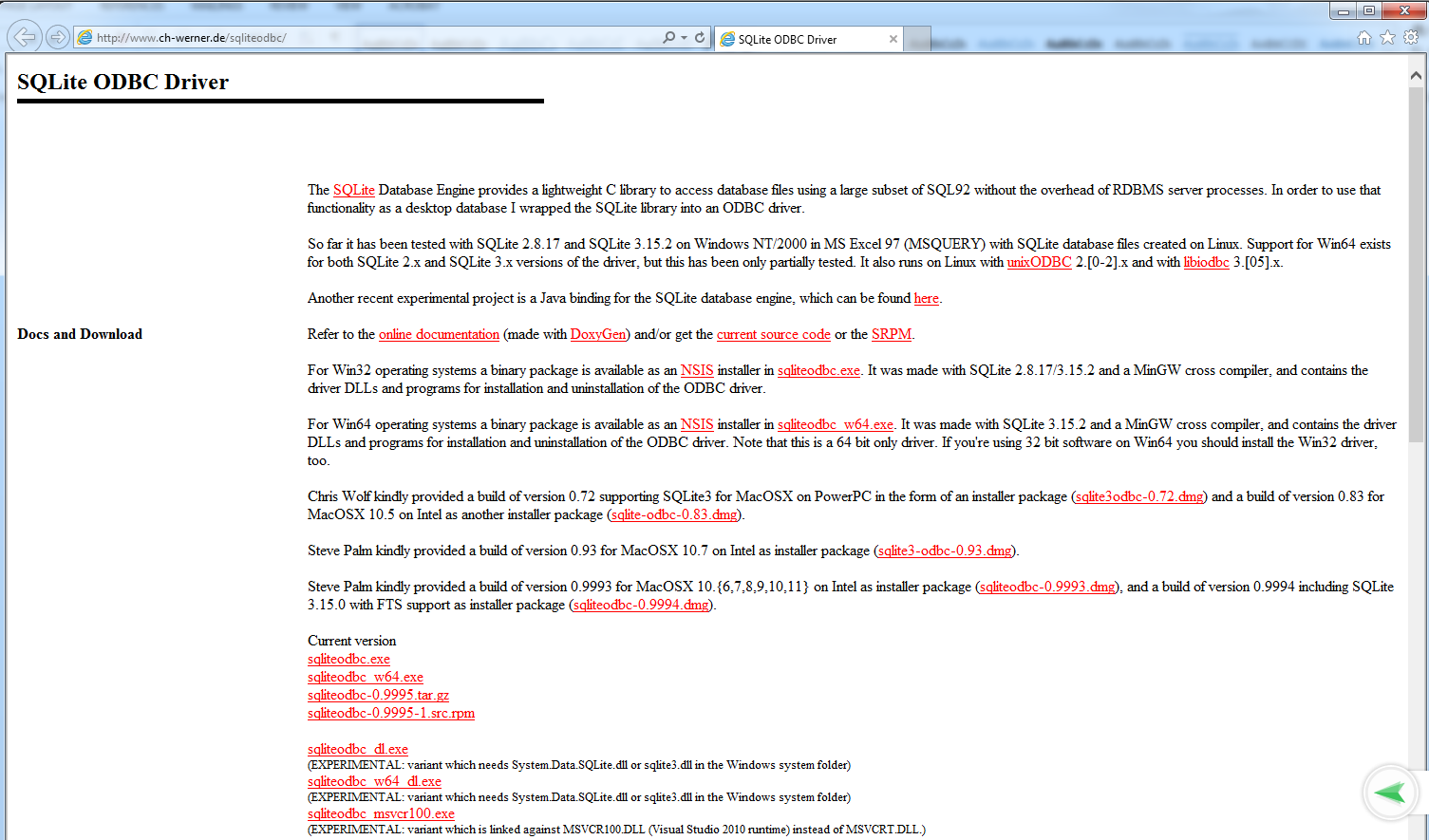
# Setup to Connect to the Database Used by FlightGUI.exe

