

# SQLite 데이터베이스



JEONJU UNIVERSITY

## 데이터베이스

### ➤ 데이터베이스 관련 용어

- DBMS : SQLite, Oracle, Sql Server 등
- 테이블 :
- 행(로우; 레코드)
- 열(컬럼; 필드)

## 파이썬에서 Sqlite 사용하기

### ➤ Sqlite 버전 알아보기

```
In [10]: # API : https://docs.python.org/ko/3/library/sqlite3.html
import sqlite3
print(sqlite3.version)

2.6.0
```

### ➤ Sqlite 데이터베이스 접속 : 메모리에 임시로 생성

```
In [11]: # SQLite 데이터베이스에 접속하기
conn = sqlite3.connect(":memory:")
print(type(conn))
print(sqlite3.Connection)

cursor = conn.cursor()

<class 'sqlite3.Connection'>
<class 'sqlite3.Connection'>
```

### ➤ Sqlite 데이터베이스 종료

```
In [12]: # 커서 닫기, 연결 끊기
cursor.close()
conn.close()
```

The Place for Superstars

## 파이썬에서 Sqlite 사용하기

### ➤ 테이블 생성하기

```
In [4]: # 테이블 생성하기
cursor.execute("CREATE TABLE addr(name text, hp text, addr text, age int)")

Out[4]: <sqlite3.Cursor at 0x1d0ce59ce30>
```

### ➤ 테이블에 값 입력하기

```
In [5]: # 테이블에 값 1개 입력하기
cursor.execute("INSERT INTO addr(name, hp, addr, age) "
              + "VALUES('홍길동', '010-1111-1111', '전라북도 전주시', 21)")
conn.commit()
```

```
In [6]: # 테이블에 값 여러개 입력하기
data = [
    ('장길동', '010-1111-1111', '전라북도 정읍시', 19),
    ('김길동', '010-2222-2222', '전라북도 익산시', 20),
    ('이길동', '010-3333-3333', '전라북도 군산시', 21),
    ('최길동', '010-4444-4444', '전라북도 김제시', 19)
]

cursor.executemany("INSERT INTO addr(name, hp, addr, age) "
                  + "VALUES(?, ?, ?, ?)", data)
conn.commit()
```

The Place for Superstars

## 파이썬에서 Sqlite 사용하기

### ▶ 테이블의 내용 읽어오기

```
In [7]: # 테이블의 내용 읽어오기
cursor.execute("SELECT * FROM addr")
result = cursor.fetchall()
result
```

```
Out[7]: [('홍길동', '010-1111-1111', '전라북도 전주시', 21),
('장길동', '010-1111-1111', '전라북도 정읍시', 19),
('김길동', '010-2222-2222', '전라북도 익산시', 20),
('이길동', '010-3333-3333', '전라북도 군산시', 21),
('최길동', '010-4444-4444', '전라북도 김제시', 19)]
```



JEONJU UNIVERSITY  
The Place for Superstars

## 파이썬에서 Sqlite 사용하기

### ▶ 테이블 값 수정하기

```
Out[6]: [('홍길동', '010-1111-1111', '전라북도 전주시', 21),
('장길동', '010-1111-1111', '전라북도 정읍시', 19),
('김길동', '010-2222-2222', '전라북도 익산시', 20),
('이길동', '010-3333-3333', '전라북도 군산시', 21),
('최길동', '010-4444-4444', '전라북도 김제시', 19)]
```

```
In [8]: # 테이블 내용 수정하기
cursor.execute("UPDATE addr SET hp='010-9999-9999', age=25 "
+ " WHERE name='홍길동'")
conn.commit()
```

```
In [9]: # 테이블의 내용 읽어오기
cursor.execute("SELECT * FROM addr")
result = cursor.fetchall()
result
```

```
Out[9]: [('홍길동', '010-9999-9999', '전라북도 전주시', 25),
('장길동', '010-1111-1111', '전라북도 정읍시', 19),
('김길동', '010-2222-2222', '전라북도 익산시', 20),
('이길동', '010-3333-3333', '전라북도 군산시', 21),
('최길동', '010-4444-4444', '전라북도 김제시', 19)]
```



