院 系 数据科学与计算机学院 学 号 　16337327　 姓 名 郑映雪

【实验题目】**Android数据库实验**

【实验目的】学习Android的ContentProvider、ActionBar、Json转换、SQlite操作、Fragment、PreferenceActivity、CustomListView、SimpleListView、AlertDialog、ProgressDialog、menu、Handler等。

【功能说明】

本实验需要编制两个项目程序，第一个是词典程序(词典app)，词典程序除了本身作为词典使用外，还可以把单词提供给其它需要使用的单词和例句的程序使用。第二个是背单词程序(被单词app)，利用词典程序获得单词，然后通过测验的方法背单词。

【实验说明】

（a）下面的实验内容把本次实验分成了很多步骤来完成，如果只完成其中一部分功能，在最后指明完成了哪些步骤。

（b）除了单步测试，调试可以采用Log.d("测试", ""+i); 通过Android Monitor，还可以使用单元检测。

（c）做每个步骤前，最好先把参考源码运行一遍。

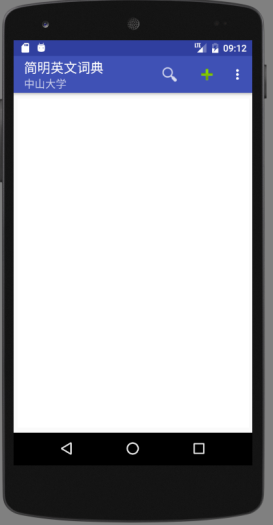
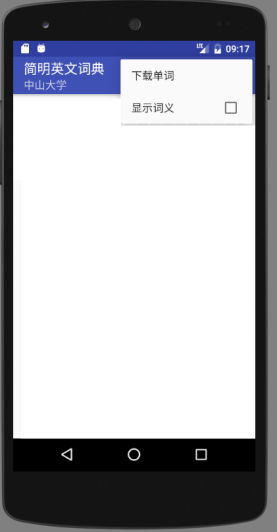
（d）可以先看一下老师的类设计（在本报告末尾）

【实验内容】

（一）词典核心功能设计

项目：GrandDictionary

1、做一个如下ActionBar界面。

**编程说明**： **android:icon="@drawable/ic\_action\_search"**

**android:icon="@android:drawable/ic\_input\_add"**

**android:checkable="true"**

**android:checked="false"**

**参考源码**：ActionBar

2、点击下载菜单完成单词下载(Json格式)，转换为Object，并保存到SQlite数据库中。如果单词已存在，则覆盖它。



**数据表：**db.execSQL(**"CREATE TABLE dict(\_id integer primary key autoincrement, word varchar(64) unique, explanation text, level int default 0, modified\_time timestamp)"**);

其中，level为单词难度，值越大越难

查看Json数据：**http://172.18.187.9:8080/dict/**

**注意事项**：

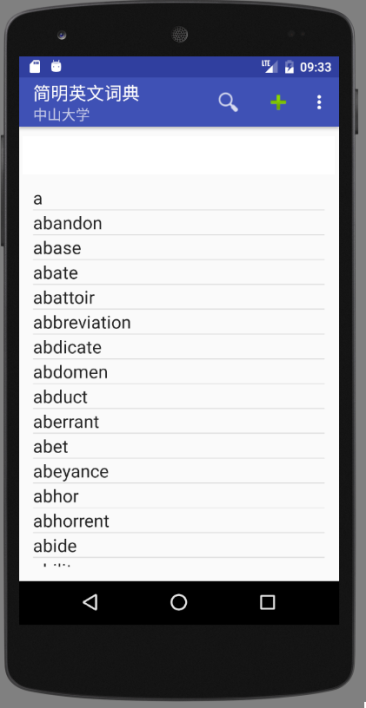
(1)AndroidManifest.xml中加入允许互联网访问：  
<**uses-permission android:name="android.permission.INTERNET"**/>

