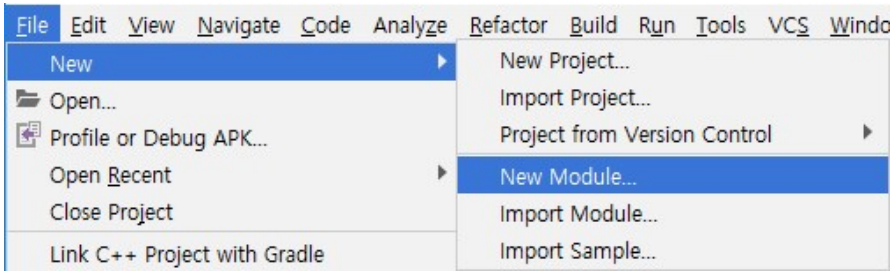
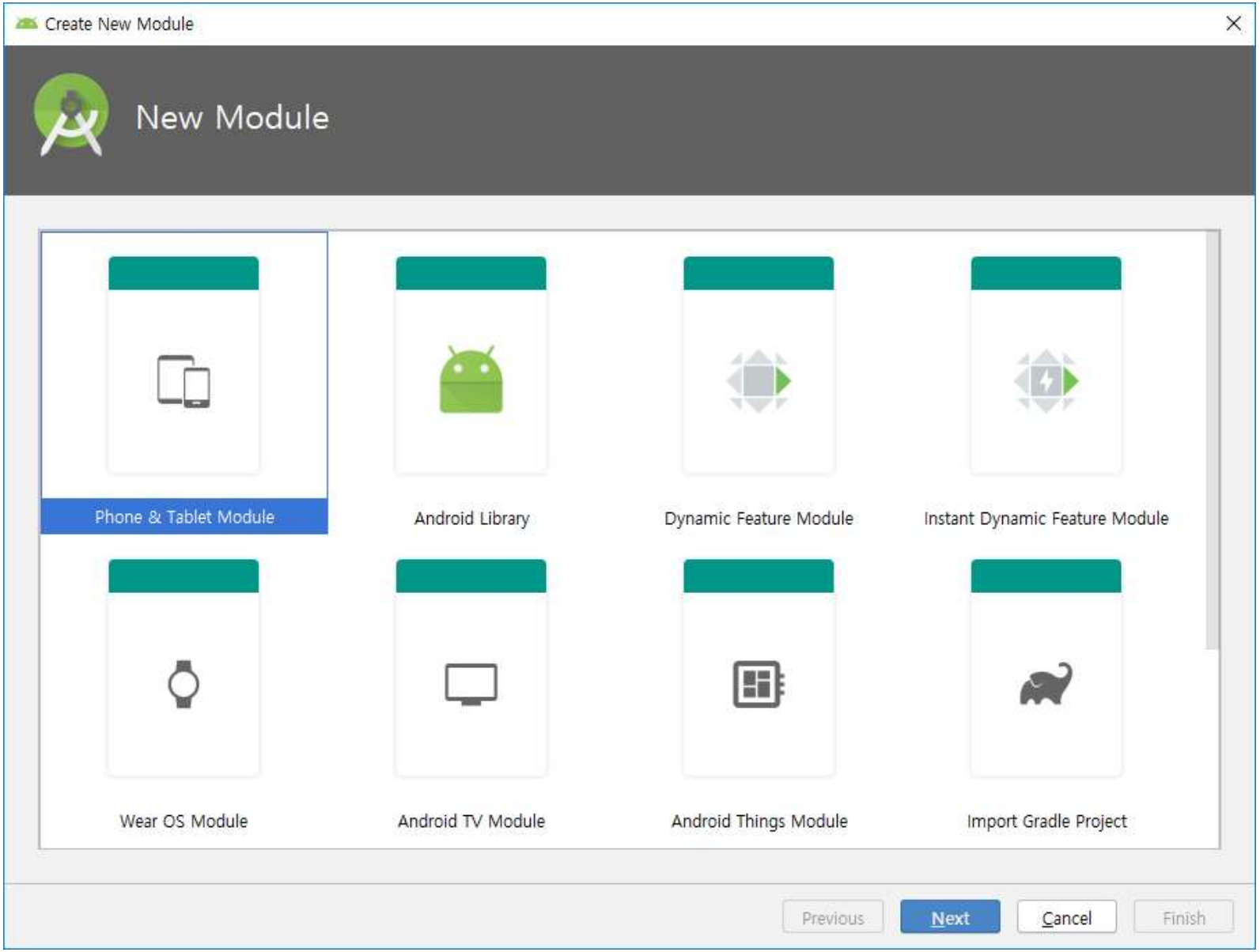
SQLite 실습

