Reading Data from SQLite

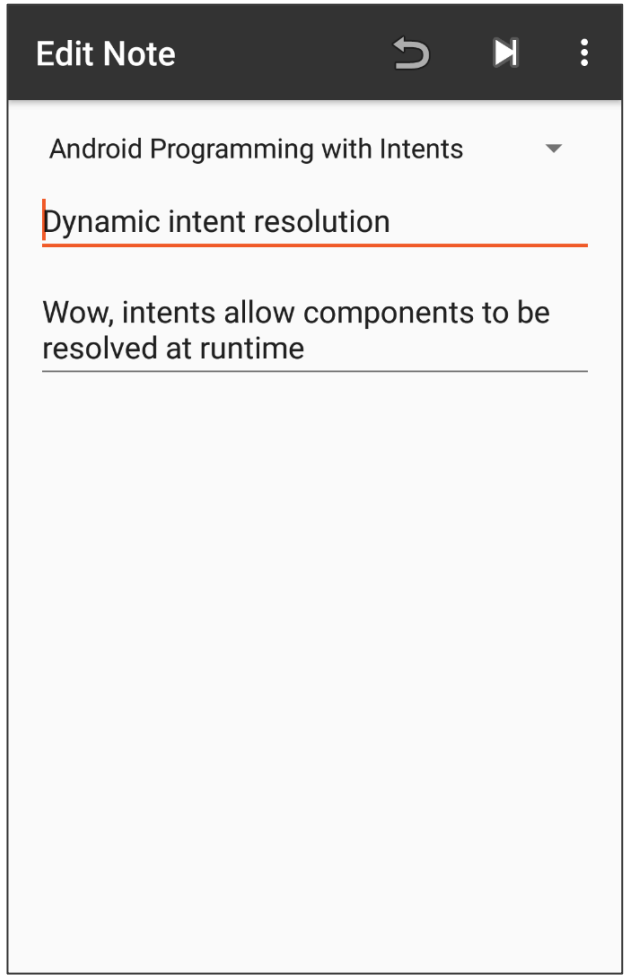
Jim Wilson
MOBILE SOLUTIONS DEVELOPER & ARCHITECT

