**SQLite Study Notes**

SQL also stands for Structured Query Language

SQL works is by comparing it to a spreadsheet software like Excel:

* A database is a whole spreadsheet file.
* A table is a tab/sheet in the spreadsheet, with each one being given a name.
* A column is a column in both.
* A row is a row in both.
* SQL then gives you a language for doing CRUD operations on these to produce *new tables or alter existing ones*.

***SQL only knows tables, and every operation produces tables. It either "produces" a table by modifying an existing one, or it returns a new temporary table as your data set.***

CREATE TABLE person (

id INTEGER PRIMARY KEY,

first\_name TEXT,

last\_name TEXT,

age INTEGER

);

CREATE TABLE person\_pet (

person\_id INTEGER,

pet\_id INTEGER

);

CREATE TABLE pet (

id INTEGER PRIMARY KEY,

name TEXT,

breed TEXT,

age INTEGER,

dead INTEGER

);

If you can put one row into person\_pet, can you put more than one? How would you record a crazy cat lady with 50 cats?

**SQLite3 data type?**

Open existing .db file

.open ex4\\ex4.db

SELECT pet.id, pet.name, pet.age, pet.dead

FROM pet, person\_pet, person

WHERE

***line1:*** pet.id = person\_pet.pet\_id AND

***line2:*** person\_pet.person\_id = person.id AND

person.first\_name = "Zed";

To connect pet to person I need to go through the person\_pet relation table. In SQL that means I need to list all three tables after the FROM.

First I connect pet to person\_pet by the related id columns pet.id and person\_pet.id in **line1.**

AND I need to connect person to person\_pet in the same way in **line2.** Now the database can search for only the rows where the id columns all match up, and those are the ones that are connected.