

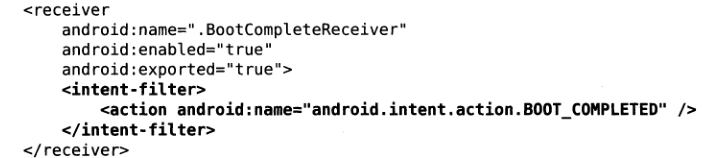
标准广播，有序广播

动态注册广播接收器



Ondestroy（）里写unRegisterReceiver

Mainifest中静态注册

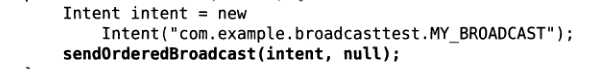


Exported属性表示广播接收器是否接收本程序以外的广播

发送自定义标准广播



发送有序广播



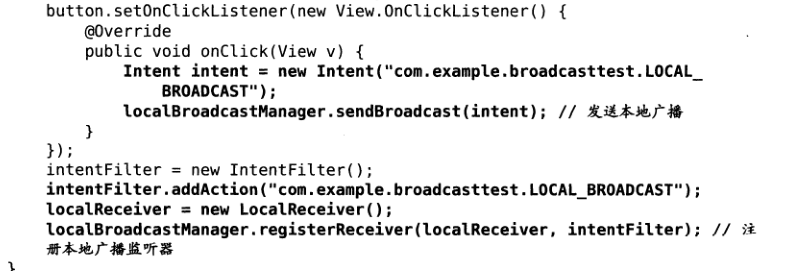
接收器优先级



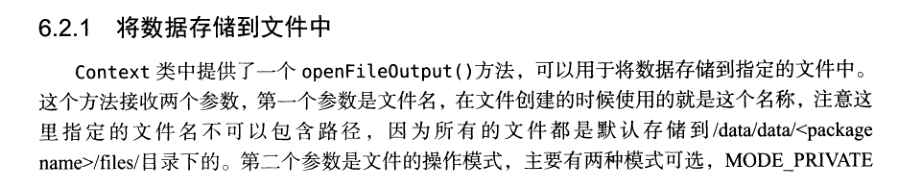
截断有序广播传播

abortBroadcast();

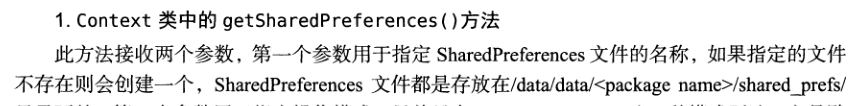
本地广播发送，注册



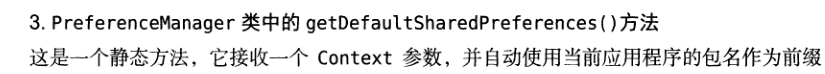
本地广播无法通过静态注册的方式来接收

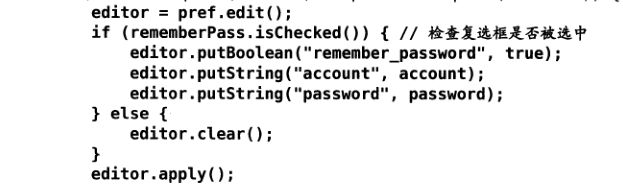


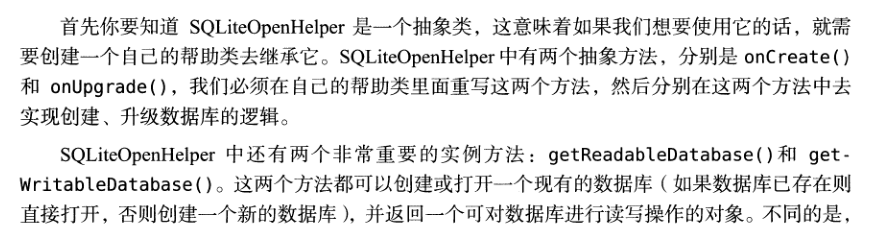
OpenfileInput()