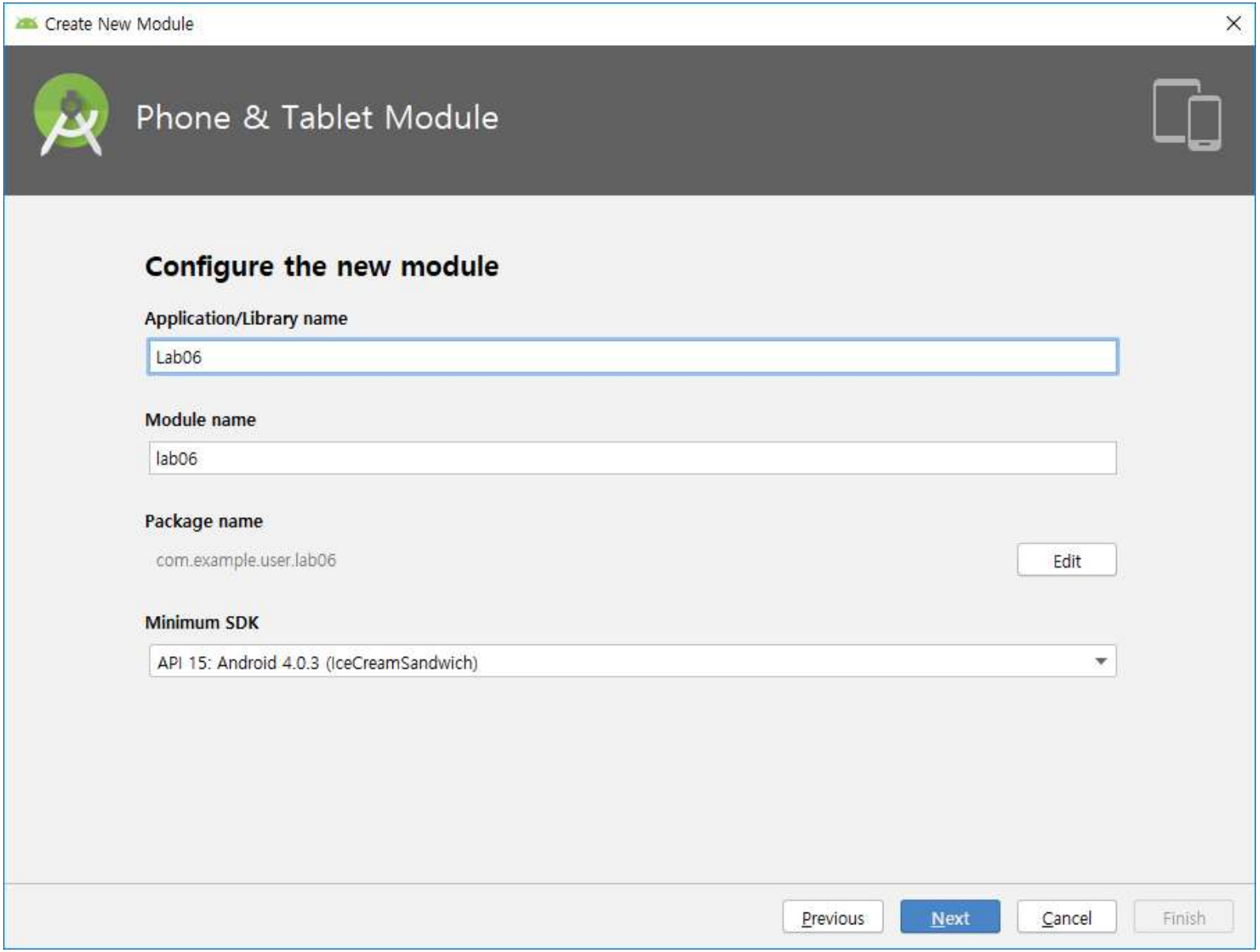
DBMS

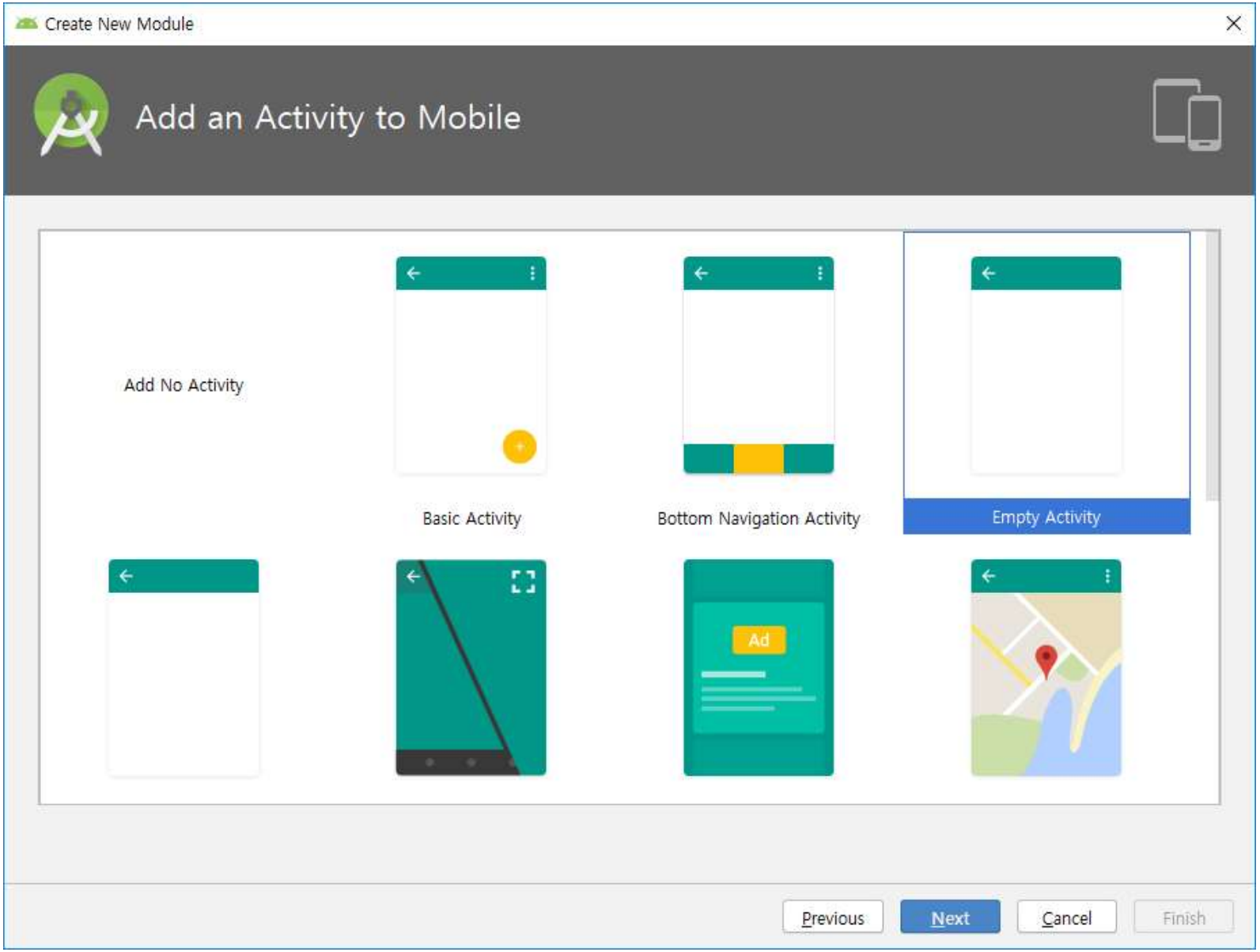
DB 관리시스템

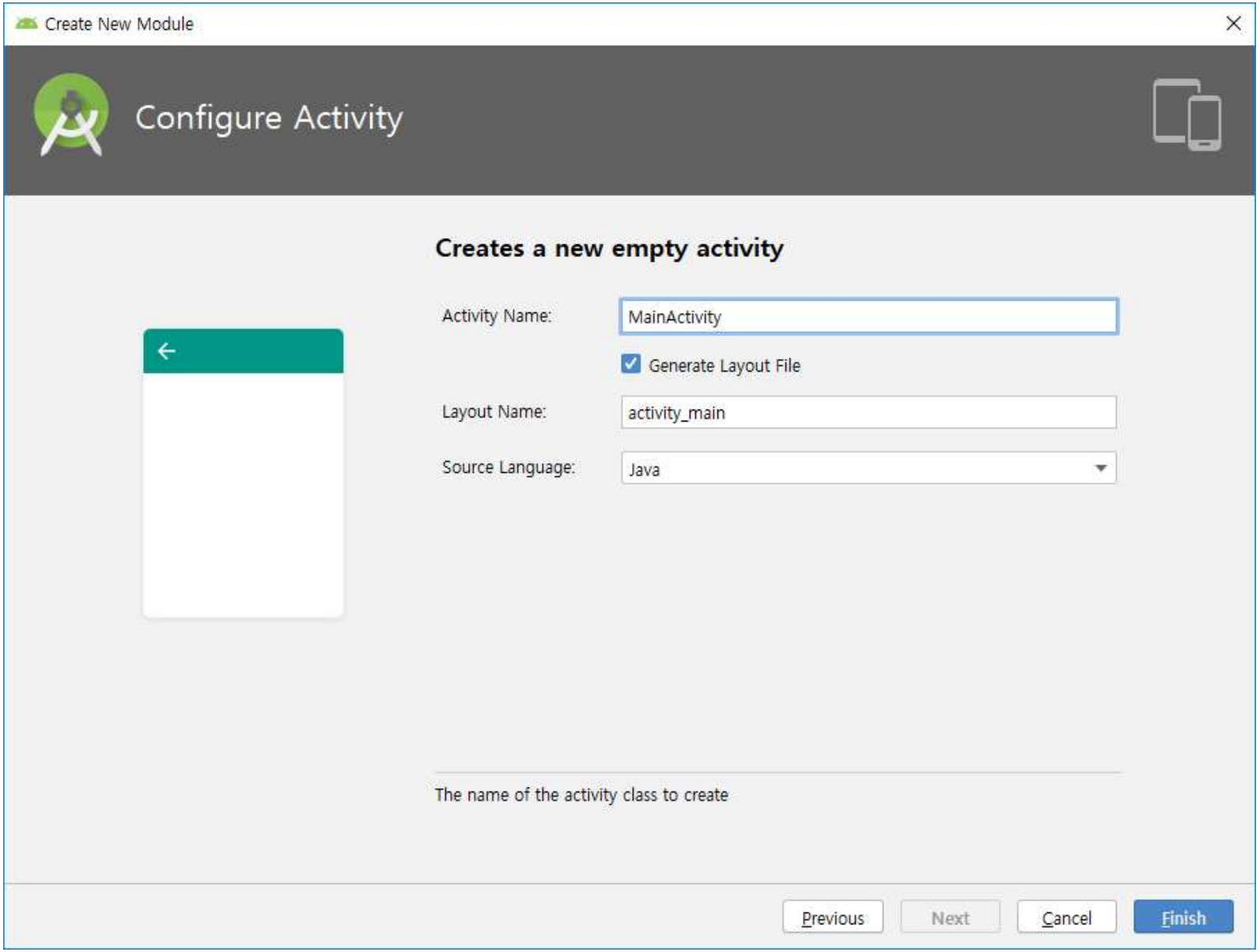
Step 1 _ 모듈 및 액티비티 생성

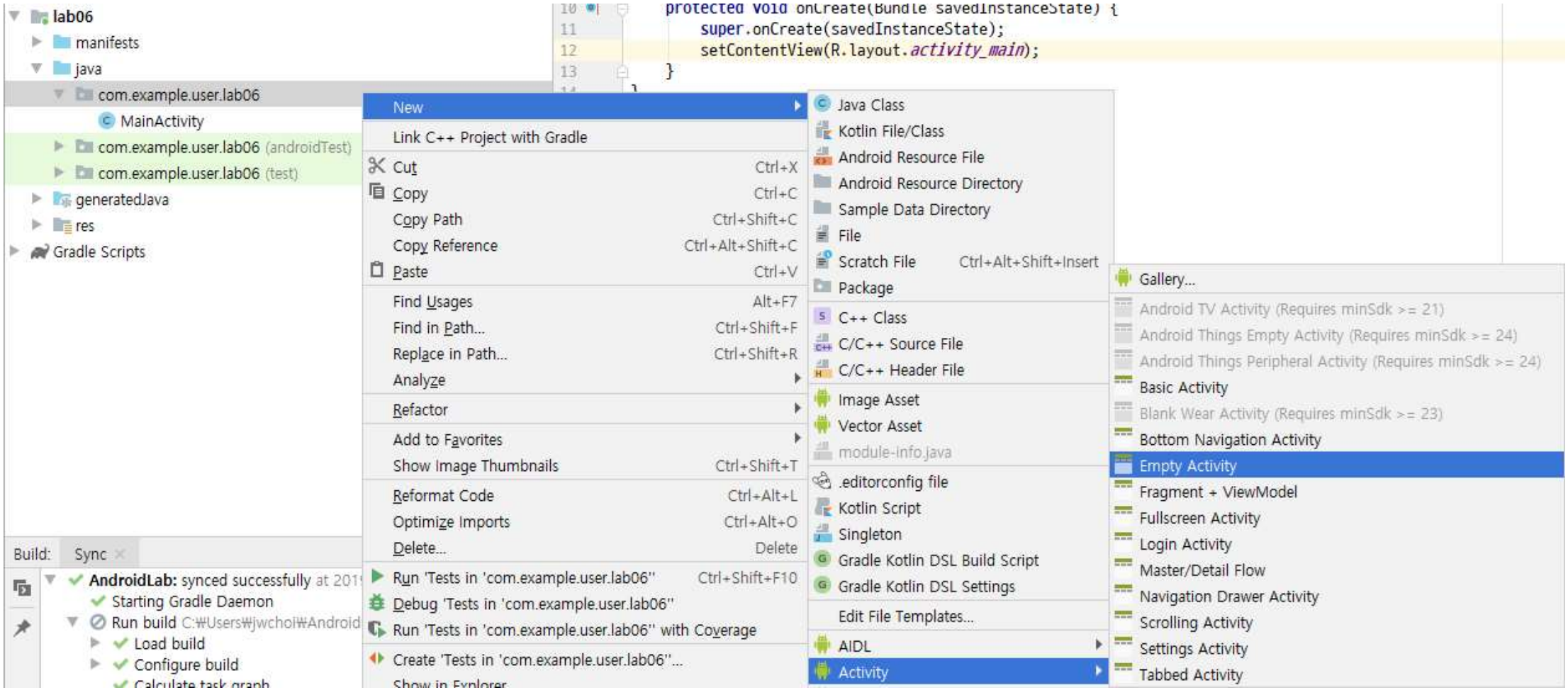


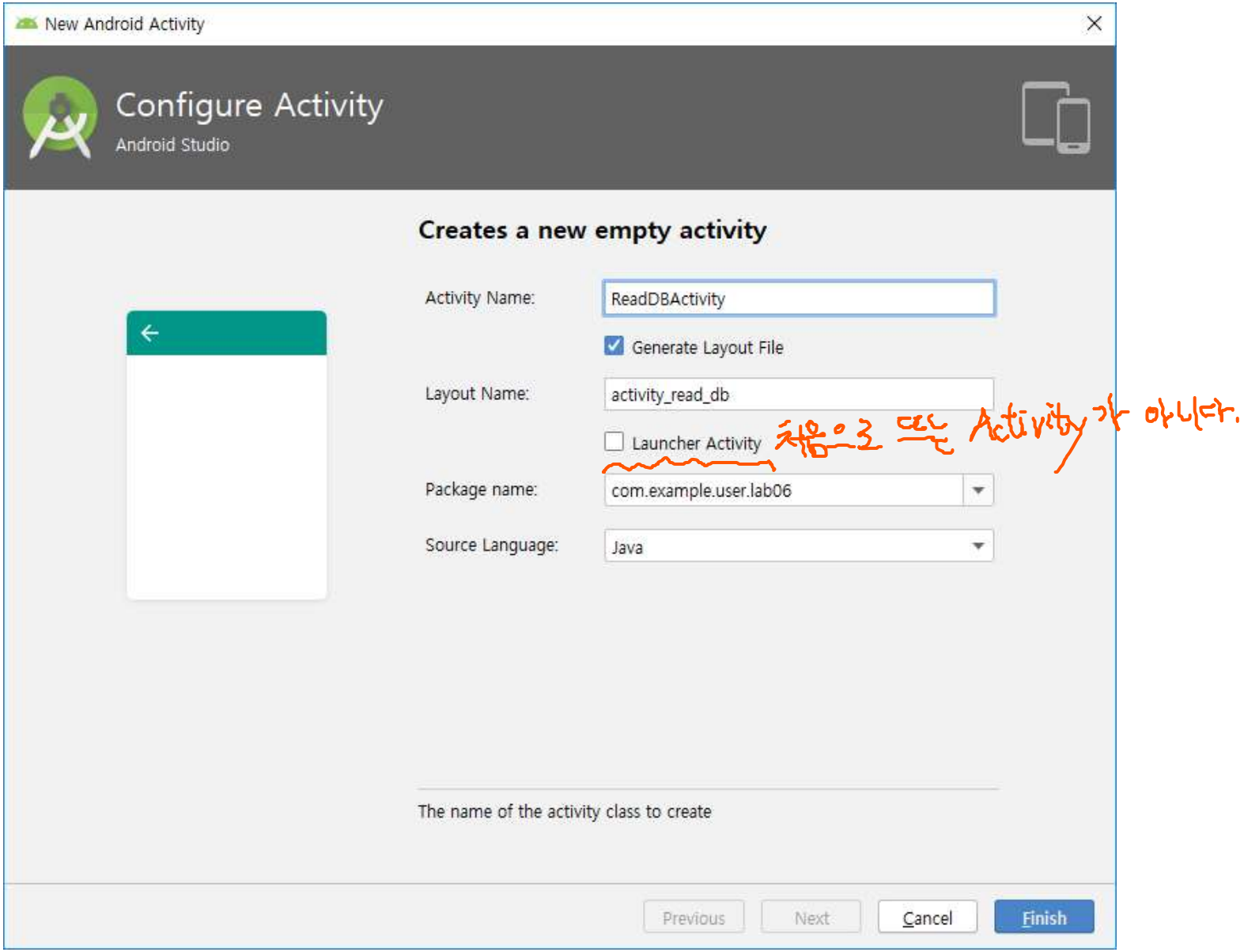

Create New Module
New Module
Phone & Tablet Module
Android Library
Dynamic Feature Module
Instant Dynamic Feature Module
Wear OS Module
Android TV Module
Android Things Module
Import Gradle Project
Previous
Next
Cancel
Finish


Create New Module
Phone & Tablet Module
Configure the new module
Application/Library name
Lab06
