

Introducing SQLite

SQLite is a relational database management system (RDBMS). It is well regarded, being:

- ☐ Open source
- ☐ Standards-compliant
- ☐ Lightweight
- ☐ Single-tier

It has been implemented as a compact C library that's included as part of the Android software stack.

By providing functionality through a library, rather than as a separate process, each database becomes an integrated part of the application that created it. This reduces external dependencies, minimizes latency, and simplifies transaction locking and synchronization.

SQLite has a reputation of being extremely reliable and is the database system of choice for many consumer electronic devices, including several MP3 players, the iPhone, and the iPod Touch.

Lightweight and powerful, SQLite differs from many conventional database engines by using a loosely typed approach to column definitions. Rather than requiring column values to conform to a single type, the values in each row for each column are individually typed. As a result, there's no strict type checking when assigning or extracting values from each column within a row.

For more comprehensive coverage of SQLite, including its particular strengths and limitations, check out the official site at www.sqlite.org/.

Cursors and Content Values

ContentValues objects are used to insert new rows into database tables (and Content Providers). Each ContentValues object represents a single row, as a map of column names to values.

Queries in Android are returned as Cursor objects. Rather than extracting and returning a copy of the result values, Cursors act as pointers to a subset of the underlying data. Cursors are a managed way of controlling your position (row) in the result set of a database query.

The Cursor class includes several functions to navigate query results including, but not limited to, the following:

- ☐ moveToFirst Moves the cursor to the first row in the query result.
- ☐ moveToNext Moves the cursor to the next row.
- ☐ moveToPrevious Moves the cursor to the previous row.
- ☐ getCount Returns the number of rows in the result set.
- ☐ getColumnIndexOrThrow Returns an index for the column with the specified name (throwing an exception if no column exists with that name).
- ☐ getColumnName Returns the name of the specified column index.
- ☐ getColumnNames Returns a String array of all the column names in the current cursor.
- ☐ moveToPosition Moves the cursor to the specified row.
- ☐ getPosition Returns the current cursor position.

Android provides a mechanism to manage Cursor resources within your Activities. The startManagingCursor method integrates the Cursor's lifetime into the parent Activity's lifetime management. When you've finished with the Cursor, call stopManagingCursor to do just that. Later in this chapter, you'll learn how to query a database and how to extract specific row/column values from the resulting Cursor objects.