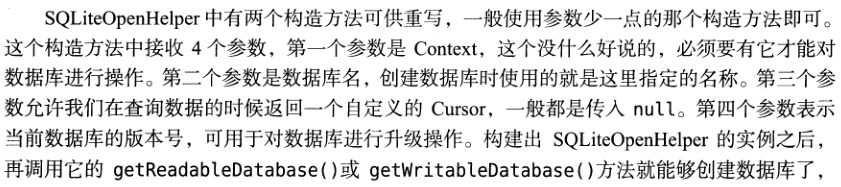




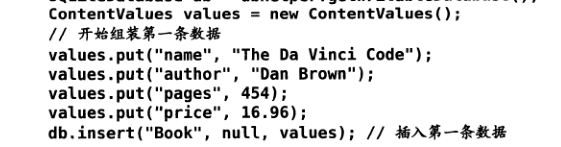




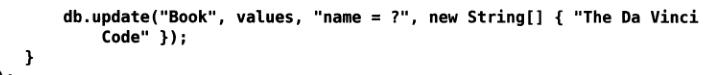




Sqllite插入数据



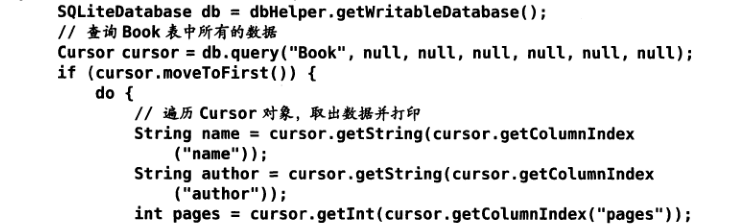
更新

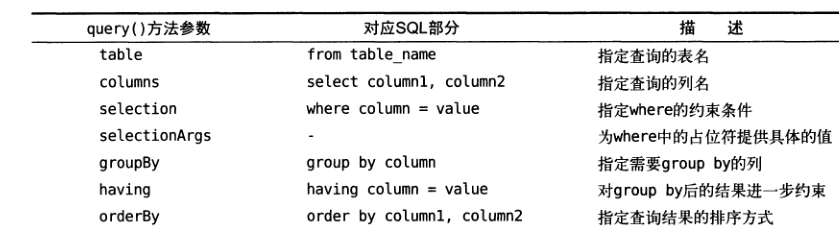


删除

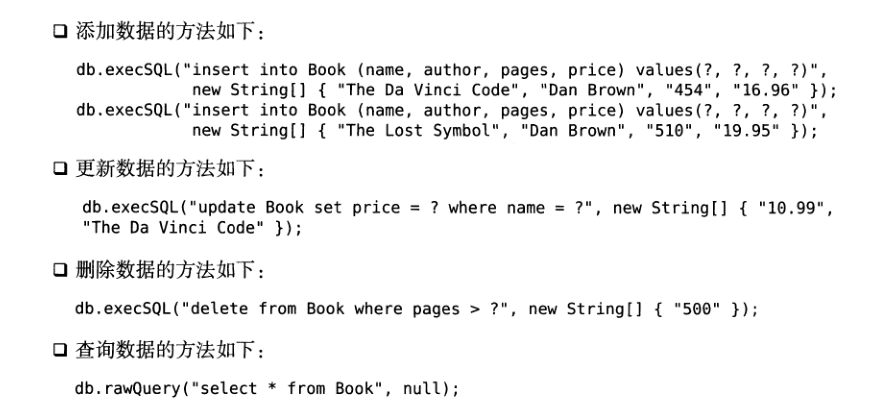


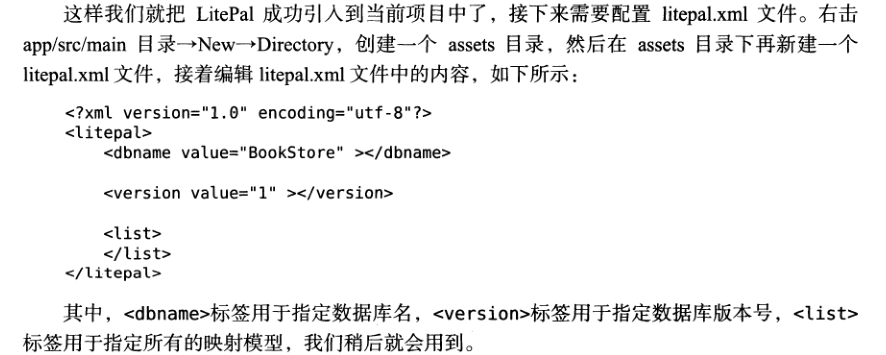
查询



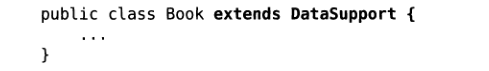


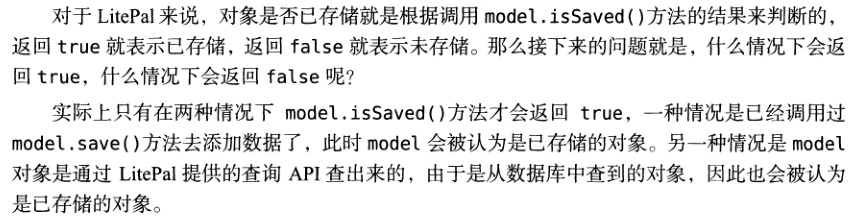
Sql操作数据库





只要进行任意一次数据库操作，数据库会自动创建出来



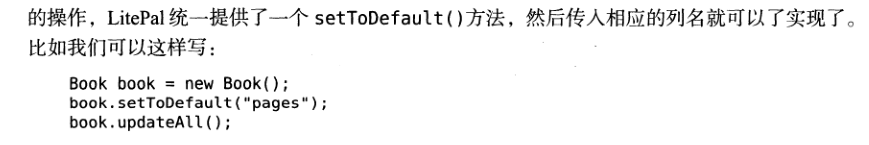


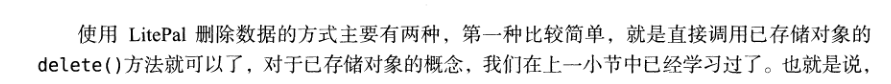
发生改动，和新new的对象 isSaved（）返回false

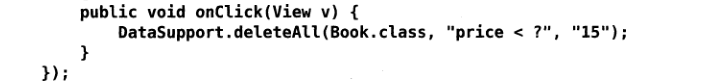
相比于查出来再更新。直接更新











* DataSupport：find/findFirst/findLast/findAll/findBySQL/isExist