Module name
lab06
Package name
com.example.user.lab06
Edit
Minimum SDK
API 15: Android 4.0.3 (IceCreamSandwich)
Previous
Next
Cancel
Finish

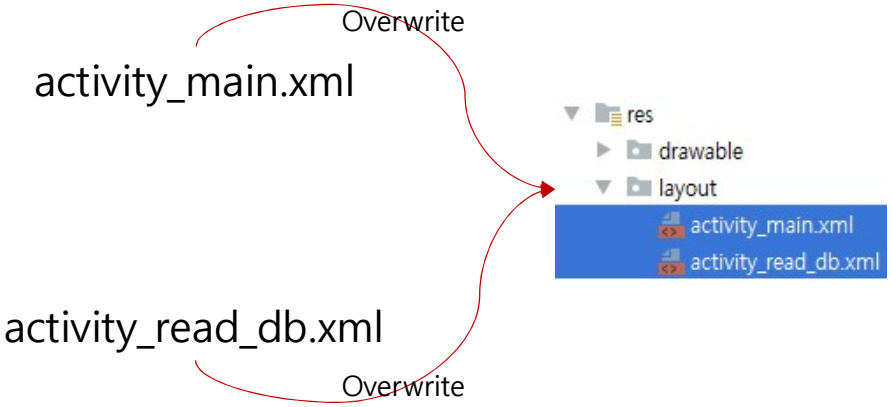

Create New Module
Add an Activity to Mobile
Add No Activity
Basic Activity
Bottom Navigation Activity
Empty Activity
Ad
Previous
Next
Cancel
Finish


Create New Module
Configure Activity
Creates a new empty activity
Activity Name:
MainActivity
Generate Layout File
Layout Name:
activity_main
Source Language:
Java
The name of the activity class to create
Previous
Next
Cancel
Finish

