

Introduction to sqlite

- public domain relational database system
- light weight, no server required
- no configuration, simple to set up
- cross platform, easy to move data files
- files void 'below' large, complex database applications

Structured Query Language

- SQL is a language
 - Data Definition: control structure
 - Data Manipulation: change data values
 - Transaction Control: execution of 'transactions'
 - Data Control: regulate access
- important commands:
 - CREATE TABLE
 - INSERT
 - SELECT
- semicolon indicates end of line/command
- comments: single line '--', multiple line /* */
- numbers (raw) and text (single quote)

Creating Database File in Python

```
import sqlite3 as sql
import os

os.remove('sample1.db')

#make database file and object
dbase=sql.connect('sample1.db')
cursor = dbase.cursor()
#execute an sql command, create table
cursor.execute('CREATE TABLE data (id INTEGER PRIMARY KEY,city,area,population)')
#add data
cursor.execute('INSERT INTO data VALUES (null,?,?,?)',('Bangor',35,35500))
cursor.execute('INSERT INTO data VALUES (null,?,?,?)',('Austin',296,790400))
cursor.execute('INSERT INTO data VALUES (null,?,?,?)',('StLouis',66,319300))
cursor.execute('INSERT INTO data VALUES (null,?,?,?)',('Portland',53,66200))
cursor.execute('INSERT INTO data VALUES (null,?,?,?)',('Boston',90,617600))
cursor.execute('INSERT INTO data VALUES (null,?,?,?)',('Pittsburgh',58,305700))
dbase.commit()
```

Using Database File in Python

```
import sqlite3 as sql

dbase=sql.connect('sample1.db')
cursor = dbase.cursor()
##get table names
cursor.execute('select name from sqlite_master')
tableNames=cursor.fetchall()
print tableNames
cursor.execute('select * from data')
print cursor.description
cursor.execute('select distinct city from data')
print cursor.fetchall()
cursor.execute('select city from data where population>? and area>? order by area', (1.e5,40))
print cursor.fetchall()
dbase.close()
```