbinfo ?DB?          Show status information about the database
.dump ?TABLE? ...      Dump the database in an SQL text format
                        If TABLE specified, only dump tables matching
                        LIKE pattern TABLE.

.echo on|off           Turn command echo on or off
.eqp on|off|full       Enable or disable automatic EXPLAIN QUERY PLAN
.exit                 Exit this program
.explain ?on|off|auto? Turn EXPLAIN output mode on or off or to automatic
.fullschema ?--indent? Show schema and the content of sqlite_stat tables
.headers on|off        Turn display of headers on or off
.help                 Show this message
.import FILE TABLE    Import data from FILE into TABLE
.imposter INDEX TABLE Create imposter table TABLE on index INDEX
.indexes ?TABLE?       Show names of all indexes
                        If TABLE specified, only show indexes for tables
                        matching LIKE pattern TABLE.
.limit ?LIMIT? ?VAL?   Display or change the value of an SQLITE_LIMIT
.lint OPTIONS          Report potential schema issues. Options:
                        fkey-indexes    Find missing foreign key indexes

.load FILE ?ENTRY?     Load an extension library
.log FILE|off          Turn logging on or off.  FILE can be stderr/stdout
.mode MODE ?TABLE?     Set output mode where MODE is one of:
                        ascii          Columns/rows delimited by 0x1F and 0x1E
                        csv            Comma-separated values
                        column        Left-aligned columns.  (See .width)
                        html          HTML <table> code
                        insert        SQL insert statements for TABLE
                        line          One value per line
                        list          Values delimited by "|"
                        quote         Escape answers as for SQL
                        tabs          Tab-separated values
                        tcl           TCL list elements

.nullvalue STRING      Use STRING in place of NULL values
.once FILENAME         Output for the next SQL command only to FILENAME
.open ?--new? ?FILE?   Close existing database and reopen FILE
                        The --new starts with an empty file
.output ?FILENAME?     Send output to FILENAME or stdout
.print STRING...       Print literal STRING
.prompt MAIN CONTINUE  Replace the standard prompts
.quit                 Exit this program
.read FILENAME         Execute SQL in FILENAME

```



```

.restore ?DB? FILE      Restore content of DB (default "main") from FILE
.save FILE              Write in-memory database into FILE
.scanstats on|off       Turn sqlite3_stmt_scanstatus() metrics on or off
.schema ?PATTERN?       Show the CREATE statements matching PATTERN
                        Add --indent for pretty-printing
.selftest ?--init?      Run tests defined in the SELFTEST table
.separator COL ?ROW?    Change the column separator and optionally the row
                        separator for both the output mode and .import
.sha3sum ?OPTIONS...?   Compute a SHA3 hash of database content
.shell CMD ARGS...      Run CMD ARGS... in a system shell
.show                  Show the current values for various settings
.stats ?on|off?         Show stats or turn stats on or off
.system CMD ARGS...     Run CMD ARGS... in a system shell
.tables ?TABLE?         List names of tables
                        If TABLE specified, only list tables matching
                        LIKE pattern TABLE.
.testcase NAME          Begin redirecting output to 'testcase-out.txt'
.timeout MS            Try opening locked tables for MS milliseconds
.timer on|off          Turn SQL timer on or off
.trace FILE|off        Output each SQL statement as it is run
.vfsinfo ?AUX?         Information about the top-level VFS
.vfslist               List all available VFSes
.vfsname ?AUX?         Print the name of the VFS stack
.width NUM1 NUM2 ...   Set column widths for "column" mode
                        Negative values right-justify
sqlite>

```

6. Writing results to a file

” Show and describe how to send result to a file.”

```

C:\>sqlite3 Exercise007
SQLite version 3.18.0 2017-03-28 18:48:43
Enter ".help" for usage hints.
sqlite> .mode list
sqlite> .separator |
sqlite> .output testfile1.txt
sqlite> select * from customerTable;
sqlite> .exit

```

- One txt file will be created which contains the table of the database file.
- The file location is the same where the database file is.

7. Recovery

” Demonstrate how to back up and recover your data base.”

- To make a backup copy of the database, simply do a "dump" and redirect the results to a file.

```
sqlite3 sample.db .dump > sample.bak
```

- Restoring the database from a backup is just as easy as backing up, except we must make sure the destination database is empty first. Alternatively you may want to delete or rename the destination database and let sqlite create a new one for you.

```
mv sample.db sample.db.old  
sqlite3 sample.db < sample.bak
```

- After restoring verify the results.