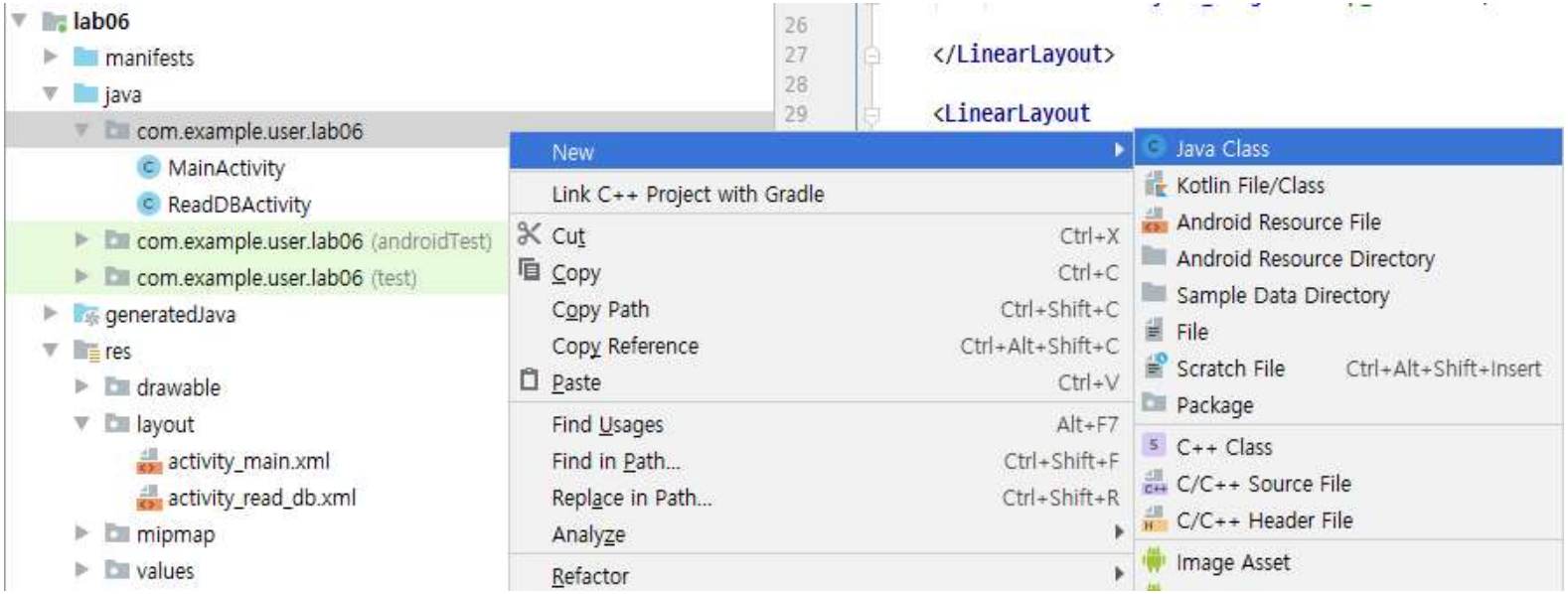

lab06
manifests
java
com.example.user.lab06
MainActivity
com.example.user.lab06 (androidTest)
com.example.user.lab06 (test)
generatedJava
res
Gradle Scripts
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
}
New
Link C++ Project with Gradle
Cut Ctrl+X
Copy Ctrl+C
Copy Path Ctrl+Shift+C
Copy Reference Ctrl+Alt+Shift+C
Paste Ctrl+V
Find Usages Alt+F7
Find in Path... Ctrl+Shift+F
Replace in Path... Ctrl+Shift+R
Analyze
Refactor
Add to Favorites
Show Image Thumbnails Ctrl+Shift+T
Reformat Code Ctrl+Alt+L
Optimize Imports Ctrl+Alt+O
Delete... Delete
Run 'Tests in 'com.example.user.lab06'' Ctrl+Shift+F10
Debug 'Tests in 'com.example.user.lab06''
Run 'Tests in 'com.example.user.lab06'' with Coverage
Create 'Tests in 'com.example.user.lab06''...
Show in Explorer
Java Class
Kotlin File/Class
Android Resource File
Android Resource Directory
Sample Data Directory
File
Scratch File Ctrl+Alt+Shift+Insert
Package
C++ Class
C/C++ Source File
C/C++ Header File
Image Asset
Vector Asset
module-info.java
.editorconfig file
Kotlin Script
Singleton
Gradle Kotlin DSL Build Script
Gradle Kotlin DSL Settings
Edit File Templates...
AIDL
Activity
Gallery...
Android TV Activity (Requires minSdk >= 21)
Android Things Empty Activity (Requires minSdk >= 24)
Android Things Peripheral Activity (Requires minSdk >= 24)
Basic Activity
Blank Wear Activity (Requires minSdk >= 23)
Bottom Navigation Activity
Empty Activity
Fragment + ViewModel
Fullscreen Activity
Login Activity
Master/Detail Flow
Navigation Drawer Activity
Scrolling Activity
Settings Activity
Tabbed Activity
Build:
Sync
AndroidLab: synced successfully at 201
Starting Gradle Daemon
Run build C:₩Users₩jwchoi₩Android
Load build
Configure build
Calculate task graph

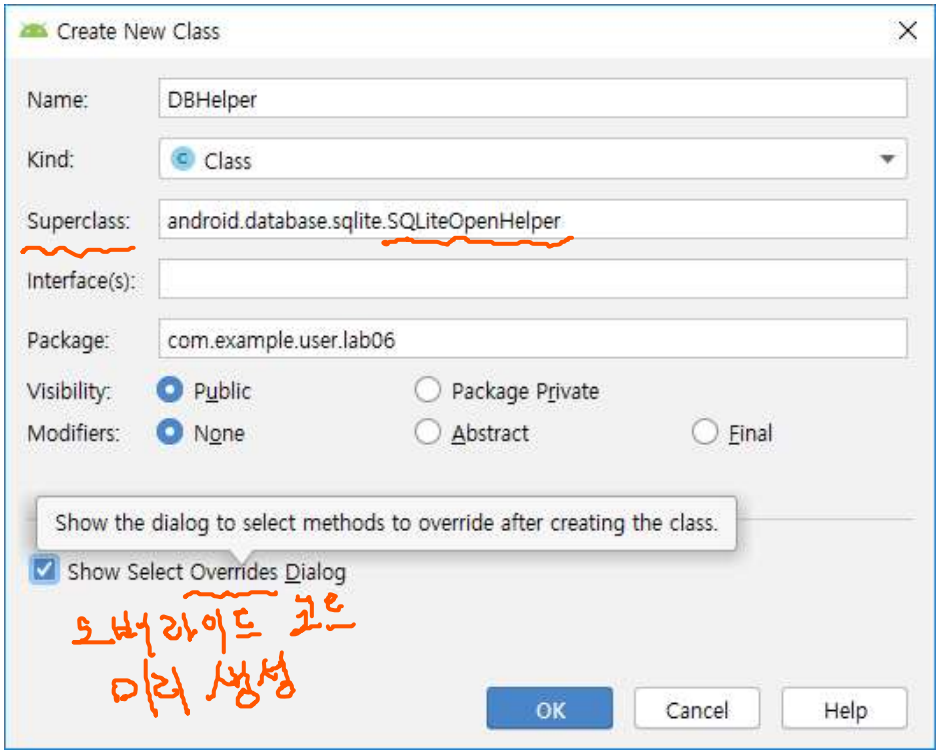

New Android Activity
Configure Activity
Android Studio
Creates a new empty activity
Activity Name:
ReadDBActivity
Generate Layout File
Layout Name:
activity_read_db
Launcher Activity
Package name:
com.example.user.lab06
Source Language:
Java
The name of the activity class to create
Previous
Next
Cancel
Finish

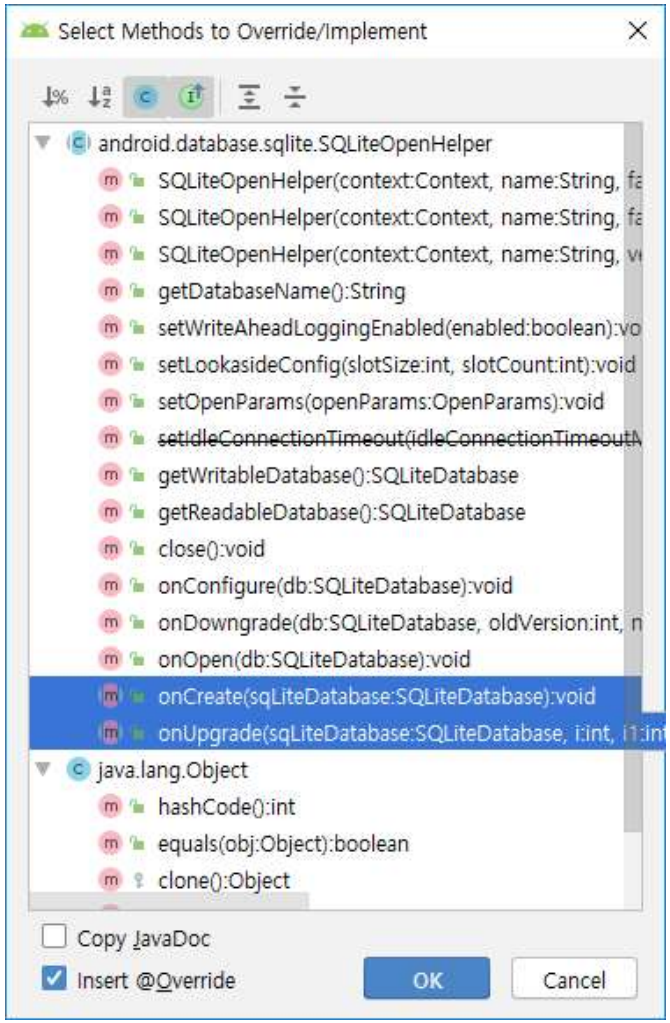
Step 2 _ 레이아웃 XML 파일 복사



Step 3 _ DBHelper 클래스 작성




Create New Class
Name:
DBHelper
Kind:
Class
Superclass:
android.database.sqlite.SQLiteOpenHelper
Interface(s):
Package:
com.example.user.lab06
Visibility:
Public
Package Private
Modifiers:
None
Abstract
Final
Show the dialog to select methods to override after creating the class.
Show Select Overrides Dialog
오버라이드 할 메서 생성
OK
Cancel
Help


Select Methods to Override/Implement
android.database.sqlite.SQLiteOpenHelper
SQLiteOpenHelper(context:Context, name:String, fa
SQLiteOpenHelper(context:Context, name:String, fa
SQLiteOpenHelper(context:Context, name:String, v
getDatabaseName():String
setWriteAheadLoggingEnabled(enabled:boolean):vo
setLookasideConfig(slotSize:int, slotCount:int):void
setOpenParams(openParams:OpenParams):void
setIdleConnectionTimeout(idleConnectionTimeoutM
getWritableDatabase():SQLiteDatabase
getReadableDatabase():SQLiteDatabase
close():void
onConfigure(db:SQLiteDatabase):void
onDowngrade(db:SQLiteDatabase, oldVersion:int, n
onOpen(db:SQLiteDatabase):void
onCreate(sqLiteDatabase:SQLiteDatabase):void
onUpgrade(sqLiteDatabase:SQLiteDatabase, i:int, i1:int
java.lang.Object
hashCode():int
equals(obj:Object):boolean
clone():Object
Copy JavaDoc
Insert @Override
OK
Cancel


