



Unit 02.03

## Basic Structured Query Language: Single Table CRUD

---

### USING DATABASES WITH PYTHON

Charles Severance Ph.D.

Clinical Associate Professor of Information  
School of Information



© 2015 Charles Severance and The Regents of the University of Michigan  
Except where otherwise noted, this work is licensed under  
<http://creativecommons.org/licenses/by-nc/3.0/>



# SQLite Browser

- SQLite is a very popular database - it is free and fast and small
- SQLite Browser allows us to directly manipulate SQLite files
  - <http://sqlitebrowser.org/>

There is also a Firefox plugin to manipulate SQLite database

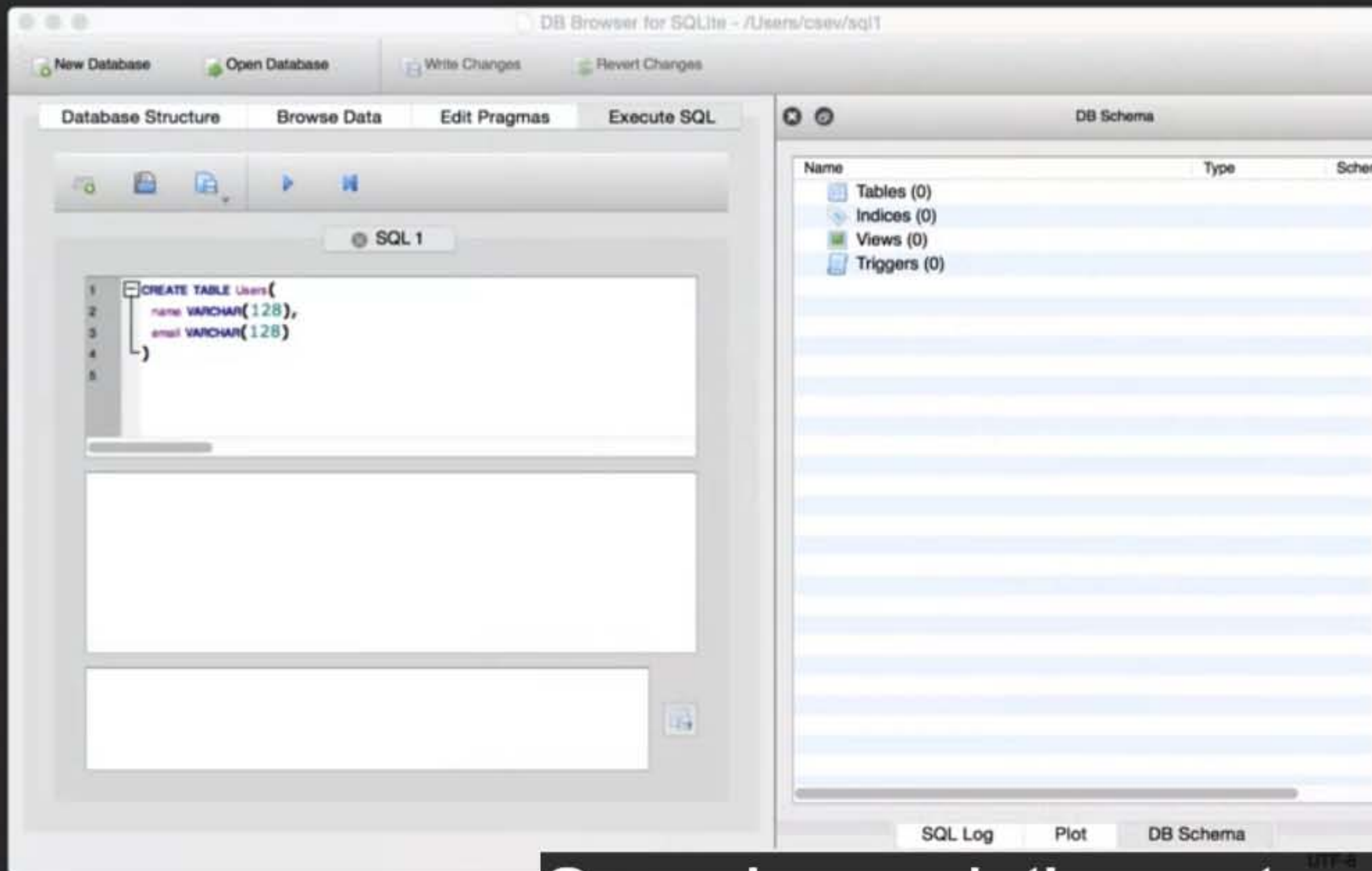
- <https://addons.mozilla.org/en-US/firefox/addon/sqlite-manager/>