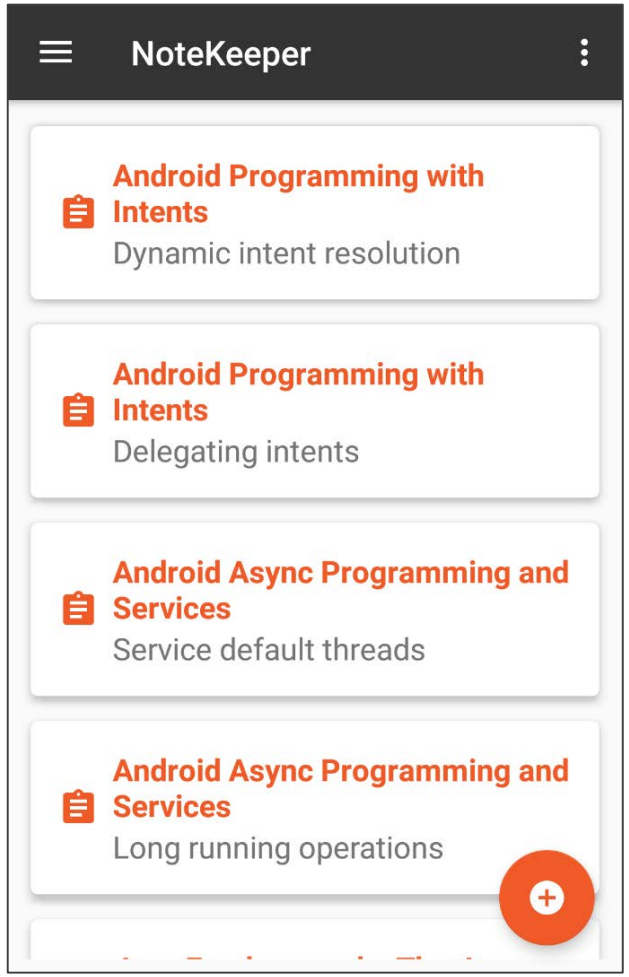
@hedgehogjim blog.jwhh.com

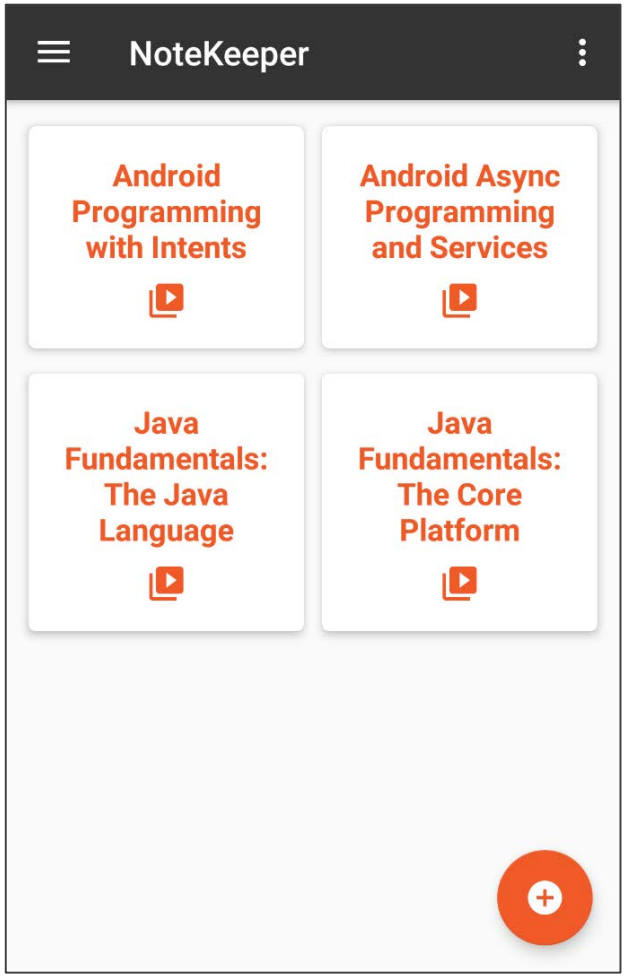
What to Expect from This Module

- Our App Plan
- Database Queries
- Moving Through Results with a Cursor
- Accessing Result Values
- Ordering Results