DBHelper.java
package com.example.user.lab06;
import ...
public class DBHelper extends SQLiteOpenHelper {
@Override
public
There is no default constructor available in 'android.database.sqlite.SQLiteOpenHelper'
}
@Override
public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
}
}
생성자 필요

```java
public class DBHelper extends SQLiteOpenHelper {

    public static final int DATABASE_VERSION = 1;

    public DBHelper(Context context) {
        super(context, "memodb", null, DATABASE_VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        String memoSQL = "create table tb_memo (" +
                "_id integer primary key autoincrement, " +
                "title, " +
                "content)";

        db.execSQL(memoSQL);
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        if (newVersion == DATABASE_VERSION) {
            db.execSQL("drop table tb_memo");
            onCreate(db);
        }
    }
}
```

생성자

db 이름

- 앱이 설치된 후 SQLiteOpenHelper가 최초로 이용되는 순간 한 번 호출
- 대부분 테이블을 생성하는 코드를 작성

oldVersion의 DB 파일을 연 SQLiteDatabase 객체

- SQLiteOpenHelper 클래스의 생성자에 전달되는 데이터베이스 버전 정보가 변경될 때마다 호출
- 테이블의 스키마 부분을 변경하기 위한 용도로 사용

SQLiteOpenHelper

Added in API level 1

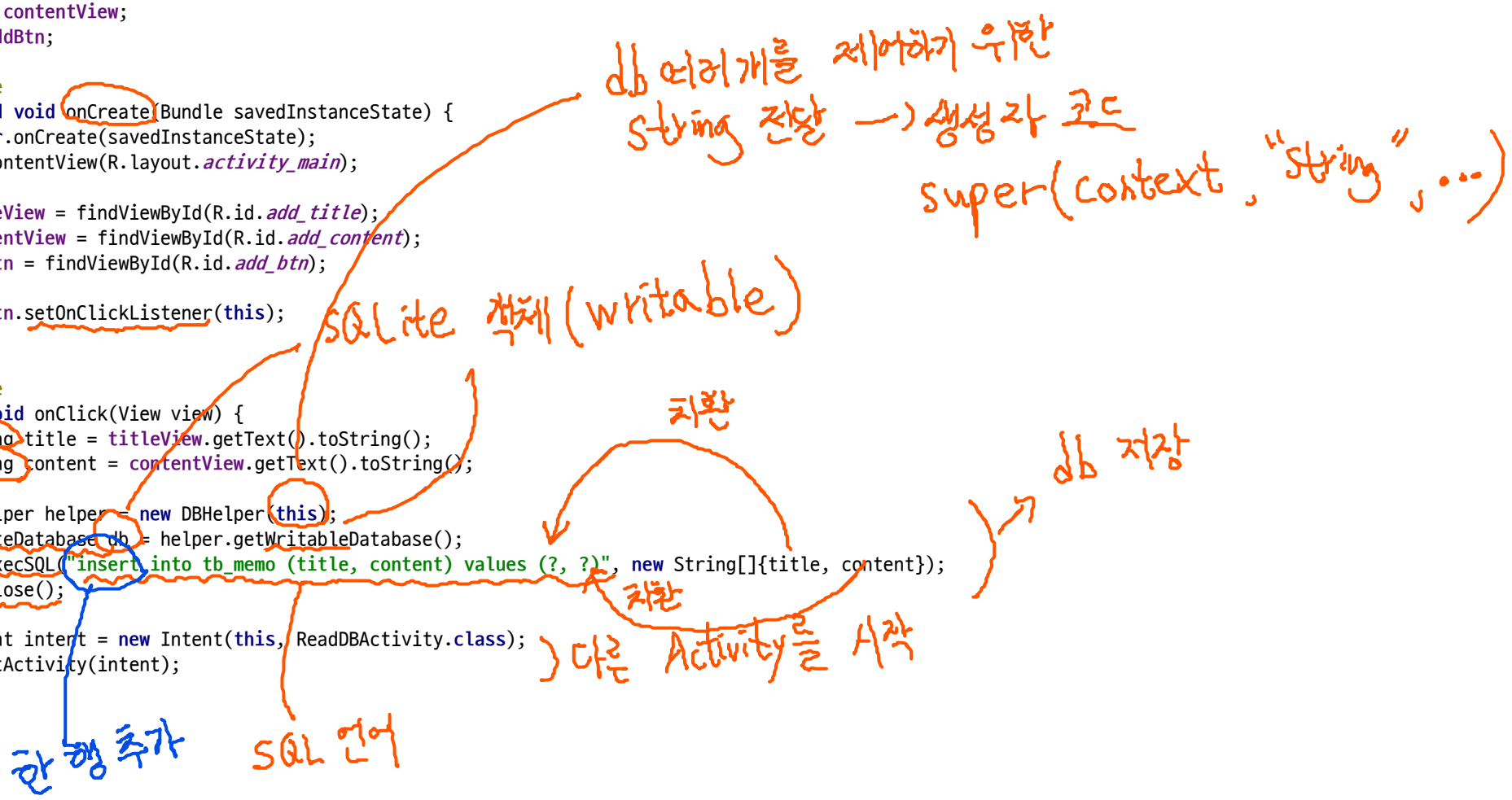
```
public SQLiteOpenHelper(Context context,
                String name,
                SQLiteDatabase.CursorFactory factory,
                int version)
```

Create a helper object to create, open, and/or manage a database. This method always returns very quickly. The database is not actually created or opened until one of `getWritableDatabase()` or `getReadableDatabase()` is called.

Parameters	
`context`	`Context`: to use for locating paths to the the database This value may be `null`.
`name`	`String`: of the database file, or null for an in-memory database This value may be `null`.
`factory`	`SQLiteDatabase.CursorFactory`: to use for creating cursor objects, or null for the default This value may be `null`.
`version`	`int`: number of the database (starting at 1); if the database is older, `onUpgrade(SQLiteDatabase, int, int)` will be used to upgrade the database; if the database is newer, `onDowngrade(SQLiteDatabase, int, int)` will be used to downgrade the database

Step 4 _ MainActivity 작성

```
public class MainActivity extends AppCompatActivity implements View.OnClickListener {

    EditText titleView;
    EditText contentView;
    Button addBtn;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        titleView = findViewById(R.id.add_title);
        contentView = findViewById(R.id.add_content);
        addBtn = findViewById(R.id.add_btn);

        addBtn.setOnClickListener(this);
    }

    @Override
    public void onClick(View view) {
        String title = titleView.getText().toString();
        String content = contentView.getText().toString();

        DBHelper helper = new DBHelper(this);
        SQLiteDatabase db = helper.getWritableDatabase();
        db.execSQL("insert into tb_memo (title, content) values (?, ?)", new String[]{title, content});
        db.close();

        Intent intent = new Intent(this, ReadDBActivity.class);
        startActivity(intent);
    }
}
```



getWritableDatabase

Added in API level 1

```
public SQLiteDatabase getWritableDatabase ()
```

Create and/or open a database that will be used for reading and writing. The first time this is called, the database will be opened and `onCreate(SQLiteDatabase)`, `onUpgrade(SQLiteDatabase, int, int)` and/or `onOpen(SQLiteDatabase)` will be called.

Once opened successfully, the database is cached, so you can call this method every time you need to write to the database. (Make sure to call `close()` when you no longer need the database.) Errors such as bad permissions or a full disk may cause this method to fail, but future attempts may succeed if the problem is fixed.

> Database upgrade may take a long time, you should not call this method from the application main thread, including from `ContentProvider.onCreate()`.

Returns	
`SQLiteDatabase`	a read/write database object valid until `close()` is called

Throws	
`SQLiteException`	if the database cannot be opened for writing

getReadableDatabase

Added in API level 1

```
public SQLiteDatabase getReadableDatabase ()
```

Create and/or open a database. This will be the same object returned by `getWritableDatabase()` unless some problem, such as a full disk, requires the database to be opened read-only. In that case, a read-only database object will be returned. If the problem is fixed, a future call to `getWritableDatabase()` may succeed, in which case the read-only database object will be closed and the read/write object will be returned in the future.

> **!** Like **`getWritableDatabase()`**, this method may take a long time to return, so you should not call it from the application main thread, including from **`ContentProvider.onCreate()`**.

Returns	
`SQLiteDatabase`	a database object valid until `getWritableDatabase()` or `close()` is called.

