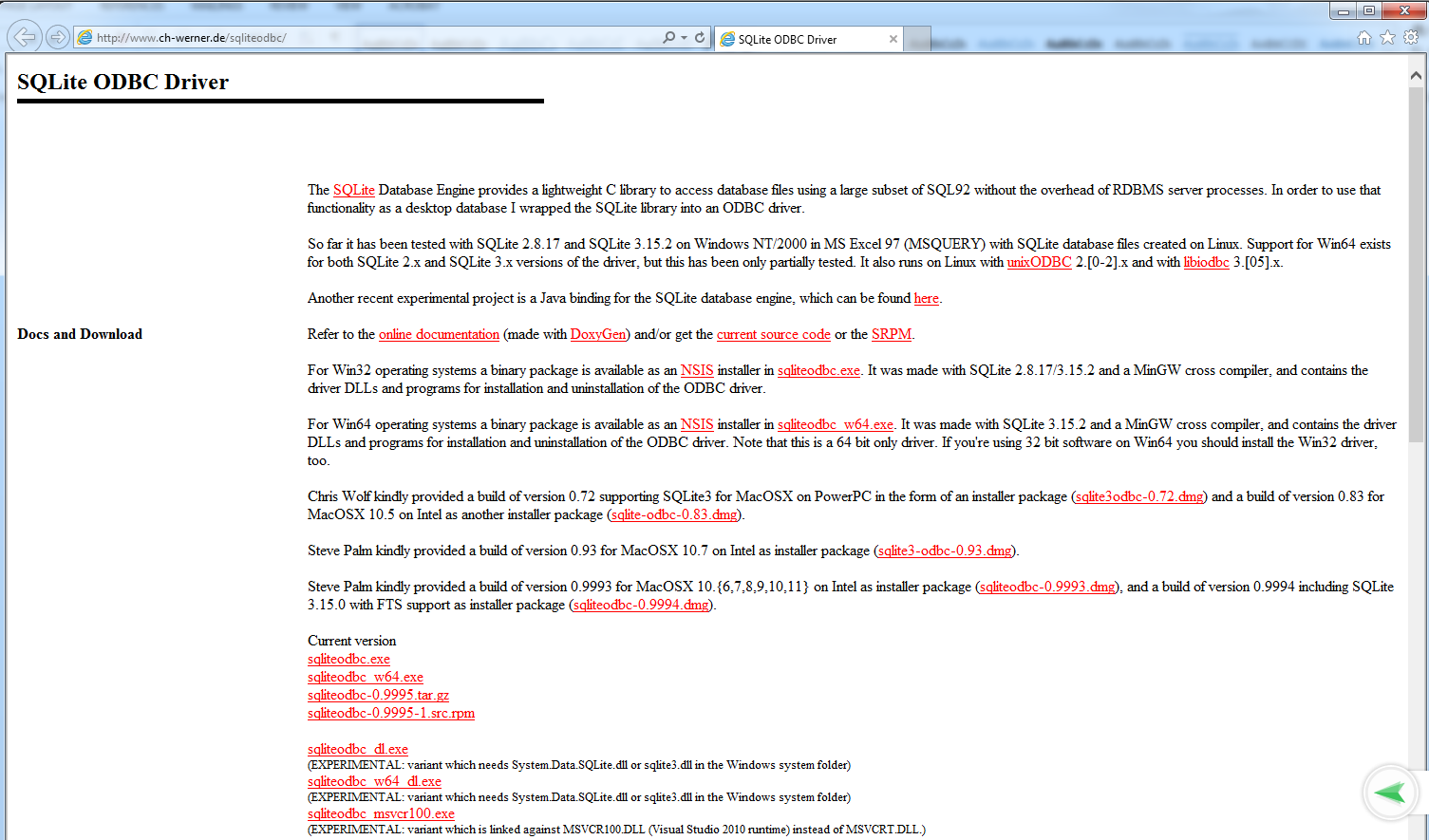
# Setup to Connect to the Database Used by FlightGUI.exe