(2) 最好采用子线程进入下载：

**new** Thread() {  
 @Override  
 **public void** run() {  
 **try** {  
 DownloadDict downloadDict = **new** DownloadDict(**mContext**,**dictDb**);  
 downloadDict.start(**""**);  
 } **catch** (Exception ex) {  
 ex.printStackTrace();  
 }  
 }  
}.start();

**参考源码或数据**：WebServiceDemo（getJsonWeather方法）、Json1（推荐）、Json2、SqliteCv

3、用SimpleListView把数据库中的数据显示出来。



**参考源码**： SqliteCv、ListViewSimple (也可使用Cursor，参见ListViewCursor，不推荐)

4、增加一个ContentProvider。（可以结合背单词功能ContentResolver一起做）

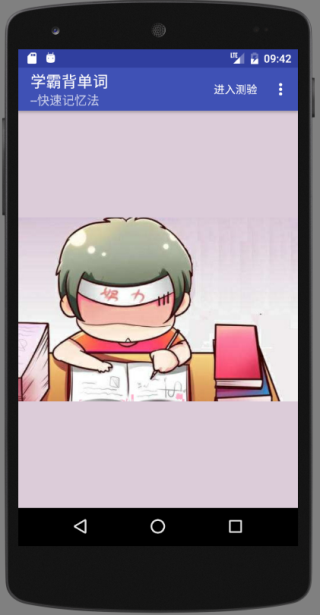
**参考源码**： SqliteCv、FirstProvider

**提示**：修改AndroidManifest.xml(用到的四大组件都要在其中配置)

（二）词典核心功能设计

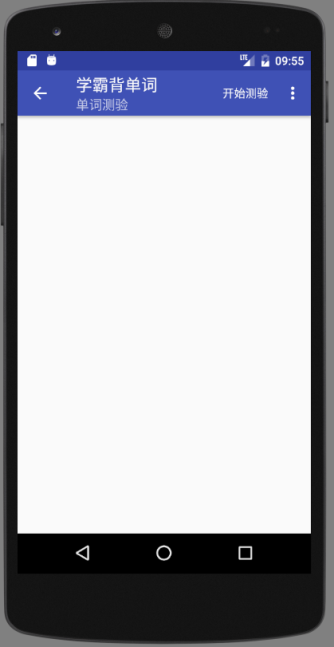
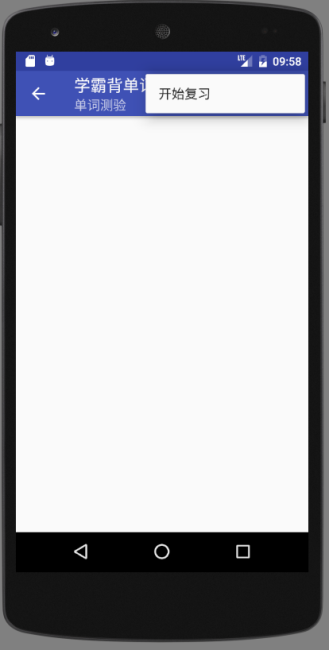
项目：GrandWordRemember

5、做一个主界面(MainActivity)

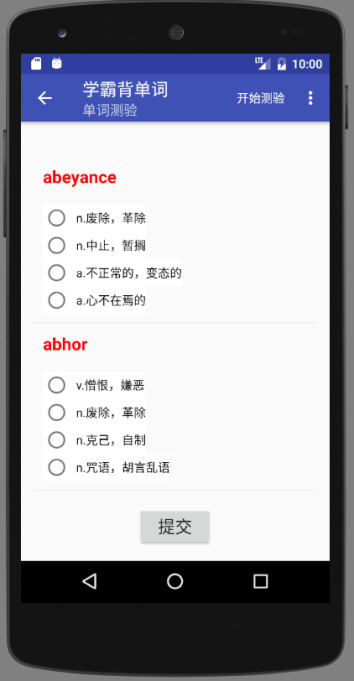
**参考资源**： rem.jpg

6、点击“进入测验”，进入测验界面（TestActivity）

**参考源码**：NewStartActivity

7、点击上面界面的开始测验，编写通过ContentResolver从词典程序中取前10个单词的程序并显示出来。选项从这10个单词的解释explanantion中随机选取，并包含要测验的单词，顺序也要打乱。