Throws	
`SQLiteException`	if the database cannot be opened

Step 5 _ ReadDBActivity 작성

```
public class ReadDBActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_read_db);

        TextView titleView = findViewById(R.id.read_title);
        TextView contentView = findViewById(R.id.read_content);
        DBHelper helper = new DBHelper(this);
        SQLiteDatabase db = helper.getWritableDatabase();
        Cursor cursor = db.rawQuery("select title, content from tb_memo order by _id desc limit 1", null);
        while (cursor.moveToNext()) {
            titleView.setText(cursor.getString(0));
            contentView.setText(cursor.getString(1));
        }
        db.close();
    }
}
```

getReadableDatabase() 가능 (메모리 부족할까보냐) 제일 위 한 행만

DB에서 data 가져옴 — like 반복자

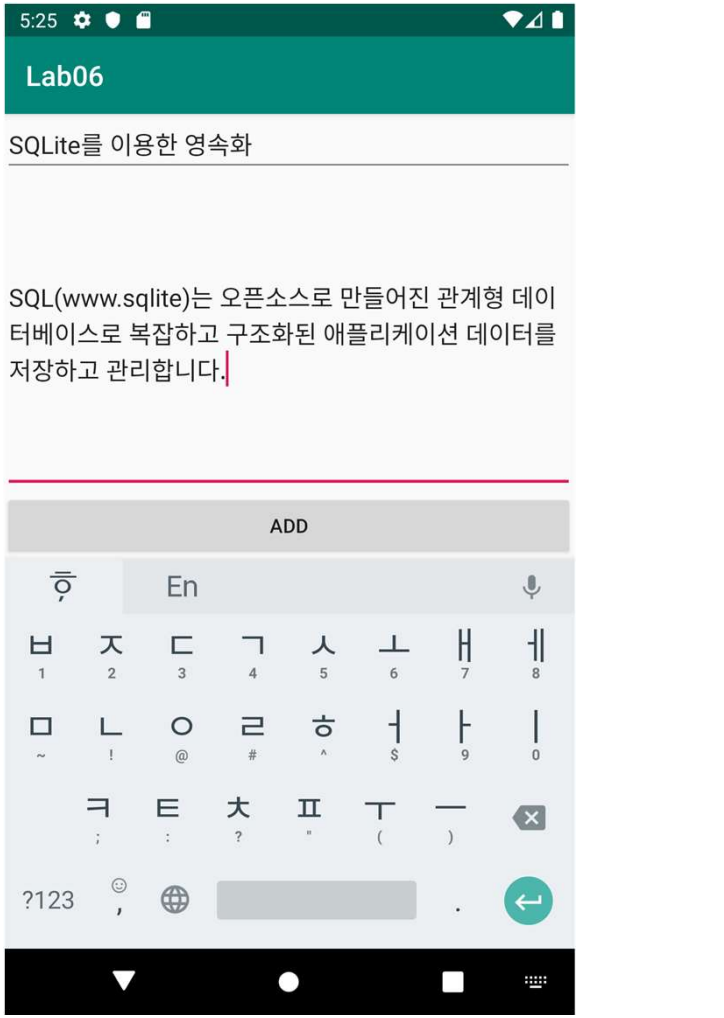
0열 — 1열

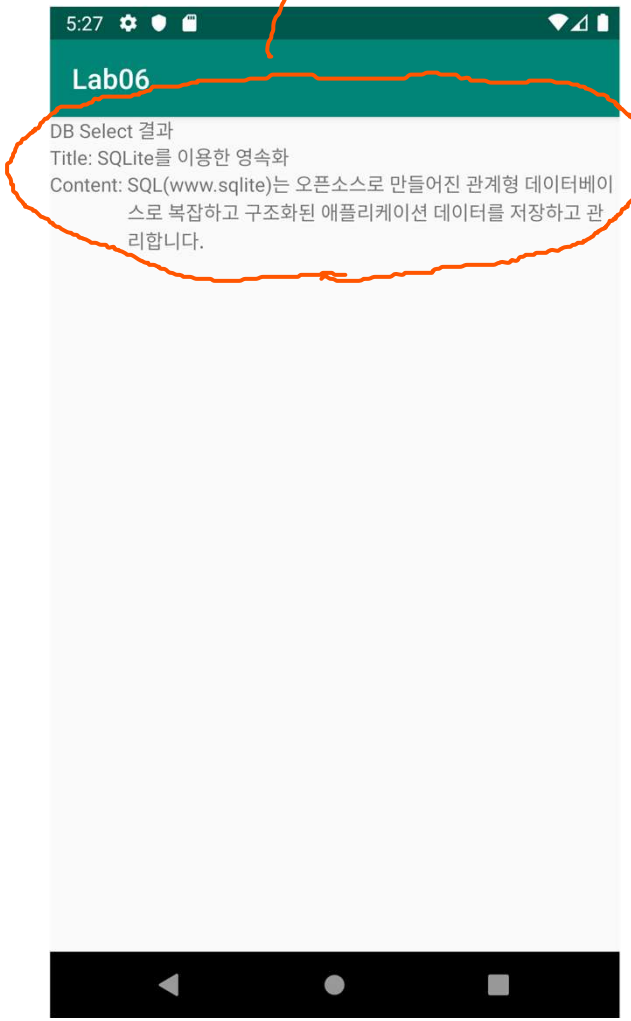
내림차순 정렬

치환 String[] 없다. ('?'가 없다)

selectionArgs	String[]	: You may include ?s in where clause in the query, which will be replaced by the values from selectionArgs. The values will be bound as Strings.

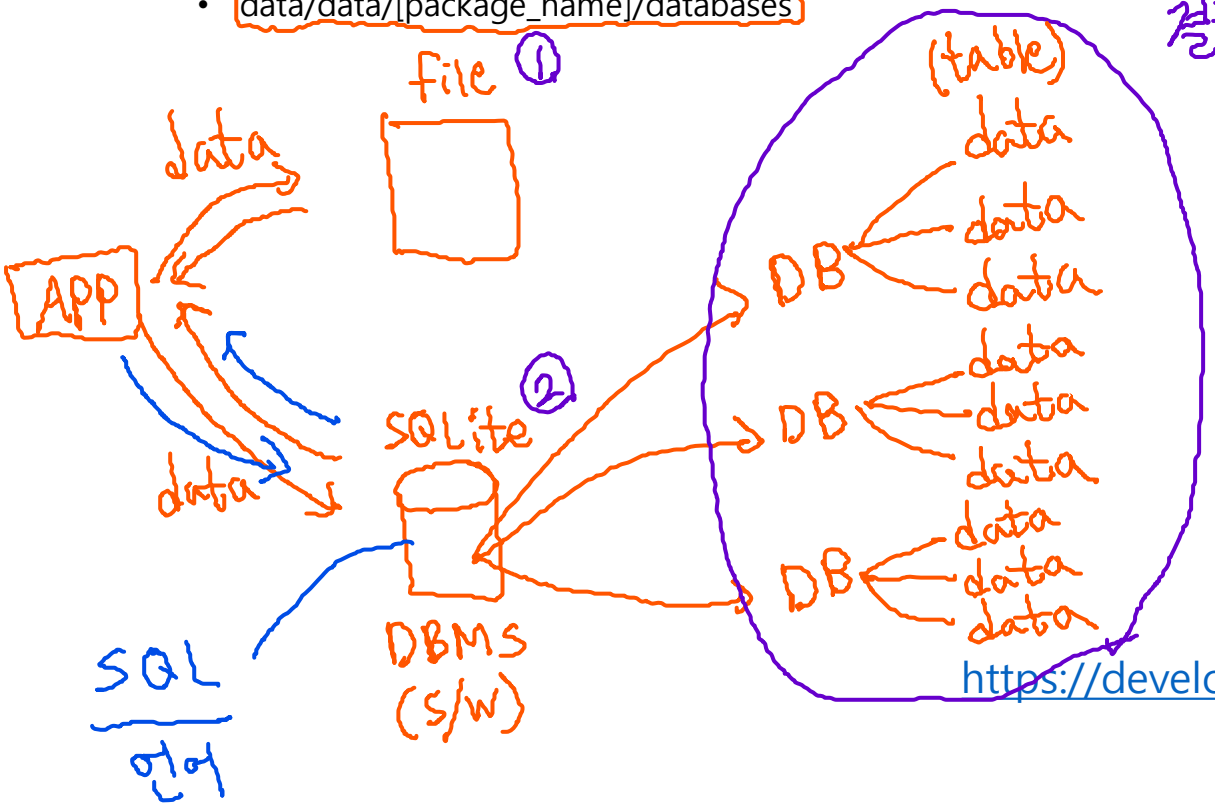
Step 6 _ 실행

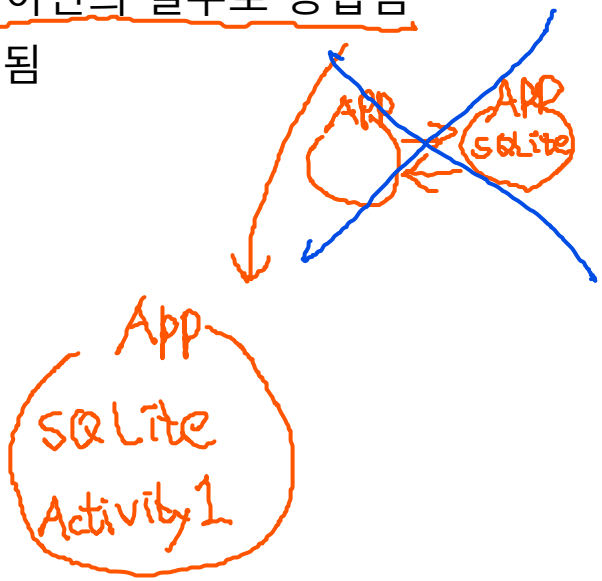

5:25
Lab06
SQLite를 이용한 영속화
SQL(www.sqlite)는 오픈소스로 만들어진 관계형 데이터베이스로 복잡하고 구조화된 애플리케이션 데이터를 저장하고 관리합니다.
ADD


5:27
Lab06
DB Select 결과
Title: SQLite를 이용한 영속화
Content: SQL(www.sqlite)는 오픈소스로 만들어진 관계형 데이터베이스로 복잡하고 구조화된 애플리케이션 데이터를 저장하고 관리합니다.