* DataSupport：select/where/order/limit/offset 查询条件

Song song = DataSupport.find(Song.class, id);

Song song = DataSupport.findFirst(Song.class); // 查询第一条

Song song = DataSupport.findLast(Song.class); // 查询最后一条

List<Song> allSongs = DataSupport.findAll(Song.class);

List<Song> songs = DataSupport.where("name like ?", "song%").order("duration").find(Song.class);

List<Song> songs = DataSupport.where("name like ?", "song%").order("duration").limit(3).offset(1).find(Song.class); // 分页查询

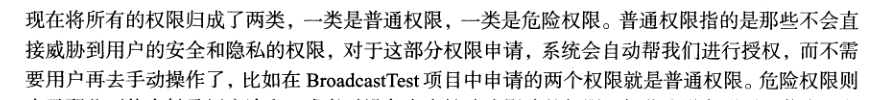
// 联合查询，可以查询关联表

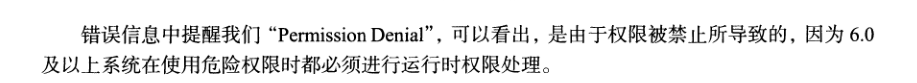
Album album = DataSupport.find(Album.class, id, true);

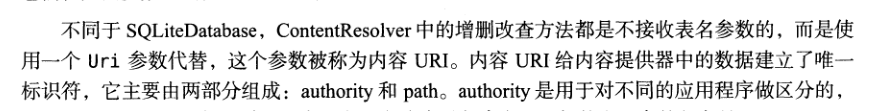
List<Song> songs = album.getSongs();