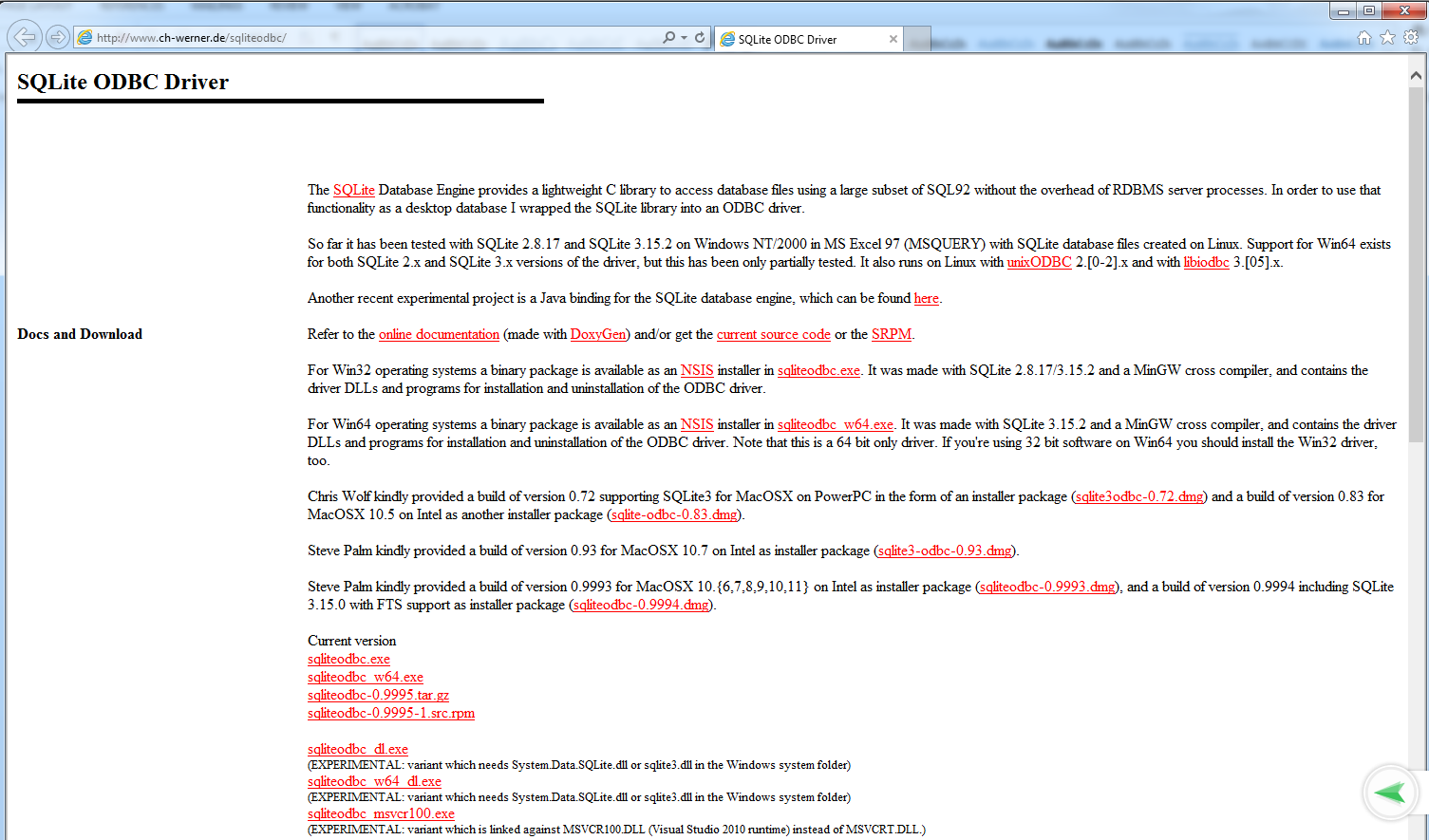
FlightGUI uses a SQLite database. Again, there may be other solutions, but the following works.

If you downloaded this from the git repository, the media files were also download. If you are doing this “stand alone”, start at this URL:

<http://www.ch-werner.de/sqliteodbc/>

Download both [sqliteodbc.exe](http://www.ch-werner.de/sqliteodbc/sqliteodbc.exe) *and* [sqliteodbc\_dl.exe](http://www.ch-werner.de/sqliteodbc/sqliteodbc_dl.exe) as indicated.



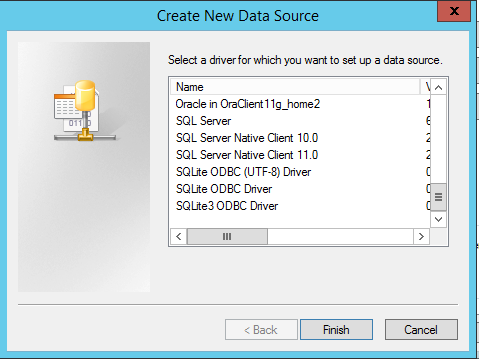


Similar to Oracle, the must both be placed, and run from the same directory. After placing, **you must run these executables in order to actually install the software**

Start the 32-bit ODBC Administrator (Section 1). (run:  %systemdrive%\Windows\SysWoW64\odbcad32.exe

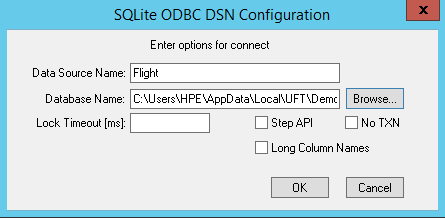
You might want to also download a GUI tool to work with sqlite databases. One possible place is: xhttps://github.com/sqlitebrowser/sqlitebrowser/releases

Add a data source. Scroll to the bottom. Choose SQLite3 ODBC driver, **not** SQLite (which is the 64-bit driver).



The following window appears.

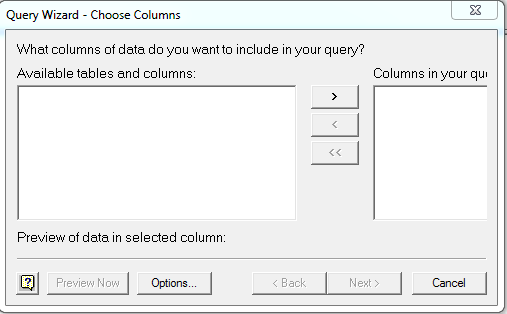
* Data Source Name is whatever you want, and will appear as a data source within UFT.
* The Database Name is C:\Users\<your login>\AppData\Local\UFT\Demo\DB\Flights.db
* Note that the Data Source Name is hard coded in the script, to be sure to use exactly the string “Flight” as shown below



Note – within UFT, if you create a new database checkpoint or output value, within Microsoft Query, you will see:



To fix this, hit OK, then select Options…



Enable System Tables by clicking the box