**参考源码**：FirstResolver、SpinnerCustom（参考做成自定义ListViewCustom）

**编程说明：**取10条记录的方法，OrderBy参数加上limit 10，即 "word limit 10"

**参考地址**：https://blog.csdn.net/Sunjianhua360/article/details/51394488

8、点击菜单“新增单词”，通过对话框为词典程序增加单词。



**操作说明**：要查询输入的单词是否存在，不存在则增加，存在则修改（允许覆盖）。

**编程提示：**可以单独定义一个类放对话框的功能。把Activity的上下文Context带入就可以了。上下文Context表示在什么地方显示这些对话框。

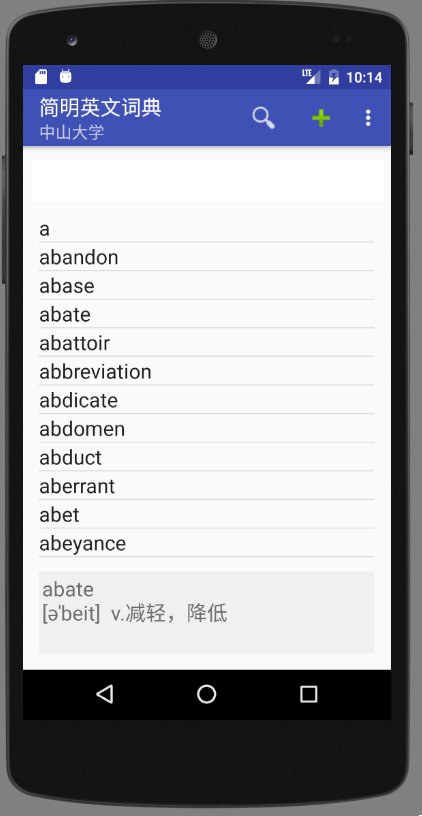
\* Context在Activity中Context可以用this或ActivityName.this取得，在事件处理程序中可以用ActivityName.this取得。

**参考源码**：FirstProvider 、FirstResolver、SqliteCv、AlertDialog

（三）词典扩展功能设计

项目：GrandDictionary

9、增加信息显示栏，点击单词(setOnItemClickListener)将显示含义。



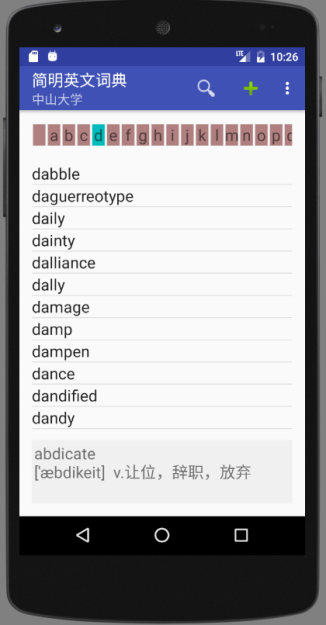
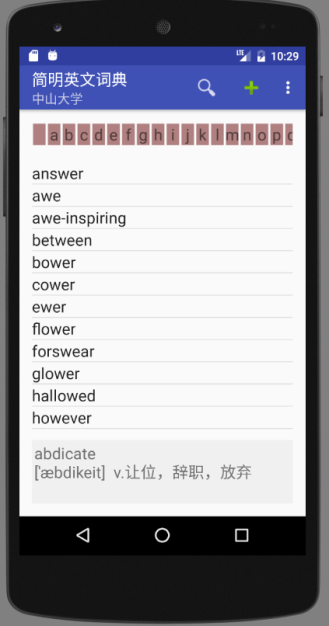
**参考源码**：ListViewSimple，SqliteCv

