

### 13.13.3 Cursor Objects

A `Cursor` instance has the following attributes and methods:

**`execute`** (*sql*, [*parameters*])

Executes a SQL statement. The SQL statement may be parametrized (i. e. placeholders instead of SQL literals). The `sqlite3` module supports two kinds of placeholders: question marks (qmark style) and named placeholders (named style).

This example shows how to use parameters with qmark style:

```
.1.py
```

This example shows how to use the named style:

```
.2.py
```

`execute()` will only execute a single SQL statement. If you try to execute more than one statement with it, it will raise a `Warning`. Use `executescript()` if you want to execute multiple SQL statements with one call.

**`executemany`** (*sql*, *seq\_of\_parameters*)

Executes a SQL command against all parameter sequences or mappings found in the sequence *sql*. The `sqlite3` module also allows using an iterator yielding parameters instead of a sequence.

```
.1.py
```

Here's a shorter example using a generator:

```
.2.py
```

**`executescript`** (*sql\_script*)

This is a nonstandard convenience method for executing multiple SQL statements at once. It issues a `COMMIT` statement first, then executes the SQL script it gets as a parameter.

*sql\_script* can be a bytestring or a Unicode string.

Example:

```
import sqlite3

con = sqlite3.connect(":memory:")
cur = con.cursor()
cur.executescript("""
    create table person(
        firstname,
        lastname,
        age
    );

    create table book(
        title,
        author,
        published
    );

    insert into book(title, author, published)
    values (
        'Dirk Gently''s Holistic Detective Agency',
```