

手机平台应用开发实验报告

学号： 1433198

班级： 周五下午班

姓名： 黄建武

实验名称： Lab3

一． 参考资料

<https://developer.android.com/guide/index.html> Android 官方文档

<http://blog.csdn.net/treaturebeauty/article/details/323121>

Vector.toArray 与强制类型转换

<http://blog.csdn.net/sysukehan/article/details/52022307> 实现圆形背景

<http://blog.csdn.net/ztp800201/article/details/7322110> 安卓文件读写

二． 实验环境

系统： Windows 10

工具： Android Studio 2.1.3

JDK： 1.8

minSdk： 19

targetSdk： 23

三． 实验目的

1. 复习事件处理
2. 学习 Intent、Bundle 在 Activity 跳转中的应用
3. 学习 ListView 以及各类适配器的用法

四． 实验步骤

1. 阅读实验要求文档和课件。

2. 运行 TA 的 Demo 程序，了解实验要求细节。

3. 实现从文件中读入信息并存进字符数组。

```
Vector<String> nameV = new Vector<String>();
InputStreamReader inputStreamReader = new InputStreamReader(getResources().openRawResource(R.raw.list));
BufferedReader reader = new BufferedReader(inputStreamReader);
try {
    String line = reader.readLine();
    while (line != null) {
        nameV.add(line.split(" ")[0]);
        line = reader.readLine();
    }
} catch (IOException e) {
    e.printStackTrace();
}
name = (String[]) nameV.toArray(new String[10]);
```

4. 设置 SimpleAdapter，完成 ListView 的样式。

5. 完成 ListView item 的点击和长按功能。

```
private void addClickListener() {
    ListView namelist = (ListView) findViewById(R.id.namelist);
    assert namelist != null;
    namelist.setOnItemClickListener(new AdapterView.OnItemClickListener() {
        @Override
        public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
            startActivity(new Intent(MainActivity.this, Activity2.class));
        }
    });
    namelist.setOnItemLongClickListener(new AdapterView.OnItemLongClickListener() {
        @Override
        public boolean onItemLongClick(AdapterView<?> parent, View view, int position, long id) {
            clickPos = position;
            isDelete.setMessage("确定删除联系人" + data.get(clickPos).get("name") + "? ").show();
            return true;
        }
    });
}
```

6. 对第一个页面进行调试。

7. 实现详情页面的布局，上方的 1/3 使用 RelativeLayout，下面的 2/3 是一个 LinearLayout。

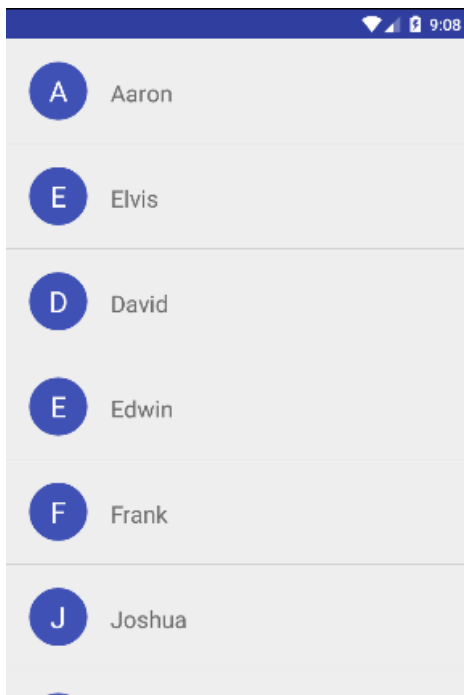
8. 从文件中读入数据，添加事件监听器，在页面跳转中传入参数，根据点击的人的名字，在显示不同的文字和背景颜色。

```
String [] user = (info.get(name));
TextView phoneNum = (TextView) findViewById(R.id.phoneNum),
    phoneDetail = (TextView) findViewById(R.id.phoneDetail),
    nametv = (TextView) findViewById(R.id.name);
assert nametv != null;
nametv.setText(user[0]);
assert phoneNum != null;
phoneNum.setText(user[1]);
assert phoneDetail != null;
phoneDetail.setText("手机 " + user[3]);
RelativeLayout top = (RelativeLayout) findViewById(R.id.topDiv);
assert top != null;
top.setBackgroundColor(Color.parseColor(user[4]));
```

9. 为页面 2 的 ListView 设置适配器。
10. 对页面 2 的改善布局，进行调试。
11. 将 SimpleAdapter 改为自定义 Adapter。