10、长点击单词(setOnItemLongClickListener)将显示弹出式菜单。



**参考源码**：Menu （PopupMenu）

11、增加单词查询功能

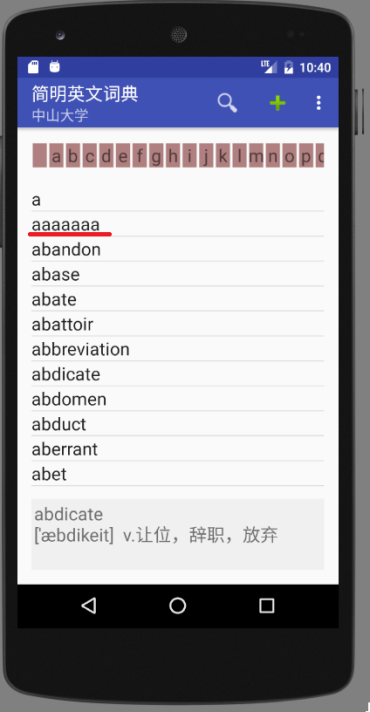
**功能说明**：在列表上方增加一个字母表，点击时只显示以这些字母开头的单词；点击查询图标可以查询包含输入子串的单词。

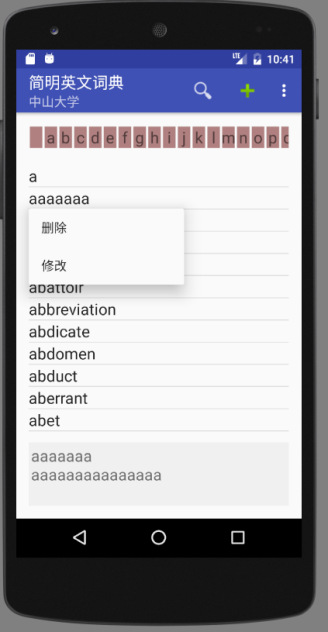
**参考源码**：DynamicView,HorizontalScrollView（自己上网查找使用方法）、SqliteCv

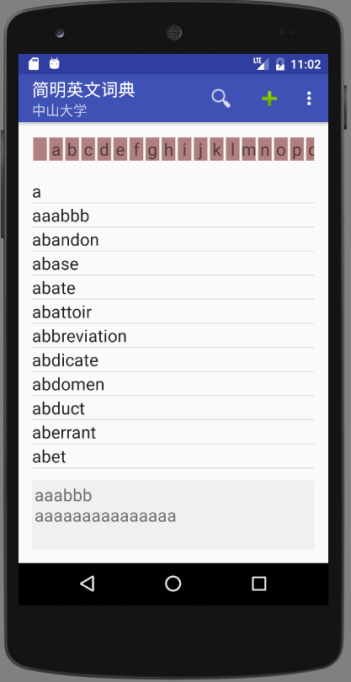
**编程参考**：每个单词可以从字母表” a…z ”取出一个字符(""+charAt()或substring()),然后创建TextView。空格查出所有单词