<https://www.jianshu.com/p/53bf32db4da0>

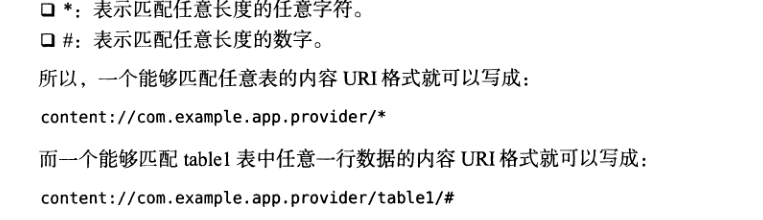
跨程序共享数据——内容提供器

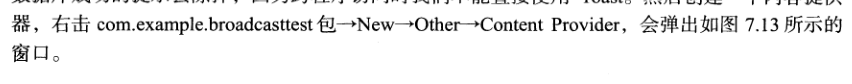


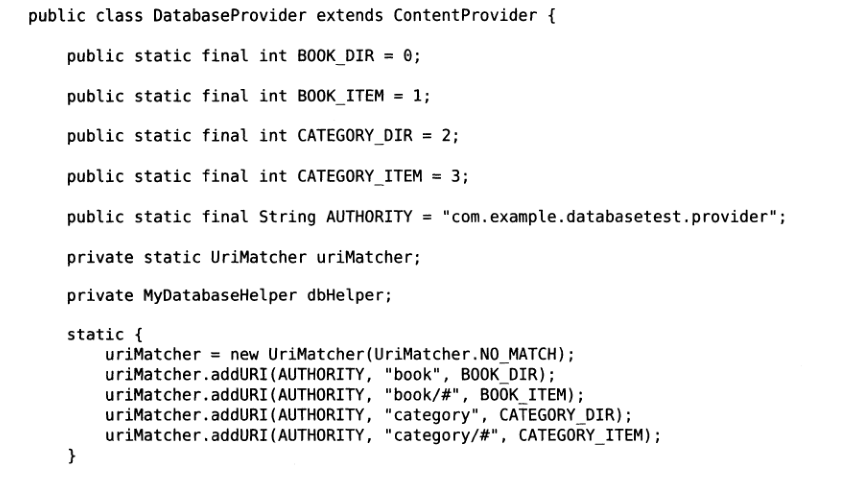


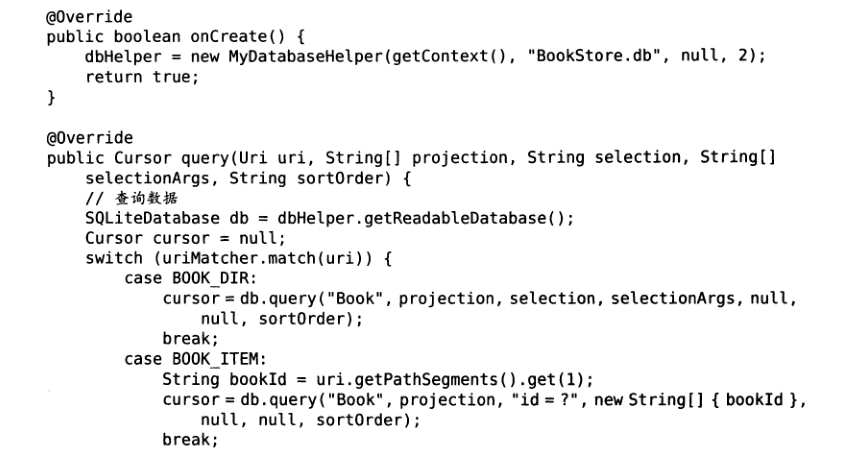


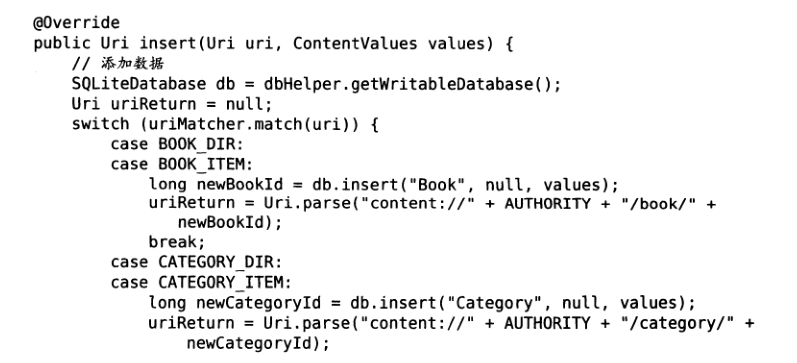




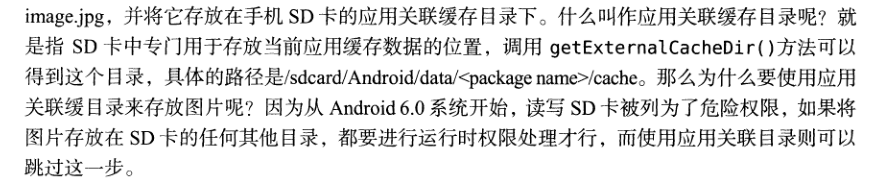












共享根目录

File sdCard = Environment.getExternalStorageDirectory();

File directory\_pictures = Environment.getExternalStoragePublicDirectory(Environment.DIRECTORY\_PICTURES);

目录为data/data/< package name >/files/

File filesDir = getFilesDir();

参数传入的为null，这样默认访问的是files文件夹，我们可以指定子文件夹

File externalFilesDir = getExternalFilesDir(null);

C:\Users\lile\AppData\Local\Temp\1533524029(1).jpg

File externalFilesDir = getExternalFilesDir("Caches");

<https://blog.csdn.net/d_o_n_g2/article/details/78027112>

在子线程中调用runOnUiThread（）