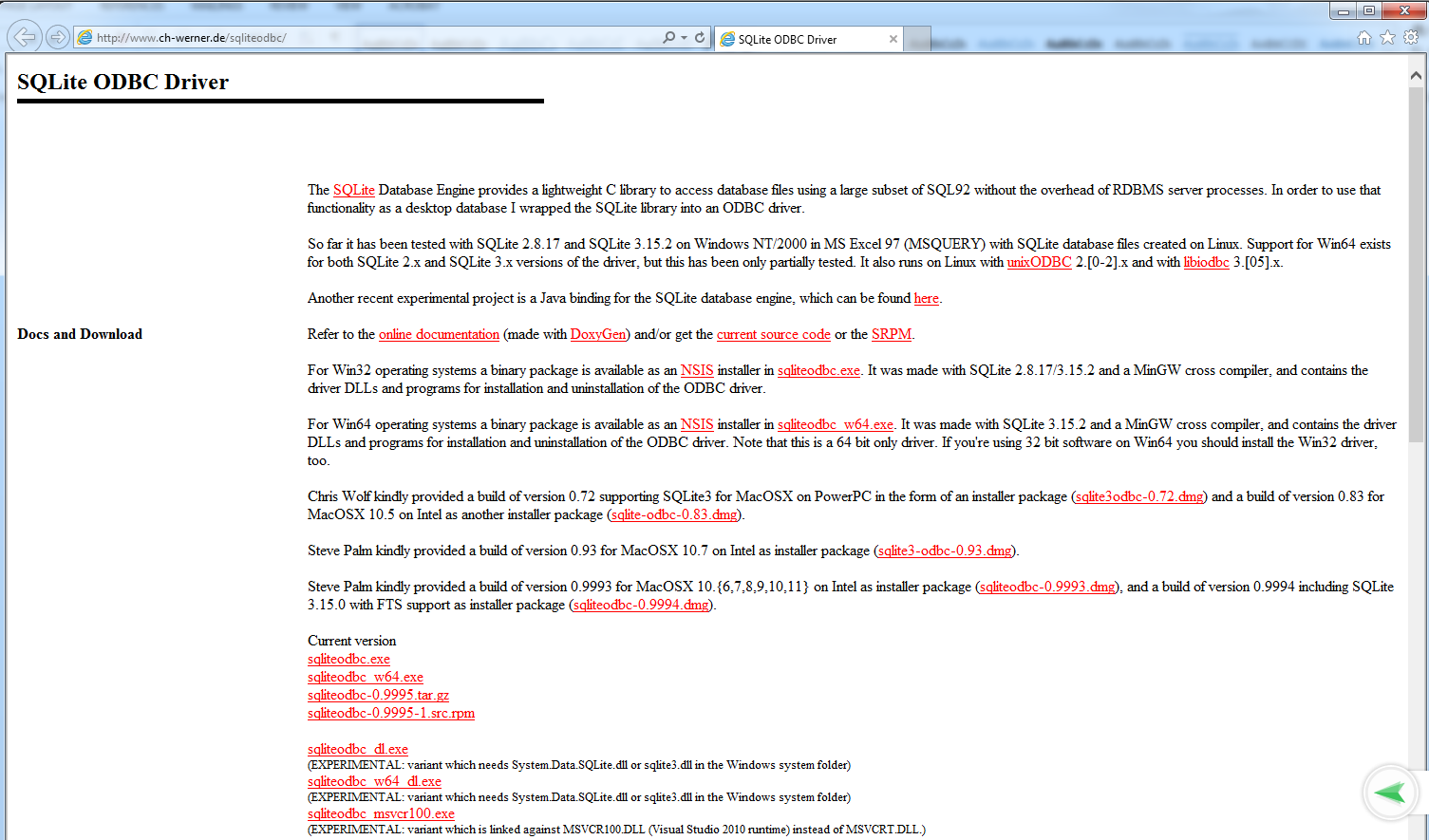
FlightGUI uses a SQLite database. Again, there may be other solutions, but the following works.

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.





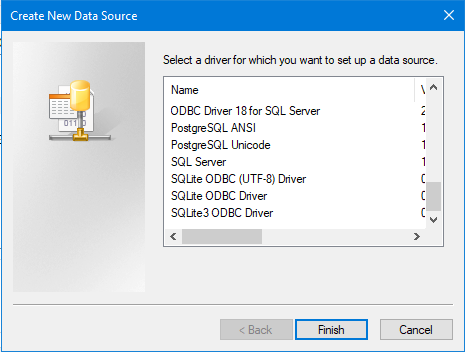
Similar to Oracle, they must both be placed, and run from the same directory. After placing, **you must run sqliteodbc\_dll.exe first, then sqliteodbc.exe.**

You might want to also download a GUI tool to work with SQLite databases. One possible place is: <https://github.com/sqlitebrowser/sqlitebrowser/releases>

Start the 32-bit ODBC Administrator (Section 1).

From a CMD prompt, type:  %systemdrive%\Windows\SysWoW64\odbcad32.exe

In the ODBC Data Source Administrator (32-bit) window and the System DSN tab, click the Add… button to add a data source. Scroll to the bottom. Choose **SQLite3 ODBC driver**, **not** SQLite (which is the 64-bit driver).



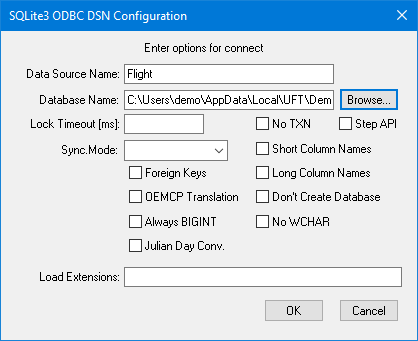
The following window appears.

* Note that the Data Source Name is hard coded in the script, so be sure to use exactly the string “Flight” as shown below
* The Database Name is probably

C:\Users\demo\AppData\Local\UFT\Demo\DB\Flights.s3db

But you should use the Browse control to verify.

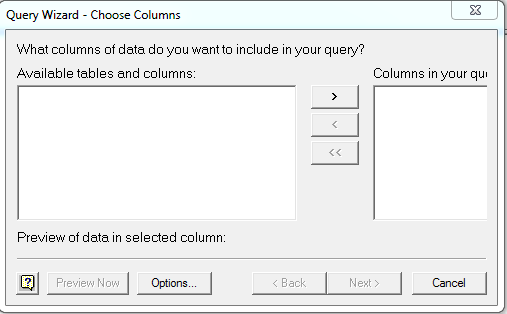
* Click **OK** to add this data source name (DSN).



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