查询采用like的方法

12、通过图标和弹出式菜单用对话框方式实现增加、删除和修改单词。



**参考源码**：AlertDialog、SqliteCv

13、通过菜单可以让词典直接显示语义。



**参考源码**：ListViewSimple、SqliteCv，getPxFromDpi.txt

**功能说明**：显示语义时不显示信息栏。

14、下载和保存词典时增加一个进度显示。



**编程说明：**进度对话框放在UI主线程，下载和保存到数据库放到子线程，通过消息把进度传递给UI主线程。

**参考源码**：ProgressDialog（里面有一个Handler的例子），NewHandlerRunnable

（四）背单词扩展功能设计

项目：GrandWordRemember

15、建立统计数据库

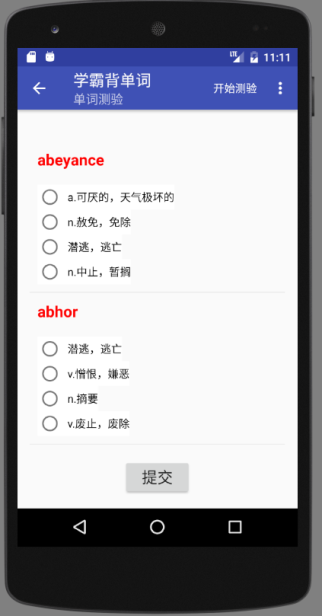
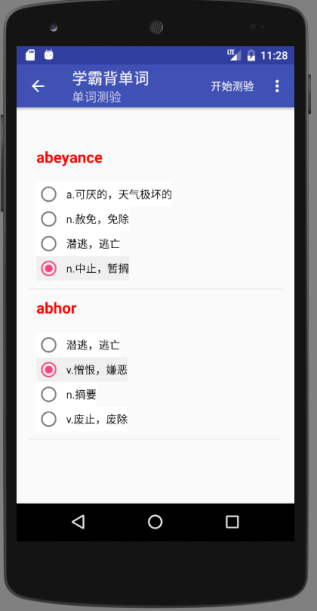
db.execSQL(**"CREATE TABLE words(\_id integer primary key autoincrement,word varchar(64) unique,level int default 0, test\_count int default 0, correct\_count int default 0,last\_test\_time timestamp)"**);

level 为单词单独级别（来自词典），test\_count为测试总次数，correct\_count为答对的次数。

16、每次取出的单词都是上述words表中没有的，实际做法可以先找出words最大的单词，然后从词典程序中按词典序取回接着的10个单词。

**参考源码**：FirstProvider，FirstResolver

17、提交后进行批改，看哪些题答对了，用背景色表示正确答案，并保存新单词到本地words表，修改已存在单词的统计数据。未选的作为错误。提交后隐藏提交按钮。

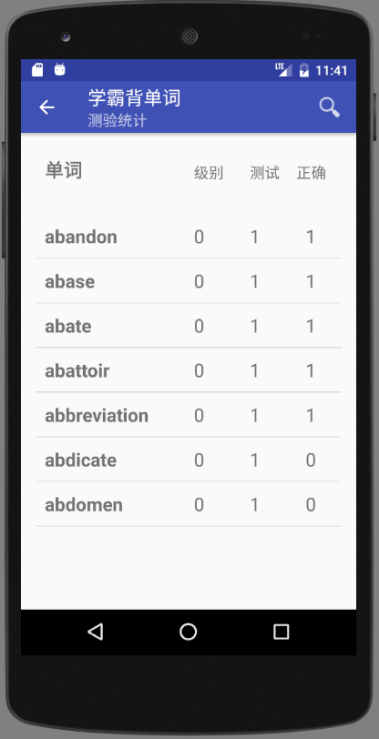
  

**参考源码**：SpinnerCustom（改为ListView），TestAdapter.java

**参考地址**：*https://blog.csdn.net/Sunjianhua360/article/details/51394488*

*https://blog.csdn.net/h623691640/article/details/52893166(*滚动混乱问题)

18、显示学习统计数据(StatActivity)