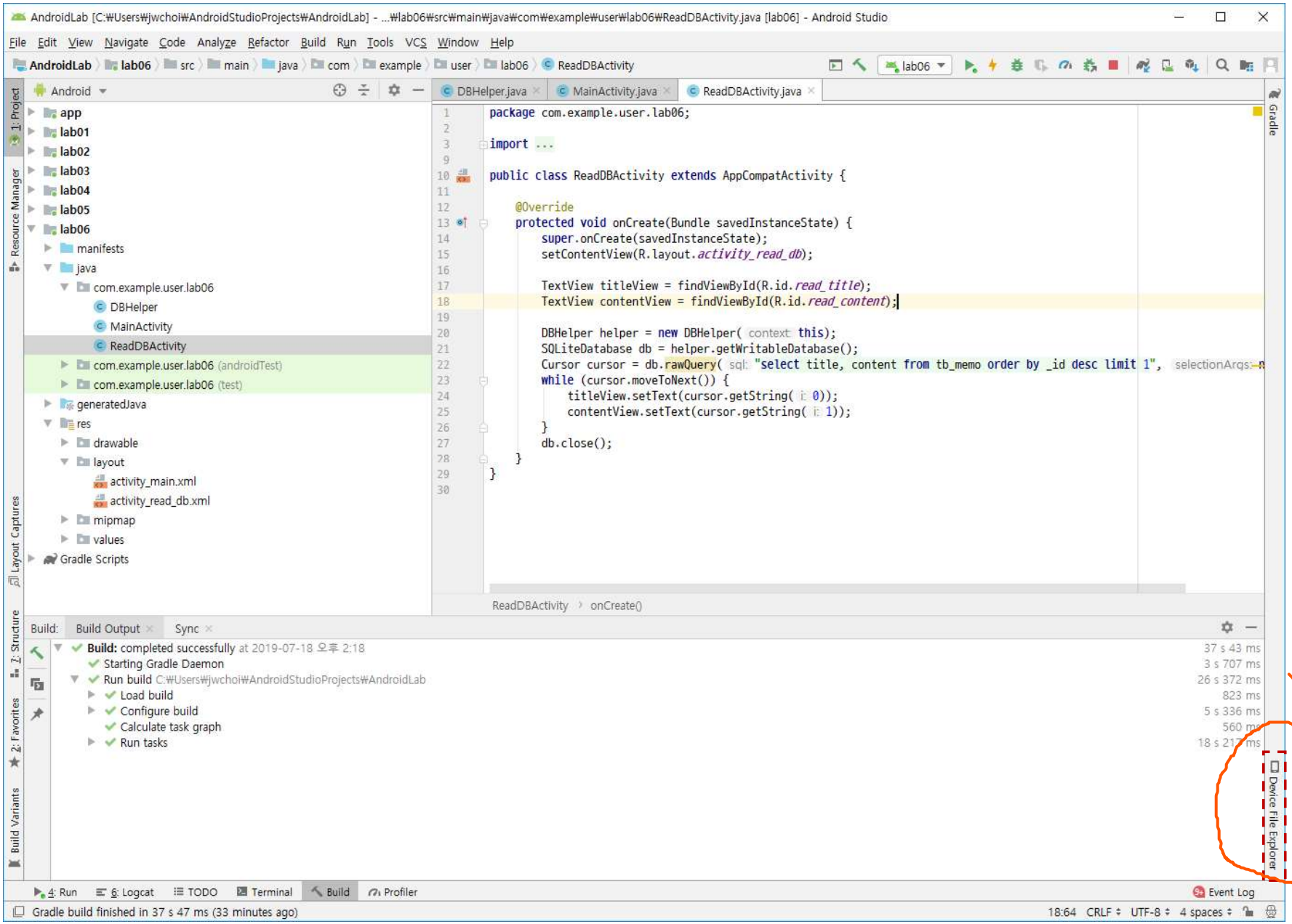
SQLite를 이용한 영속화

- SQLite (www.sqlite.org)는 오픈소스로 만들어진 관계형 데이터베이스
- 프로세스가 아닌 라이브러리를 이용하므로 데이터베이스는 애플리케이션의 일부로 통합됨
- SQLite를 이용한 데이터는 파일에 저장되며 다음과 같은 경로에 저장됨
  - data/data/[package_name]/databases

