

湖南科技大学计算机科学与工程学院

综合实践能力创新实训(2) 课程设计指导书

(17 级计算机、网络、信息安全、物联网、软件工程专业用)

2019 年 12 月

前 言

随着 Android 智能手机平台在市场上的广泛普及和移动计算技术的迅速发展，基于 Android 操作系统的增值业务开发变得越来越流行。当前移动互联网业务很快在当今社会占据了主导地位，WEB 游戏、电子商务、交易及购物、社交网站等层出不穷，可以说未来的社会是基于移动互联网的社会。

Google Android 操作系统是 Google 公司于 2007 年 11 月 5 日发布的基于 Linux 平台的开源手机操作系统，其特色在于拥有强大的用户界面设计，采用了优化图形显示技术及专用图标；从程序员的角度出发，向应用程序提供了完备的系统调用、进程管理与进程通信以及应用程序开发接口等。Android 操作系统在发布初期就强调在全球范围内公开源码以供用户免费使用，是首个为移动终端打造的真正开放和完整的移动软件。

随着 Android 操作系统在智能手机市场中的占有率不断提升，Google 公司不断升级优化 Android 操作系统，社会上对应 Android 应用系统的开发需求不断提升。

因此，本课程设计基于最流行的 Java 语言，对 Android 应用开发的相关经典应用的需求、设计、开发以及相关开发工具的使用进行相关课程设计，期望能为学生走上社会奠定较好的 Android 应用开发基础。

本课程设计指导书是应学院要求编写的，编写过程中得到了学院领导的支持、系部同事的帮助，在此表示衷心的感谢。由于编写时间仓促，本课程设计指导书中的不当之处，欢迎老师、同学们提出意见。

编者
2019 年 12 月

第一部分 课程设计的目的、内容及安排

一、课程设计目的

智能终端系统及应用开发课程设计是计算机相关专业的一个综合性实践教学环节,是学习完《智能终端系统及应用开发》课程后进行的一次全面的综合练习。其目的在于促进学生复习和巩固 Android 应用技术,加深对软件设计方法、软件设计技术和设计思想的理解,并能运用所学 Android 设计知识,通过本课程设计让学生进行简单 Android 应用软件的开发,掌握软件设计的方法和面向对象程序设计的基本技术,提高学生在软件开发方面的综合应用能力。

二、课程设计内容

本课程设计提供了 15 个项目。每位同学独立完成指定的一个项目,也可在征得指导老师同意下另行确定项目。所设计的系统包含 Android 数据库访问、网络、内容提供者、Activity 等基本功能。根据课程设计 2 周时间的安排,同学们宜选择适当大小的设计课题,并根据题目的基本需求,给出概念模型与逻辑模型,编写程序和写出详细的设计说明书。开发工具主要使用 Java,数据库推荐使用嵌入式设备中比较流行的 SQLite 数据库。

三、本课程设计的基本要求

首先,要求同学们先仔细阅读本指导书的第一部分。然后根据下面要求完成本课程设计的任务。

- 1、熟悉 Eclipse、SQLite 等常用 Android 开发工具。
- 2、熟悉一种语言。
- 3、设计并实现一个具体的 Android 应用系统。
 - (1) 选定的项目进行需求分析,写出需求说明书,并搜集、整理所需素材。
 - (2) 根据项目需求,进行数据库(如果需要数据库的话)的概要设计与详细设计。
 - (3) 根据需求说明进行项目的功能设计,画出每个界面的原型。
 - (4) 进行详细设计,实现每个模块的功能。
 - (5) 对项目进行部署与测试。
 - (6) 分析总结项目的创新点和存在的不足,提出优化思路。
 - (7) 设计书中采用的代码、截图仅作参考。

要求所设计的 APP 能正常运行,界面漂亮美观,至少能实现对应项目的等基本功能。按指导书的题目要求进行设计,对超出题目要求者,可以适当提高计分。

- 4、按课程设计指导书要求,提交实习报告。

- (1) 课程设计报告要求字迹清晰、版面整齐,篇幅至少 8 页,装订整齐(建议和双面打印)。
- (2) 报告应书写设计内容、设计方法、实现方法、心得体会等内容,绝对不能写成源程序清单。
- (3) 每位同学都必须提交课程设计报告,没有提交报告者总评成绩直接计为不及格。
- (4) 所交设计报告完全雷同,互相抄袭者,总评成绩直接计为不及格。