**参考源码**：ListViewSimple

19、增加“系统设置”功能，可以选择单词颜色，可以设置每次测试的单词数，以及保存时是否进行统计。要修改程序符合已设置的参数。

**参考源码**：NewPreferenceActivity

（五）自己设计功能（选做）

例如：

（a) 背单词的开始复习是从已学过的单词中选取要记忆的单词

（b）背单词的查找单词，从词典中找出单词。

（c）背单词的统计加入首字母筛选查询。

（d）把背单词的ActionBar上开始测试用带背景图的圆形图标代替

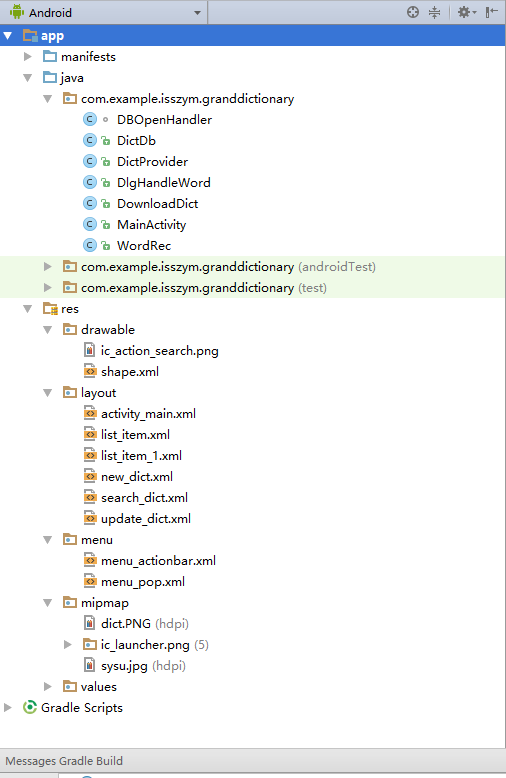
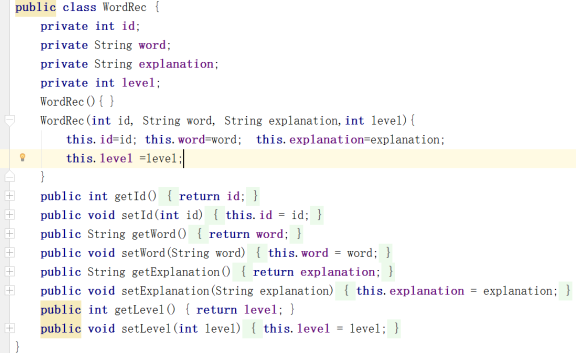
（e）利用中文解释，选择英文单词

（f）给出中文解释，填写英文单词

*……*

【参考Android Studio工程截屏】

词典程序(GrandDictionary)

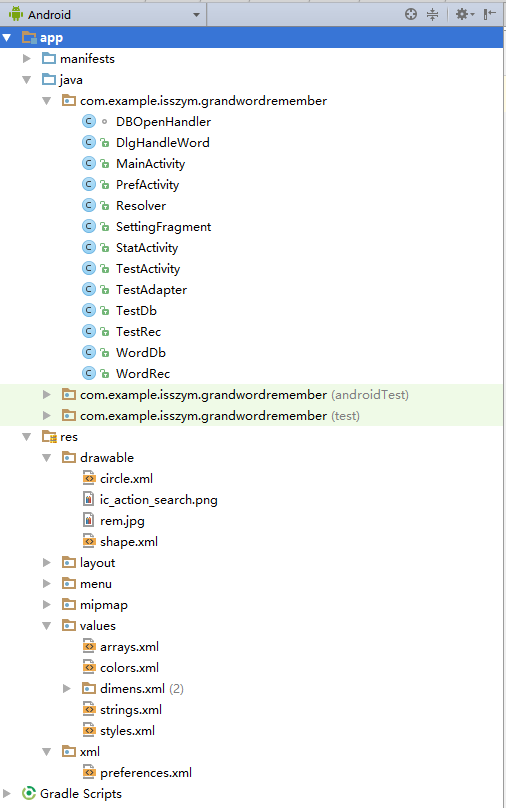
 

DbOpenHandler：打开数据库类 DictDb:所有数据库操作类

DictProvider：词典ContentProvider类 DlgHandleWord：所有对话框类

DownloadDict：下载词典类 MainActivity：主界面类 WordRec：保存单词记录类（见上图）

词典程序(GrandWordRemember)



