

Databases

Joseph Hallett

January 25, 2023



What's all this about?

When you write a program all your data disappears after the program ends

- ▶ Unless we save it somewhere

SQL Databases are a sensible choice for where to save your data

- ▶ Highly optimized storage of tabular data
- ▶ Fast and well understood query language
- ▶ Fault tolerant protocols

So what is a database?

Super fancy spreadsheet

- ▶ Each database will contain *tables* that store data
- ▶ Data in tables can be *queried* using a language called SQL
- ▶ Data in tables can be *joined* with data in other tables to answer questions

Designing them so you don't tie yourself in knots is tricky!

So why not just use Spreadsheets?

- ▶ See Matt Parker's excellent *Stand-up Maths* video: UK Government loses data because of Excel mistake.



<https://youtu.be/zUp8pkoeMss>

Different types of database

Traditionally, the database would reside on a separate machine

- ▶ Space is expensive!
- ▶ If you wanted to use the database you had to connect to it

But nowadays space is cheap

- ▶ Local per app databases very common

If you need remote data access:

- ▶ Use a *server-style* database like *MariaDB* or *MySQL*

Otherwise use a file-style database:

- ▶ ...just use *Sqlite*.

To use SQLite

Install the packages however your OS likes to install packages (apk get on Alpine)

```
$ sqlite3 database.db
-- Loading resources from /home/joseph/.sqliterc
SQLite version 3.40.1 2022-12-28 14:03:47
Enter ".help" for usage hints.
sqlite> .exit
```

(Or connect via whatever programming language you like)

Seriously, unless you're 100% sure you can't use SQLite: just use SQLite. It's great. Much simpler.

MySQL and MariaDB

If you need a server style install... try MariaDB

History

- ▶ It used to be MySQL named after the developer's kid My and the languages used to query it SQL
- ▶ It was acquired by Oracle...
- ▶ A lot of developers don't like Oracle...
- ▶ The original developer forked the open source one to make *MariaDB*
- ▶ Guess what his other kid's name is?

The command is `mysql` for both

Using MariaDB

```
$ mysql
```

```
ERROR 2002 (HY000): Can't connect to local MySQL server through socket '/var/lib/mysql/mysql.sock'
```

You need to start the server running first or say where to connect to.

On most Linux distros it'll be via SystemD:

```
systemctl start mariadb
```

```
systemctl enable mariadb
```

On Alpine Linux it'll be via OpenRC:

```
rc-service mariadb start
```

Security

Once Maria is up and running it'll have some test databases and a `root` user with no password

- ▶ Up to you to secure it
- ▶ `mysql_secure_installation` can automate most of it...

But if someone is paying you money to do it:

- ▶ Set usernames and passwords
- ▶ Firewall off ports
- ▶ Add logging and intrusion detection
- ▶ Backup
- ▶ Secure backups
- ▶ Have a get out plan...

Otherwise you'll have your database stolen!
(And potentially a *very* large regulatory fine)

Conclusion: Should I use a database?

Am I being paid to store/process this data?

- ▶ Yes? Use a database.
- ▶ No? Use a spreadsheet (or a database)

Does the data need to be accessed remotely?

- ▶ Yes? Use a server-style database (MySQL/MariaDB)
- ▶ No? Use a file-style database (SQLite)

Am I just playing with data or is my data tiny (gigabytes in size)?

- ▶ Yes? Use a database or plain text data storage (i.e. CSV).

Is my data *really* big (petabytes in size)?

- ▶ Yes? Use a NoSQL database (beyond scope of this course)

Does my data contain recursive data structures (i.e. lists of lists of arbitrary length)

- ▶ Yes? Use *Prolog* or *Datalog*. (or abuse a database ;-))

Aside: Pronunciation

How do you say SQL?

- ▶ *ess-kew-ell?*
- ▶ *sequel?*
- ▶ *squirrel?*