四、课程设计安排

- 1、课程设计时间:2019-2020 学年上学期第 15-16 周(17-18 周)

2、课程设计地点：逸夫楼专业实验室。详细安排另附表。

3、分组和选题

每人一组，从课程设计指导书第三部分给定的参考题目中任选一题，独立完成，经指导老师许可也可选择其它难度和工作量相当或更高的类似题目。学习委员将本班选题情况交指导老师。如果学生任务完成较好，超出题目要求，指导老师可适当提高学生成绩。每班在选题时尽量分散，同一个题目不准超过 3 人选。禁止直接从网上下载源码。

4、学生课程设计注意事项

15-16 周（17-18 周）均在实验室进行课程设计，不得无故缺勤，不允许迟到早退，不得闲聊和玩游戏、玩手机。屡教不改者，指导老师可直接记其成绩为不及格。

5、其他说明

- (1) 各班课程设计的过程管理与质量控制由各班指导老师全权负责。
- (2) 各班指导老师可对本次课程设计的任何题目酌情调整工作量及难度，也可另外布置题目，以更好地训练同学们运用技术解决实际问题的能力。
- (3) 鼓励利用所学的 SSH、SpringMVC 等 Java Web 开发框架设计 Android APP 后台。
- (4) 不管是原生 Android APP Web APP，还是 Hybrid APP，唯一要求就是：代码能打包成 APK，能够安装在模拟器和真机上，能正常运行演示。禁止直接拿浏览器来演示。

五、课程设计的考查

1、考查时间：分三个时间点检查：主要是检查题目的需求理解、数据库或网络接口、概要设计；主要模块测试；综合测试。

2、评分标准：由指导教师根据学生课程设计过程中的学习态度、完成任务的情况和课程设计报告的质量等综合打分。成绩评定实行优秀、良好、中等、及格和不及格五个等级。

3、考查内容：包括系统测试和课程设计报告两部分。

六、注意事项

安卓开发比较耗资源，且经常容易出问题，同学们需要有耐心，可以用 AVD 或者 Genymotion 模拟器，或者直接用自己的 Android 手机真机。建议同学们平时留意一下日常生活中所接触的各种 APP 的界面及其设计原理，平时多动动脑筋思考，学习现有的 APP 和代码，多开发程序，累计经验。

相关资源的下载网址：

图标资源	http://www.iconfont.cn/
JDK 1.8.0_45 64bit	http://pan.baidu.com/s/1hqEu0DY
JDK 1.8.0_45 32bit	http://pan.baidu.com/s/1sRU0m
adt-bundle-windows-x86_64-20130917	http://pan.baidu.com/s/1gdJMdZl
adt-bundle-windows-x86-20130917	http://pan.baidu.com/s/1dDJkkVZ
SQLite Browser	http://pan.baidu.com/s/1ntnfPPn
Genymotion	http://pan.baidu.com/s/1ntITUTb
Android Studio 等开发环境	http://tools.android-studio.org/

第二部分 Android 开发环境搭建

一、目的与要求

1. 掌握 Android 环境（Android Studio, Eclipse）的搭建。
2. 学会 Java、Eclipse、Android SDK 以及 ADT 的安装与环境变量设置。

二、实验内容

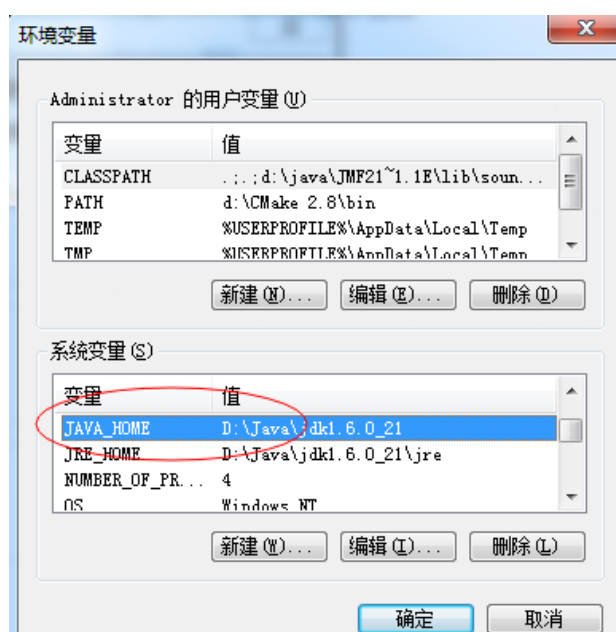
1. 安装 JDK6.0 以上版本。

(1) 访问以下网址：

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

下载 JDK。

(2) 配置环境变量：安装 JDK 以后，若需要在 cmd 下使用 Java 命令和编译、运行程序，可以配置环境变量：新建环境变量 JAVA_HOME，右击“我的电脑”——“属性”——“高级”——“环境变量”，如图所示：