DbOpenHandler：打开本地数据库 DlgHandleWord：包含所有对话框操作的类

MainActivity：主界面类 PrefActivity：配置界面类

Resolver：用于访问ContentProvider的ContentResolver类

SettingFragment：配置用的Fragment的类 StatActivity：统计界面类

TestActivity：测验界面类 TestAdapter：用作测验界面ListView的Adapter类（继承SimpleAdapter）

TestDb：本地数据库操作类 TestRec：本地统计记录类 WordDb：（远程）词典操作类

WordRec：单词记录类

*完成后程序运行截屏及简要说明：*

【词典程序】

（1）

（2）

（3）

…

【背单词程序】

（1）

（2）

（3）

…

*完成后的主要源码(.java和.xml)：*

【词典程序】

（1）

（2）

（3）

…

【背单词程序】

（1）

（2）

（3）

…

【完成情况】

完成了哪些内容？(√完成 ×未做或未完成)

1 [ ] 2 [ ] 3[ ] 4[ ] 5 [ ] 6 [ ] 7[ ] 8[ ] 9[ ] 10[ ]

11 [ ] 12 [ ] 13[ ] 14[ ] 15 [ ] 16 [ ] 17[ ] 18[ ] 19[ ]

自己增加的功能：  
（1）

（2）

（3）

…

【实验体会】

首先说一个不算制作里的问题：打不开老师的文件，没有办法build……最后通过修改app文件夹里build.gradle文件里的sdk版本成功解决。下面是制作时遇到的问题或体会：

第一个项目：

首先在做界面的时候，我发现老师给的demo主题是绿色的，好奇如何可以改变主题的颜色。后来新建了一个项目与demo做对比，发现只要在values文件夹里，先修改colors.xml里的各个颜色，再在styles.xml里设置各个部分的颜色即可。在创建数据库时，发现app出现崩溃现象，原因在于不可以把autoincrement加在text类型的属性上，会出现：AUTOINCREMENT is only allowed on an INTEGER PRIMARY KEY。

在数据库导入的操作中，遇到的主要的内容总结如下：

一、把Json转换为object的步骤：

1、读取输入流为JsonArray的格式；

2、遍历JsonArray,创建WordRec类将其转化为object，提取键值对插入到数据库中。

二、Cursor对象绑定数据时，Cursor必须包含“\_id”字段，否则将无法完成数据绑定。所以在写sql语句时，要使用select xxx as \_id……不然会造成程序崩溃。

在写contentprovider时，出现报错：

Installation failed with message INSTALL\_FAILED\_CONFLICTING\_PROVIDER。原因是这个provider已经有应用使用（可能我原本运行了老师的示例代码），只要在注册时改一个名字即可。（比如我就随性地在后面加了个数字）