五． 实验结果截图

初始页面



详情页面



不同详情页面显示不同颜色和信息



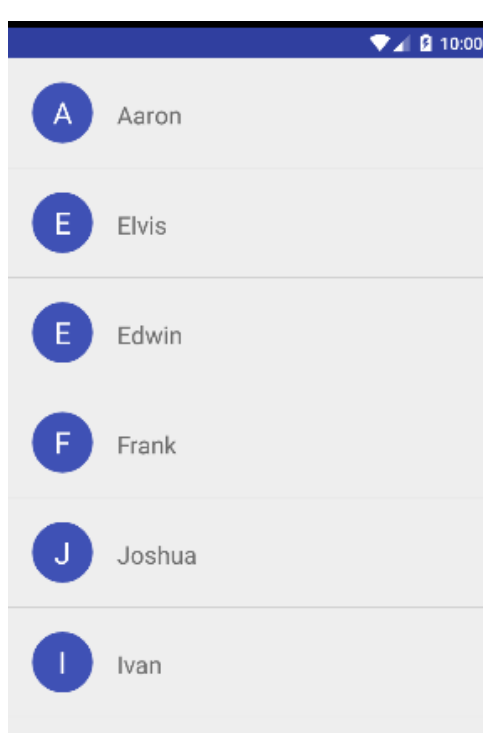
点击星星图标



长按联系人



成功删除联系人



六． 实验过程遇到的问题

1. java.lang.ClassCastException: java.lang.Object[] cannot be cast to java.lang.String[]

```
name = (String[]) nameV.toArray();
```

直接将 Object[] 转化为 String[] 会报错，需要指定转换的类型和数组大小。

```
name = (String[]) nameV.toArray(new String[10]);
```

2. 点击联系人列表没有跳转
写监听器时靠补全 setOnItemClickListener 写成了 setOnItemSelectedListener
3. 跳转到详情页面程序崩溃
新建页面 2 后没有更新 AndroidManifest.xml
4. 长按联系人后出现删除对话框后页面又跳转到了详情页面
原因：一开始 onItemClick 的返回值设置为 false，表示并没有完成处理该事件，更希望其他方法继续对其进行处理，于是又会触发 onItemClick 函数。
将返回值改为 true 解决了问题。
5. 图片背景不透明



没有为图片设置背景颜色，而且图片本身是 png 的，可是还是有灰色背景，原来一开始是把 ImageButton 的 src 设置为图片，后来把 background 设置为图片，背景就透明了，可是图片非常小，最后改为 ImageView 加监听器成功解决了问题。

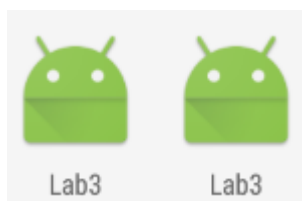
6. 布局出错

D:\Study\Android\AndroidStudioProjects\Lab3\app\src\main\res\layout\activity2.xml

- ! No resource found that matches the given name (at 'layout_alignTop' with value '@id/chat').
- ! No resource found that matches the given name (at 'layout_alignBottom' with value '@id/chat').
- ! No resource found that matches the given name (at 'layout_toStartOf' with value '@id/chat').

分割线在聊天图片的空间之前，定位时找不到 id 对应的控件，将两个控件在 xml 文件中调换位置即可。

7. 运行时生成了两个 apk



其中一个 apk 一打开就崩溃，报错说访问空数组，找了好久发

现一个进去是第一个页面一个是直接进入第二个页面的，第二个页面需要接受来自第一个页面的参数，直接进入第二个页面参数为空所以程序奔溃。通过修改配置文件解决了问题。

```
<activity android:name=".MainActivity">
    <intent-filter>
        <action android:name="android.intent.action.MAIN"/>
        <category android:name="android.intent.category.LAUNCHER"/>
    </intent-filter>
</activity>
<activity android:name=".Activity2">
</activity>
```

七. 思考与总结

这次实验难度明显比前两次难得多，花了比较多的时间，特别是页面的布局和 ListView 的适配器上，出了很多的 bug，不过最后都成功的解决了，只有遇到问题并成功解决才能对知识记得更深，通过这次实验，学习到了 RelativeLayout 的布局，ListView 的适配器设置和事件处理，页面跳转和传参等内容，付出的时间也有了收获。