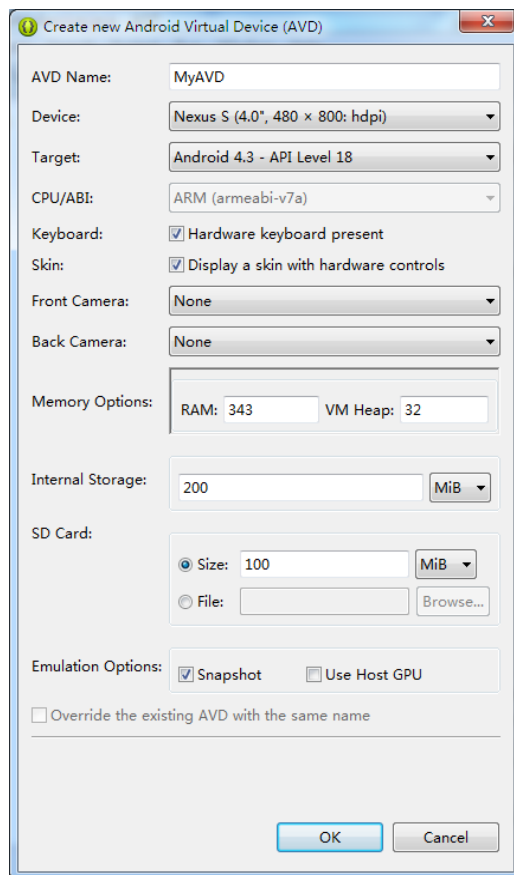
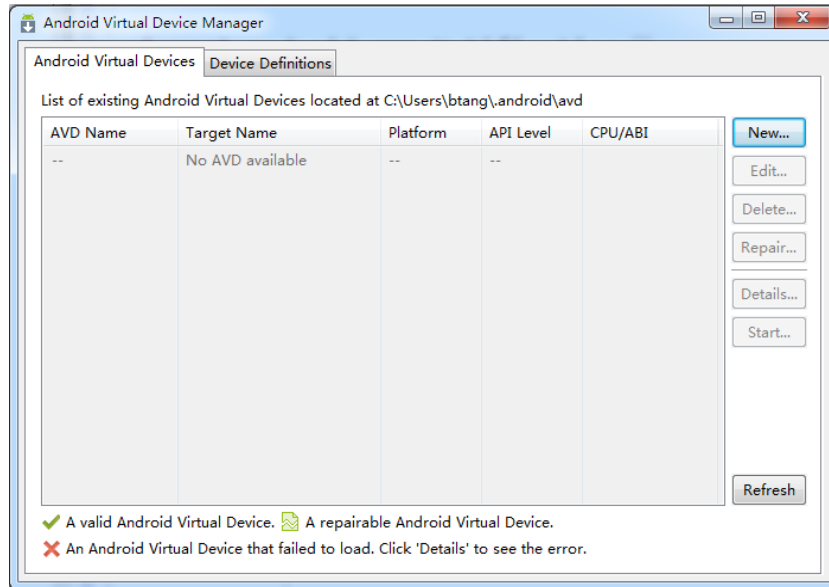