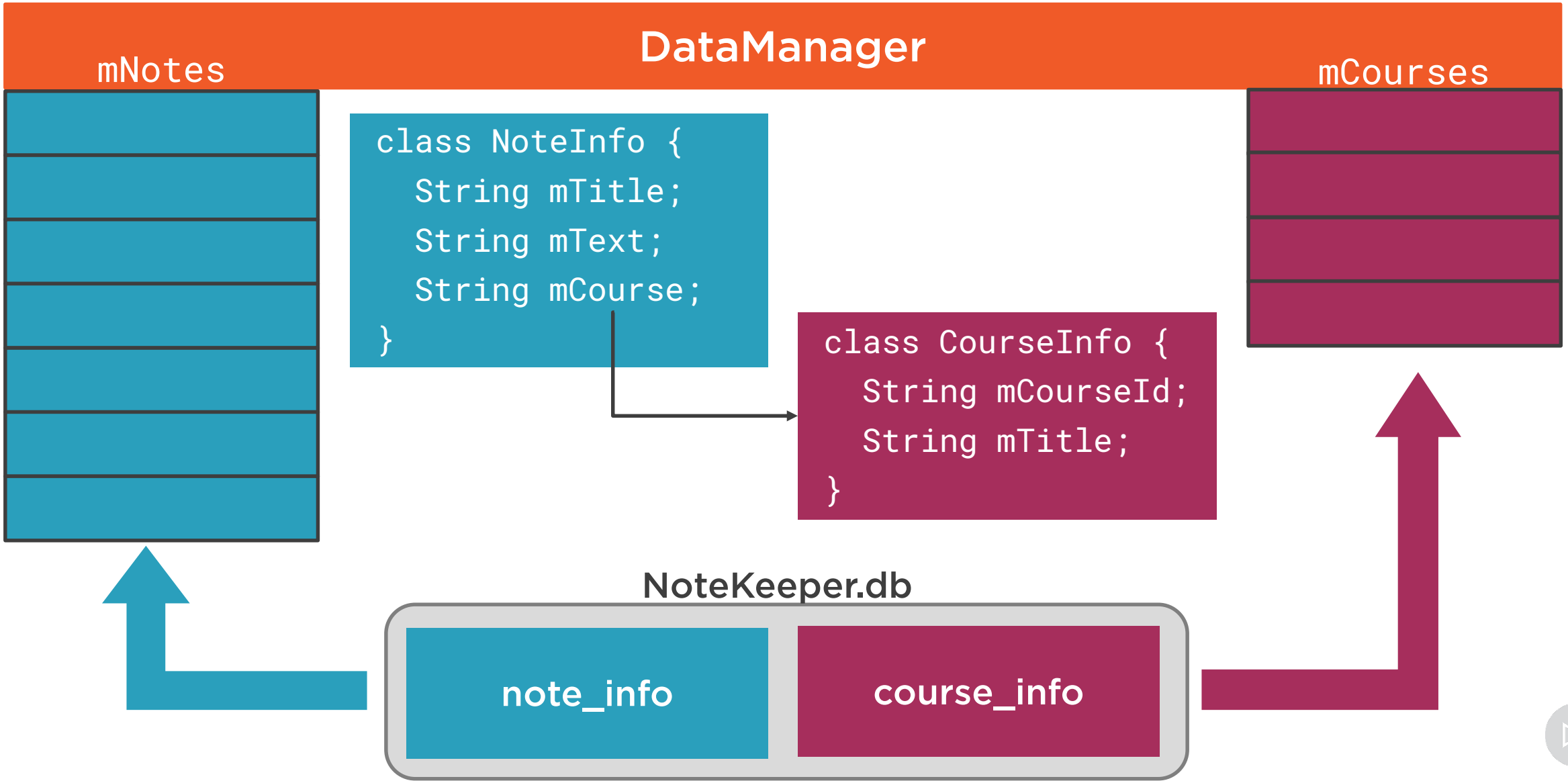
Our App Plan


Edit Note
Android Programming with Intents
Dynamic intent resolution
Wow, intents allow components to be resolved at runtime


NoteKeeper
Android Programming with Intents
Dynamic intent resolution
Android Programming with Intents
Delegating intents
Android Async Programming and Services
Service default threads
Android Async Programming and Services
Long running operations


NoteKeeper
Android Programming with Intents
Android Async Programming and Services
Java Fundamentals: The Java Language
Java Fundamentals: The Core Platform

Our App Plan


DataManager
mNotes
mCourses
class NoteInfo {
String mTitle;
String mText;
String mCourse;
}
class CourseInfo {
String mCourseId;
String mTitle;
}
NoteKeeper.db
note_info
course_info

Accessing Data

Data access has distinct phases

- Request database connection
- Issue query
- Move through results

Request Database Connection

Connection provided by SQLiteOpenHelper

- Use your open helper implementation
- Call getReadableDatabase
- Returns SQLiteDatabase reference

Issue Query

SQLiteDatabase query method

- Parameters describe query details
- Unneeded parameters can be null

Fundamental query parameters

- Table name
- Columns to be returned

Move Through Results

Cursor

- Provides access to query result
- Query result can have 0, 1, or more rows
- Result interaction occurs row-by-row

