Managing Android App Data with SQLite

UNDERSTANDING SQLITE AND SQLITE DATABASE CREATION

Jim Wilson
MOBILE SOLUTIONS DEVELOPER & ARCHITECT

@hedgehogjim blog.jwhh.com

What to Expect from This Module

- SQLite Overview
- Designing Our App Database
- Database Contract Class
- Describing SQLite Table Columns
- Making Our Database Framework Friendly
- Database Creation and Access
- Verifying Database Structure and Content

SQLite

Relational database

- Provides structured data storage

Lightweight

- Runs in-process
- Optimized for common usage scenarios

Portable

- Database stored as a file
- Broad operating system support

Database Interaction

Structured Query Language (SQL)

- Domain specific language
- English-like in nature

SQL Challenges

- Parsing is resource intensive
- Potential for SQL based attacks

Database Interaction

Operation specific methods

- Preferable to SQL commands
- Help protect against SQL based attacks

Apps use combination of SQL and methods

- Most data tasks can be done with methods
- Database management tends to need SQL
- Use methods whenever available

Relational Structure

Table

- A set of related entities
- Composed of columns and rows
- Generally 1 or more tables in a database

Relational Structure

Column

- An attribute of an entity
- Sometimes referred to as a field
- Generally 1 or more columns in a table

Row

- An entity instance
- Sometimes referred to as record
- 0 or more rows in a table

Our Database

note_info

note_title	note_text	course_id
Dynamic intent resolution	Wow, intents allow components...	android_intents
Anonymous classes	Anonymous classes simplify...	java_lang
Delegating intents	PendingIntents are powerful...	android_intents

course_info

course_id	course_title
android_intents	Android Programming with Intents
android_async	Android Async Programming and Services
java_lang	Java Fundamentals: The Java Language
java_core	Java Fundamentals: The Core Platform

Database Contract Class

Holds information about database

- Organization and structure (i.e. schema)
- Primarily constants describing DB
- Class is normally non-creatable

Database Contract Class

Nested class for each table

- Provides constant for table name
- Provides constants for column names
- May contain key SQL statements

Table creation SQL

- Uses “CREATE TABLE” statement
- Names the table
- Describes the table columns

Table Columns

Describing table columns

- Name
- Storage class
- Constraints

Storage Class

Storage class is optional

- SQLite does not have rigid typing
- Any column can store any type
- Storage class influences storage affinity

	“123”	123	123.0	
BLOB	“123”	123	123.0	
TEXT	“123”	“123”	“123.0”	
INTEGER	123	123	123	
REAL	123.0	123.0	123.0	
NUMERIC	123	123	123	123.1 → 123.1

Storage Class

Storage class is optional

- SQLite does not have rigid typing
- Any column can store any type
- Storage class influences storage affinity

	“123”	123	123.0	“abc”
BLOB	“123”	123	123.0	“abc”
TEXT	“123”	“123”	“123.0”	“abc”
INTEGER	123	123	123	“abc”
REAL	123.0	123.0	123.0	“abc”
NUMERIC	123	123	123	“abc”

Constraints

Constraints restrict allowable content

- Associate with columns as needed
- Automatically enforced by database

Common Constraints

NOT NULL

- Column cannot contain null

UNIQUE

- No two rows can have the same value for the specified column

Primary Key

PRIMARY KEY

- Provides unambiguous row identity
- A table has no more than one

Android Framework Friendly Tables

SQLite tags each row with a unique integer

- Automatically assigns the value
- Provides fast and efficient row access

Create tables that work well with framework

- Give table an integer primary key
- Primary key is associated with row tag
- Use BaseColumn._ID as column name

Our Database

note_info

_id	note_title	note_text	course_id
1	Dynamic intent resolution	Wow, intents allow components...	android_intents
2	Anonymous classes	Anonymous classes simplify...	java_lang
3	Delegating intents	PendingIntents are powerful...	android_intents

course_info

_id	course_id	course_title
1	android_intents	Android Programming with Intents
2	android_async	Android Async Programming and Services
3	java_lang	Java Fundamentals: The Java Language
4	java_core	Java Fundamentals: The Core Platform

Database Access and Creation

Database access has implications

Database creation

- Must check whether database exists
- DB must be created if doesn't exist

Database versioning

- App DB structure may change over time
- Must track current DB version
- App must indicate expected DB version
- DB must be updated when necessary

Database Open Helper

SQLiteOpenHelper

- Provides DB access methods
- Manages DB creation and versioning

Extend to meet your app's needs

- Provide expected database version
- Provide database filename
- Override appropriate methods

Database Open Helper

Override onCreate

- Called if database doesn't exist
- Execute SQL to create tables
- Add initial data if necessary

Override onUpgrade

- Called when newer DB version expected
- Execute SQL to upgrade tables
- Preserve existing data if necessary

Database Access and Creation

Using your database open helper class

- Instantiate in activity's onCreate
- Maintain reference as member field
- Close in activity's onDestroy

Database Access and Creation

Helper class provides DB access methods

- getReadableDatabase
- getWritableDatabase
- Trigger creation/version checks as needed
- Return SQLiteDatabase reference

SQLiteDatabase

- Provides database interaction methods
- Operation specific methods
- Execute SQL statements with execSQL

Summary

SQLite

- Relational database
- Lightweight and portable

Relational structure

- Table
  - A set of related entities
  - Composed of columns and rows
- Column
  - An attribute of an entity
- Row
  - An entity instance

Summary

Database contract class

- Holds information about database

Nested class for each table

- Provides constant for table name
- Provides constants for column names
- May contain key SQL statements

Summary

Table columns

- Name
- Storage class
- Constraints

Framework friendly tables

- Give table integer primary key
- Use BaseColumn._ID as column name

Summary

SQLiteOpenHelper

- Provides DB access methods
- Manages DB creation and versioning

Extend to meet your app's needs

- Provide database name and version
- Override onCreate
- Override onUpgrade