UNIVERSITY  
The Place for Superstars

## 파이썬에서 Sqlite 사용하기

### ▶ 테이블 값 삭제하기

```
Out[9]: [('홍길동', '010-9999-9999', '전라북도 전주시', 25),
        ('장길동', '010-1111-1111', '전라북도 정읍시', 19),
        ('김길동', '010-2222-2222', '전라북도 익산시', 20),
        ('이길동', '010-3333-3333', '전라북도 군산시', 21),
        ('최길동', '010-4444-4444', '전라북도 김제시', 19)]
```

```
In [11]: # 테이블 내용 수정하기
         cursor.execute("DELETE FROM addr WHERE age=19 ")
         conn.commit()
```

```
In [12]: # 테이블의 내용 읽어오기
         cursor.execute("SELECT * FROM addr")
         result = cursor.fetchall()
         result
```

```
Out[12]: [('홍길동', '010-9999-9999', '전라북도 전주시', 25),
          ('김길동', '010-2222-2222', '전라북도 익산시', 20),
          ('이길동', '010-3333-3333', '전라북도 군산시', 21)]
```

## 파이썬에서 Sqlite 사용하기

### ▶ 명령어 여러줄을 한꺼번에 처리

```
In [7]: # 명령어를 모아서 한꺼번에 처리
         cursor.executescript("""
         CREATE TABLE phone(name text, hp text);
         INSERT INTO phone(name, hp) VALUES('홍길동', '010-1111-1111');
         INSERT INTO phone(name, hp) VALUES('김길동', '010-2222-2222');
         INSERT INTO phone(name, hp) VALUES('이길동', '010-3333-3333');
         """)
```

```
Out[7]: <sqlite3.Cursor at 0x25119523b20>
```

```
In [8]: # 테이블의 내용 읽어오기
         cursor.execute("SELECT * FROM phone")
         result = cursor.fetchall()
         result
```

```
Out[8]: [('홍길동', '010-1111-1111'), ('김길동', '010-2222-2222'), ('이길동', '010-3333-3333')]
```

## SQL 명령어 모음

- DDL
  - CREATE : 테이블 생성
  - DROP : 테이블 삭제
  - ALTER : 테이블 수정
- DML
  - INSERT : 테이블에 값 저장
  - DELETE : 테이블에서 값 삭제
  - UPDATE : 테이블에서 값 수정
- DQL
  - SELECT : 테이블에서 값 읽어오기

## SQL 명령어 모음

- DDL
  - CREATE : **CREATE TABLE** addr(name text, hp text, age int);
  - DROP : **DROP TABLE** addr;
  - ALTER : **ALTER TABLE** addr **ADD** juso text;
- DML
  - INSERT : **INSERT INTO** addr(name, hp, age) **VALUES**( 'gildong' , '010-1111-1111' , 21);
  - DELETE : **DELETE FROM** addr **WHERE** name= 'gildong' ;
  - UPDATE : **UPDATE** addr **SET** age=22 **WHERE** name= 'gildong' ;
- DQL
  - SELECT : **SELECT \* FROM** addr;
- 참조 : <http://jimoak.iwinv.net/wp/?p=13996>

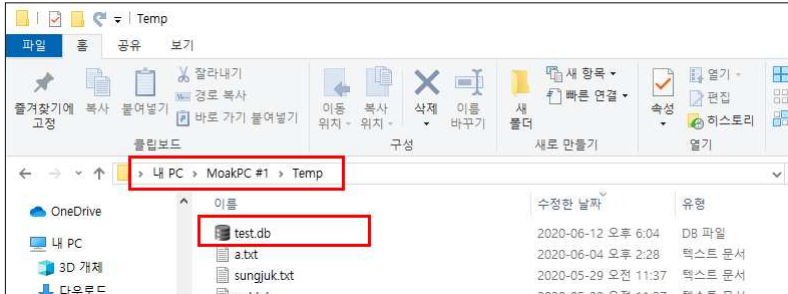
## 파이썬에서 Sqlite 사용하기

### ➤ Sqlite 데이터베이스 접속 : 실제 디스크에 생성

```
In [12]: # Sqlite 데이터베이스에 접속하기 test.db가 있으면 그냥 사용, 없으면 생성
conn = sqlite3.connect("c:/temp/test.db")
print(type(conn))
print(sqlite3.Connection)

cursor = conn.cursor()

<class 'sqlite3.Connection'>
<class 'sqlite3.Connection'>
```