Move Through Results

Cursor maintains a current position

- Initially positioned before first row
- Must explicitly move to desired row

Cursor.moveToNext

- Positions to next row in result
- Returns false when moves past end

Moving Through Results

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

query ( );

course_id	note_title
android_intents	Dynamic intent resolution
java_lang	Anonymous classes
android_intents	Delegating intents

Move Through Results

Cursor.moveToPrevious

- Positions to previous row in result
- Returns false when moves past beginning

Moving Through Results

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

query ("note_info", {"course_id", "note_title"}, . . .);

course_id	note_title
android_intents	Dynamic intent resolution
java_lang	Anonymous classes
android_intents	Delegating intents

Move Through Results

Cursor.moveToFirst

- Positions to first row in result

Cursor.moveToLast

- Positions to last row in result

Cursor.moveToPosition

- Positions to specified row in result
- Uses zero-based position index

Moving Through Results

Can access current row column values

- Use typed Cursor.getXXX methods
- Use zero-based column position

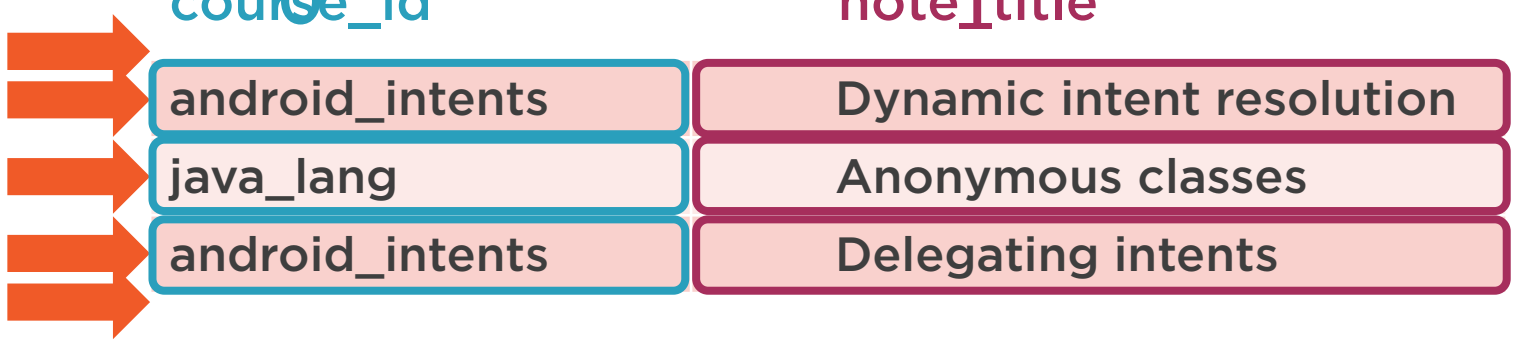
Moving Through Results

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

0 1

query (“note_info”, {“course_id”, “note_title”}, . . .);



Moving Through Results

Avoid hard-coding column positions

- Makes code fragile

Request position of column name

- Cursor.getColumnIndex
- Accepts column name
- Returns column position in result

Moving Through Results

Close Cursor when done with it

- Cursor.close
- May leak system resources if not closed

Ordering Results

Can specify row order

- Order by columns in result

Passed as a string to query method

- Pass column name

Ordering Results

Can sort by multiple columns

- List columns comma separated
- First column is primary sort
- Second column sorted within primary
- Third sorted within second and so on

Ordering Results

Can sort in descending order if desired

- Follow column name by “ DESC”
- Applies only to that column

Summary

SQLiteDatabase query method

- Requests data from database

Fundamental query parameters

- Table name
- Columns to be returned

Summary

Cursor

- Provides access to query result
- Result interaction occurs row-by-row

Cursor positioning

- Must explicitly move to desired row
- Initially positioned before the first row

Summary

Column value access

- Use typed Cursor.getXXX methods
- Accessed by column position
- Use getColumnIndex to find position

Close cursor when done

- Use Cursor.close

Summary

Can specify row order

- Pass column name
- Comma separate for multiple columns
- Use “ DESC” for descending order