SQLite is embedded in Python and a number of other languages

I've told you a couple of times, installed the SQLite browser. If you haven't, do it.



# Start Simple - A Single Table



```
CREATE TABLE Users(  
    name VARCHAR(128),  
    email VARCHAR(128)  
)
```

So we're up, let's create a table.

## 02.03 Basic Structured Query Language: Single Table CRUD



The screenshot shows the DB Browser for SQLite application. The main window is titled "DB Browser for SQLite - /Users/csev/sql1". It has a menu bar with "New Database", "Open Database", "Write Changes", and "Revert Changes". Below the menu bar are four tabs: "Database Structure", "Browse Data", "Edit Pragmas", and "Execute SQL". The "Browse Data" tab is selected, showing a table named "Users" with two columns: "name" and "email". The table contains four rows of data:

	name	email
1	Chuck	csev@umich...
2	Colleen	cvl@umich.edu
3	Ted	ted@umich....
4	Sally	a1@umich.edu

At the bottom of the window, there are navigation buttons: "<", "< 0 - 0 of 0 >", ">", and ">". A "Go to:" field with the value "1" is also present.

On the right side, there is a "DB Schema" panel. It shows a tree view with "Tables (1)" expanded, showing the "Users" table. The schema for the "Users" table is displayed as:

```
CREATE TABLE Users(  
  name VARCHAR(128),  
  email VARCHAR(128)  
)
```

Below the schema, it shows "Indices (0)", "Views (0)", and "Triggers (0)".

At the bottom right of the window, there are tabs for "Log", "Plot", and "DB Schema". The "DB Schema" tab is selected. The encoding is set to "UTF-8".

**Our table with four rows**

So there's your table,  
we're going to put some data into it.



# SQL Insert

- The Insert statement inserts a row into a table

```
INSERT INTO Users (name, email) VALUES ('Kristin', 'kf@umich.edu')
```

So the first thing we're going to do is insert another record.

# SQL Delete

- Deletes a row in a table based on a selection criteria

```
DELETE FROM Users WHERE email='ted@umich.edu'
```

4

Now we could delete a row.



# SQL: Update

- Allows the updating of a field with a where clause

```
UPDATE Users SET name='Charles' WHERE  
email='csev@umich.edu'
```

Update allows us to sort of reach in to a particular cell or set of cells in



# Retrieving Records: Select

- The select statement retrieves a group of records - you can either retrieve all the records or a subset of the records with a **WHERE** clause

```
SELECT * FROM Users
```

```
SELECT * FROM Users WHERE email='csev@umich.edu'
```



# Sorting with ORDER BY

- You can add an **ORDER BY** clause to **SELECT** statements to get the results sorted in ascending or descending order

```
SELECT * FROM Users ORDER BY email
```

```
SELECT * FROM Users ORDER BY name
```

You can also throw another clause on the end of a SELECT.



# SQL Summary

```
INSERT INTO Users (name, email) VALUES ('Kristin', 'kf@umich.edu')
```

```
DELETE FROM Users WHERE email='ted@umich.edu'
```

```
UPDATE Users SET name="Charles" WHERE email='csev@umich.edu'
```

```
SELECT * FROM Users
```

```
SELECT * FROM Users WHERE email='csev@umich.edu'
```

```
SELECT * FROM Users ORDER BY email
```