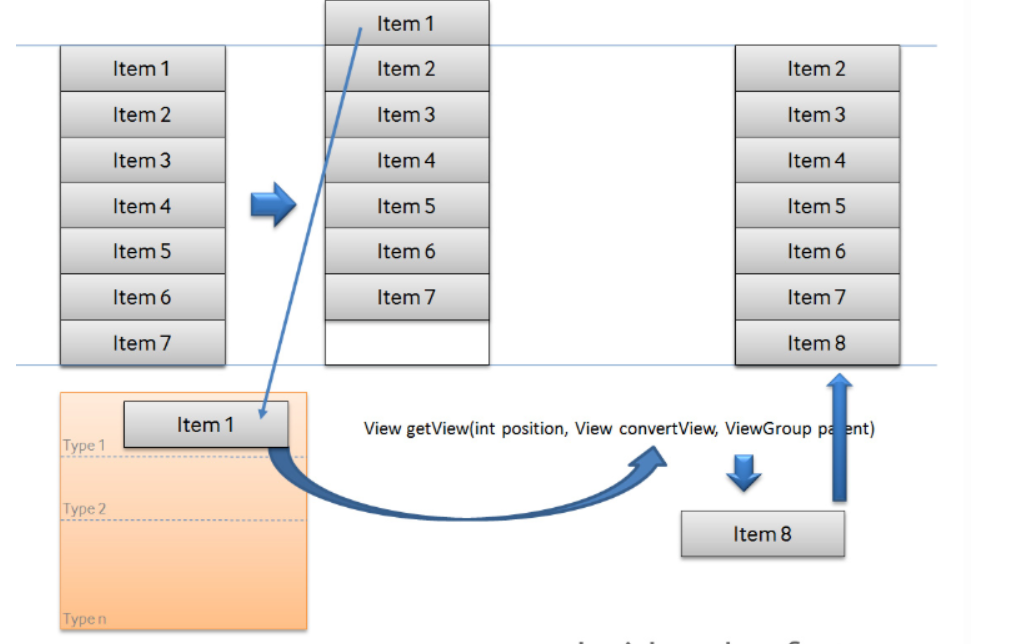
第二个项目：

在做界面的时候，首先设置背景颜色要为粉红色，但是要修改style文件很不方便。这个时候我想了一个投机取巧的办法——截图rem的背景，保存到drawable中，此时就可以直接使用background啦。

此时连接项目一的provider发现返回的是一个空指针，说明这个时候的数据库是一个空对象。经已经踩过雷的小伙伴帮助，我发现原来项目一创建了一个新的数据库，此时那些下载部分的insert需要调用content的insert，修改之后，项目二可以连接上数据库下载的单词了。

在做listview嵌套radiogroup时，自带的适配器已经无法起作用，所以需要进行重构。在进行音标删除的时候，我也特别注意到要使用equals而不是==判断相等才能成功。

在测试单词功能制作时出现了ListView嵌套RadioGroup时选中混乱、显示现象，原因是listview的复用机制导致的监听器混乱，查阅了一下可以如下图所示：



博客作者给的解释是：屏幕滑动第一条数据消失，复用机制就会发生了，先把item1放到recycler中，等待屏幕底部数据出现，当item8出现时，就会复用item1的布局和数据。解决方法是每次取消掉监听器，每次选时进行先行条件判断，如果已经选中则设置为未选中，每次进行恢复就可以了。

在做“新增单词”对话框的时候，进入自带的模拟器，对话框显示了一两秒，模拟器就会崩溃退出，遍寻不到原因。最终我下载了Genymotion（反正自带的模拟器太……慢……），在Genymotion运行就不会闪退了……看来是模拟器的问题，这也是我无法解决的问题。

在做长按事件时，需要注意的细节是把onItemLongClick返回设置为true，否则长按是会执行setOnItemClickListener。

在做增删改功能时，由于我使用适配器时SimpleCursorAdapter，所以实时更新非常麻烦。上网查询之后发现可以直接从最基本的cursor入手，在每进行一步相关操作，即设置一个新的cursor并赋给当前的cursor，即可实现实时的修改了显示了。在做查找条时，我如法炮制，点击一个按钮后即赋予一个新的cursor，这个cursor来源于以该按钮字母为开头的query集合。在变背景颜色的时候出了一些小插曲，即不点击了也不会变成原来的粉色背景，这时候我遍历了layout元素里面的所有其他元素，全部设置为原背景即可。

在进行“显示语义”实现时，出现了只能根据当前页面显示，导航都无法显示的问题，解决办法是另用一个adapter和cursor刷新。在写查找的时候也犯了一个低级错误——取出edittext的时候没有gettext导致查找的是空白（没有想到低级错误，导致查别的错查了很久）。

【交实验报告】

(a) 每位同学独立完成本实验内容并填写实验报告。

(b) 截止时间：2019年6月2日23:00

上交作业网站：<http://172.18.187.9/netdisk/default.aspx?vm=16and>

文件夹: /实验上交/02、安卓数据库实验

上传文件：学号\_姓名\_数据库实验.doc (实验报告)   
学号\_姓名\_数据库实验.rar (包含所有项目，注意删除build和app/build目录)

学号\_姓名\_数据库实验.mp4 (动态运行)