SOFTWARE

CONVERT MS SQL SERVER DATABASE TO SQLITE

23.11.2022

It took me basically 2 days to figure that out as most hits at Google were misleading.
My recommendation now is to install Docker and use a 2019 version of MS SQL Server.

```
docker pull mcr.microsoft.com/mssql/server:2019-latest
docker run -e "ACCEPT_EULA=Y" -e
"MSSQL_SA_PASSWORD=you_should_know_it" -e "MSSQL_PID=Express" -p
1433:1433 --name sql1 --hostname sql1 -d
mcr.microsoft.com/mssql/server:2019-latest
sudo docker ps -a
# gives you the id like bea83... below that is needed to copy the mdf
file to the container
docker cp /Users/you_should_know/database.mdf
bea8369066ed:/var/opt/mssql/data
docker cp /Users/you_should_know/database_log.ldf
bea8369066ed:/var/opt/mssql/data
```

Login as su and change permission of the uploaded file. Afterwards attach the database

```
sudo docker exec -u 0 -it sql1 "bash"
chmod 777 /var/opt/mssql/data/data*.*
/opt/mssql-tools/bin/sqlcmd -S localhost -U sa -P "you_should_know_it"
-Q "CREATE DATABASE [new_data_base] ON (FILENAME =
N'/var/opt/mssql/data/database.mdf'),(FILENAME =
N'/var/opt/mssql/data/database_log.ldf') FOR ATTACH"
```

Cool, on localhost:1443 we can now see MS SQL Server working..
Connecting with Azure was a dead end, also with SQL Pro Studio as the export could not be read by SQLITE.

Razor SQL worked (do not use DB Tools -> Export but DB Tools -> Database Conversion). This generates a sql file that SQLITE will understand.

```
cat run.sql | sqlite3 database.db
```