The screenshot shows a Windows File Explorer window for the 'Temp' folder. The breadcrumb path is '내 PC > MoakPC #1 > Temp'. A file named 'test.db' is highlighted with a red box. Other files visible include 'a.txt' and 'sungjuk.txt'. The file details table shows the following information:


이름	수정된 날짜	유형
test.db	2020-06-12 오후 6:04	DB 파일
a.txt	2020-06-04 오후 2:28	텍스트 문서
sungjuk.txt	2020-05-29 오전 11:37	텍스트 문서

## 파이썬에서 Sqlite 사용하기

### ➤ 테이블 생성하고 값 입력하기

```
In [21]: cursor.execute("CREATE TABLE addr(name text, hp text, addr text, age int)")
Out[21]: <sqlite3.Cursor at 0x2511967f420>
```

```
In [22]: cursor.execute("INSERT INTO addr(name, hp, addr, age) VALUES('홍길동', '010-1111-1111')")
cursor.execute("INSERT INTO addr(name, hp, addr, age) VALUES('김길동', '010-2222-2222')")
cursor.execute("INSERT INTO addr(name, hp, addr, age) VALUES('이길동', '010-3333-3333')")
cursor.execute("INSERT INTO addr(name, hp, addr, age) VALUES('최길동', '010-4444-4444')")
conn.commit()
```



JEONJU UNIVERSITY  
The Place for Superstars

## 파이썬에서 Sqlite 사용하기

### 테이블의 내용 읽어오기, 종료하기


```
In [23]: cursor.execute("SELECT * FROM addr")
result = cursor.fetchone()
result
```

```
Out[23]: ('홍길동', '010-1111-1111', '전라북도 전주시', 21)
```

```
In [24]: cursor.execute("SELECT * FROM addr")
result = cursor.fetchall()
result
```

```
Out[24]: [('홍길동', '010-1111-1111', '전라북도 전주시', 21),
('김길동', '010-2222-2222', '전라북도 익산시', 20),
('이길동', '010-3333-3333', '전라북도 군산시', 21),
('최길동', '010-4444-4444', '전라북도 김제시', 19)]
```

```
In [18]: cursor.close()
conn.close()
```



JEONJU UNIVERSITY  
The Place for Superstars

## DB Browser for SQLite

### 홈페이지

- URL : <https://sqlitebrowser.org/>
- 다운로드

#### Downloads

- [DB.Browser.for.SQLite-3.12.0-rc1-win32.msi](#) - Standard installer for 32-bit Windows
- [DB.Browser.for.SQLite-3.12.0-rc1-win32.zip](#) - .zip (no installer) for 32-bit Windows
- [DB.Browser.for.SQLite-3.12.0-rc1-win64.msi](#) - Standard installer for 64-bit Windows
- [DB.Browser.for.SQLite-3.12.0-rc1-win64.zip](#) - .zip (no installer) for 64-bit Windows
- [DB.Browser.for.SQLite-3.12.0-rc1.dmg](#) - Standard package for macOS



JEONJU UNIVERSITY  
The Place for Superstars

# DB Browser for SQLite

## ▶ 실행 화면

DB Browser for SQLite

파일(F) 편집(E) 뷰(V) Tools 도움말(H)

새 데이터베이스(N) 데이터베이스 열기(O) 본문서한 저장하기(W) 본문서한 취소하기(R) Open Project Attach Database

데이터베이스 구조 데이터 보기 Pragma 수정 SQL 실행

데이터베이스 생성하기(C) 인덱스 생성하기(I) 데이터베이스 수정하기(M)

이름 타입 스키마

데이터베이스 뷰 수정하기(C)

모드: 문자열 가져오기(O) 내보내기(E) NULL로 만들기

현재 데이터 타입: 널  
0 바이트 적용

필터

이름 커밋 마지막 수정 크기

SQL 로그 플러그인 DB 스키마 원격

UTF-8 Superstars

JU UNIVERSITY