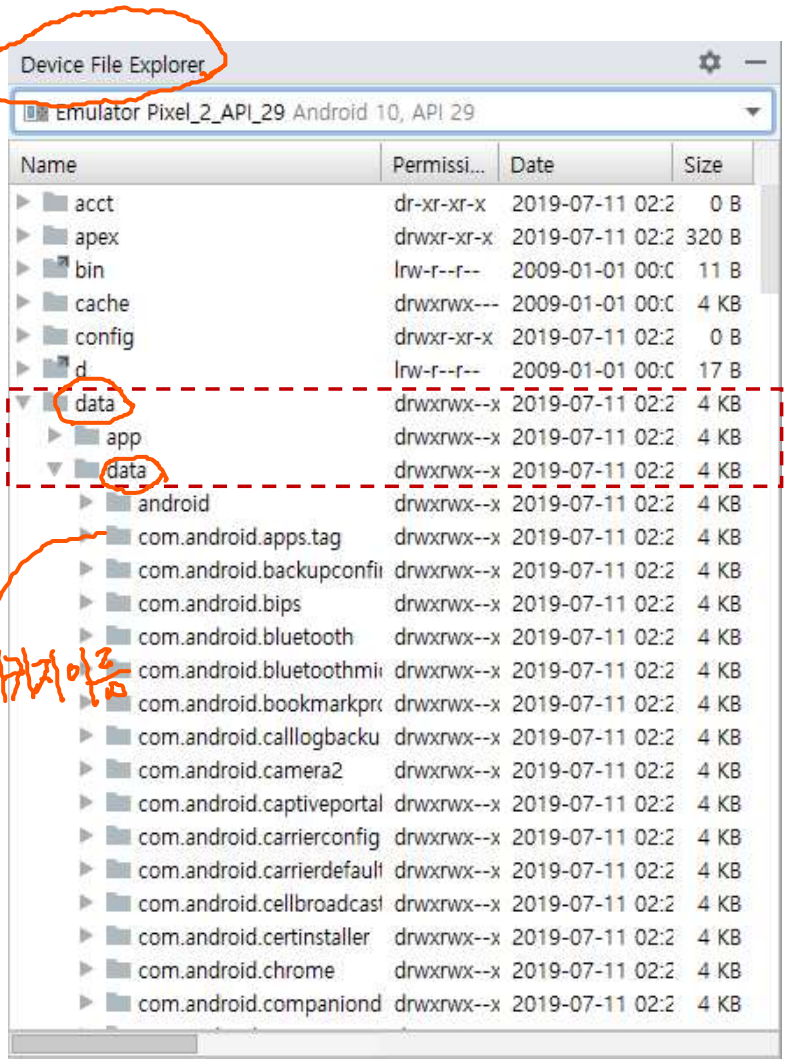




https://developer.android.com/training/data-storage/sqlite.html

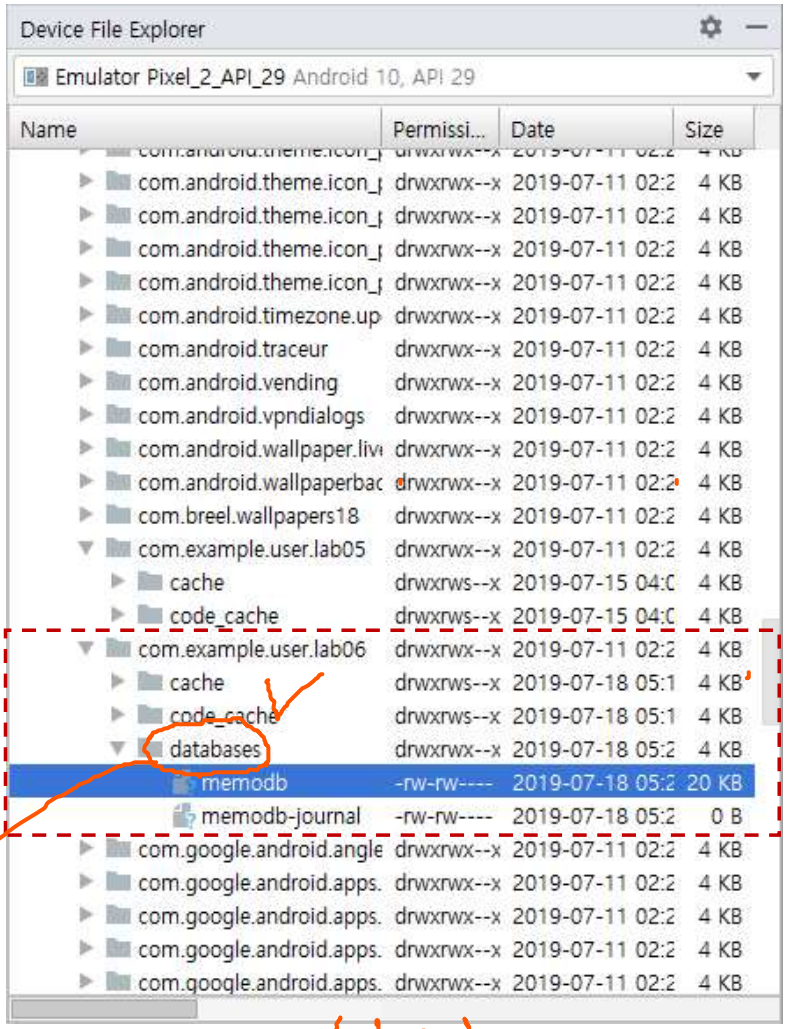

AndroidLab [C:₩Users₩jwchoi₩AndroidStudioProjects₩AndroidLab] - ...₩lab06₩src₩main₩java₩com₩example₩user₩lab06₩ReadDBActivity.java [lab06] - Android Studio
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
AndroidLab lab06 src main java com example user lab06 ReadDBActivity
lab06
Android
app
lab01
lab02
lab03
lab04
lab05
lab06
manifests
java
com.example.user.lab06
DBHelper
MainActivity
ReadDBActivity
com.example.user.lab06 (androidTest)
com.example.user.lab06 (test)
generatedJava
res
drawable
layout
activity_main.xml
activity_read_db.xml
mipmap
values
Gradle Scripts
DBHelper.java
MainActivity.java
ReadDBActivity.java
package com.example.user.lab06;
import ...
public class ReadDBActivity extends AppCompatActivity {
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_read_db);
TextView titleView = findViewById(R.id.read_title);
TextView contentView = findViewById(R.id.read_content);
DBHelper helper = new DBHelper(context: this);
SQLiteDatabase db = helper.getWritableDatabase();
Cursor cursor = db.rawQuery(sql: "select title, content from tb_memo order by _id desc limit 1", selectionArgs:
while (cursor.moveToNext()) {
titleView.setText(cursor.getString(i: 0));
contentView.setText(cursor.getString(i: 1));
}
db.close();
}
}
ReadDBActivity › onCreate()
Build: Build Output Sync
Build: completed successfully at 2019-07-18 오후 2:18
Starting Gradle Daemon
Run build C:₩Users₩jwchoi₩AndroidStudioProjects₩AndroidLab
Load build
Configure build
Calculate task graph
Run tasks
37 s 43 ms
3 s 707 ms
26 s 372 ms
823 ms
5 s 336 ms
560 ms
18 s 21 ms
Device File Explorer
4: Run 6: Logcat TODO Terminal Build Profiler
Event Log
Gradle build finished in 37 s 47 ms (33 minutes ago)
18:64 CRLF UTF-8 4 spaces

DB사용X → DB 폴더 존재X



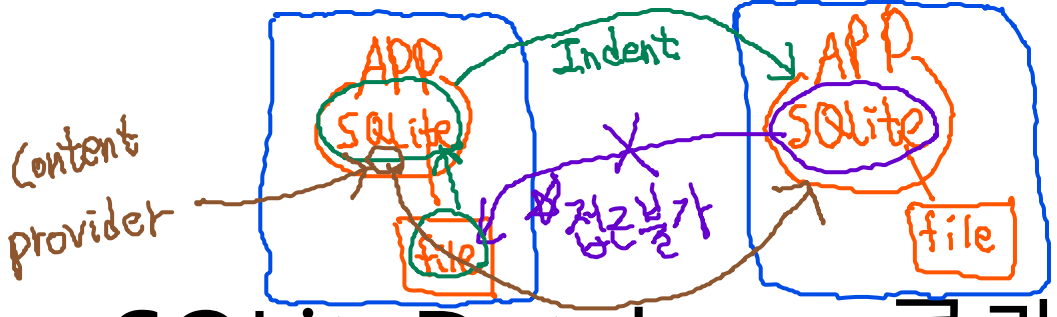
앱이름 = 패키지이름

DB사용 → DB폴더존재



data / data / [패키지 이름] / databases

|| 앱 이름



① Indent로 전달
② Content provider로 전달

SQLiteDatabase 클래스

- 데이터베이스 열기

helper를 통하지 않고

```
SQLiteDatabase db = openOrCreateDatabase("memodb", null);
```

static SQLiteDatabase	openOrCreateDatabase(String path, SQLiteDatabase.CursorFactory factory) Equivalent to openDatabase(path, factory, CREATE_IF_NECESSARY).

Writable

- 데이터 삽입 (,수정, 삭제)

insert update delete

```
db.execSQL("insert into tb_memo (title, content) values (?, ?)", new String[]{title, content});
```

insert, update 등 select 문이 아닌 나머지 SQL 수행

Readable, Writable

- 데이터 검색

rawQuery

Cursor	rawQuery(String sql, String[] selectionArgs) Runs the provided SQL and returns a Cursor over the result set.

```
Cursor cursor = db.rawQuery("select title, content from tb_memo order by _id desc limit 1", null);
```

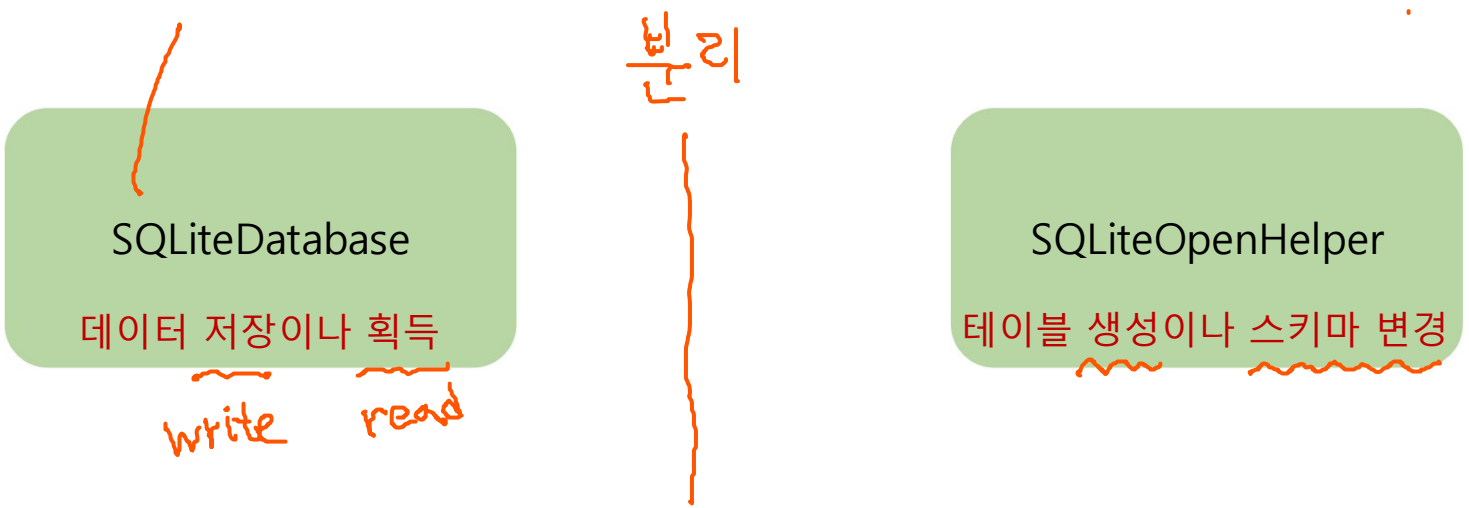
select SQL 수행

Cursor는 선택된 행(row)의 집합 객체
- moveToNext(): 순서상으로 다음 행 선택
- moveToFirst(): 가장 첫 번째 행 선택
- moveToLast(): 가장 마지막 행 선택
- moveToPrevious(): 순서상으로 이전 행 선택

```
while (cursor.moveToNext()) {
    titleView.setText(cursor.getString(0));
    contentView.setText(cursor.getString(1));
}
```

SQLiteOpenHelper 클래스

- SQLiteDatabase와 Cursor 클래스만 사용해도 모든 SQL 문을 수행할 수 있음
- But
  - SQLiteOpenHelper 클래스를 사용하면 편리한 점이 많음
  - SQLiteOpenHelper: 데이터베이스 관리만을 목적으로 하는 클래스



SQLiteOpenHelper 클래스