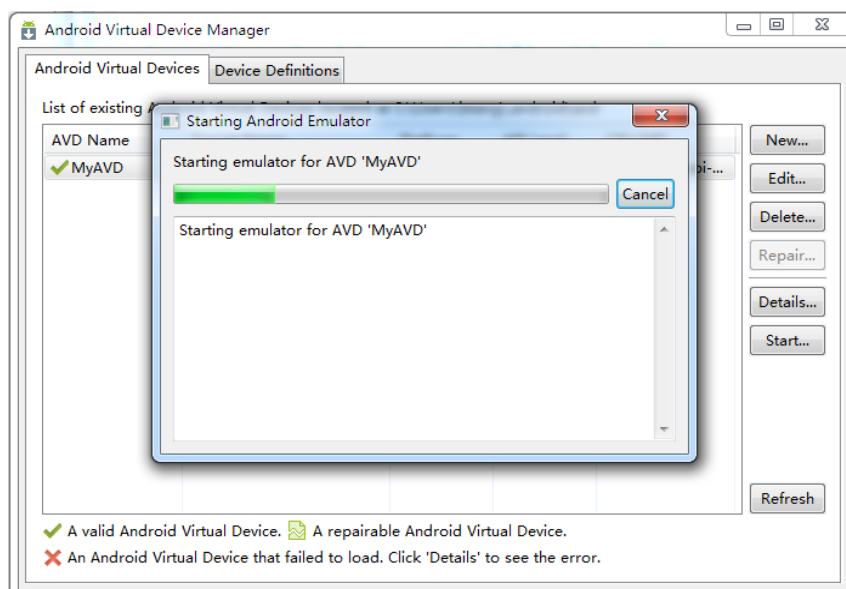
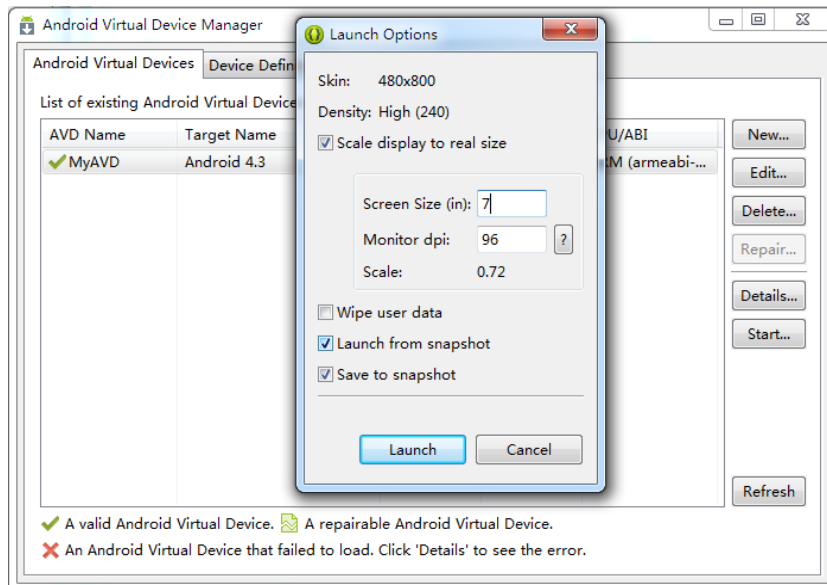
2. 安装 Eclipse。

将下载的 adt-bundle-windows-x86-20130917.zip 解压缩即可，进入 eclipse 文件夹，运行 eclipse.exe。

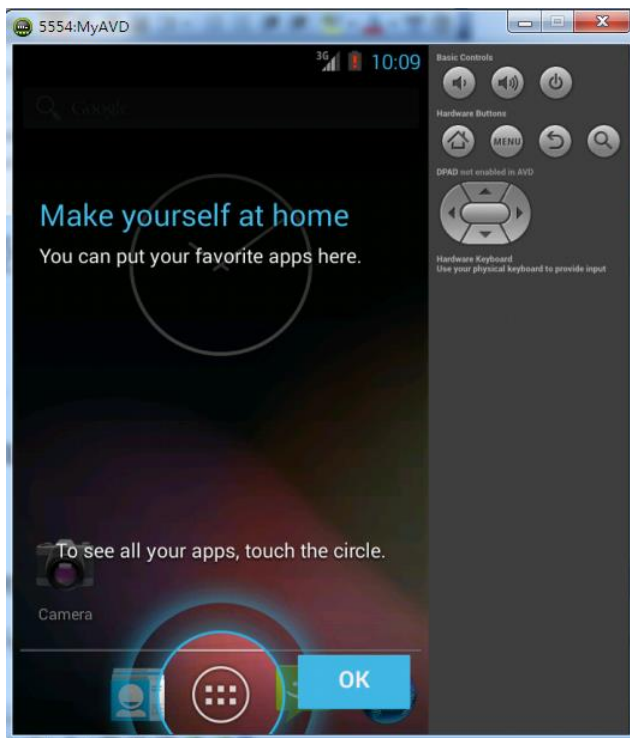
3. 配置 AVD(Android Virtual Device)。

在 Eclipse 集成开发环境中，在 Windows 菜单下，点击 Android Virtual Device Manager，然后点击 New，指定参数，创建一个模拟器，基本操作步骤如下图所示。





至此，整个环境已经安装和配置好，等待模拟器启动完毕（如下图所示），可进入下面的具体实验环节。

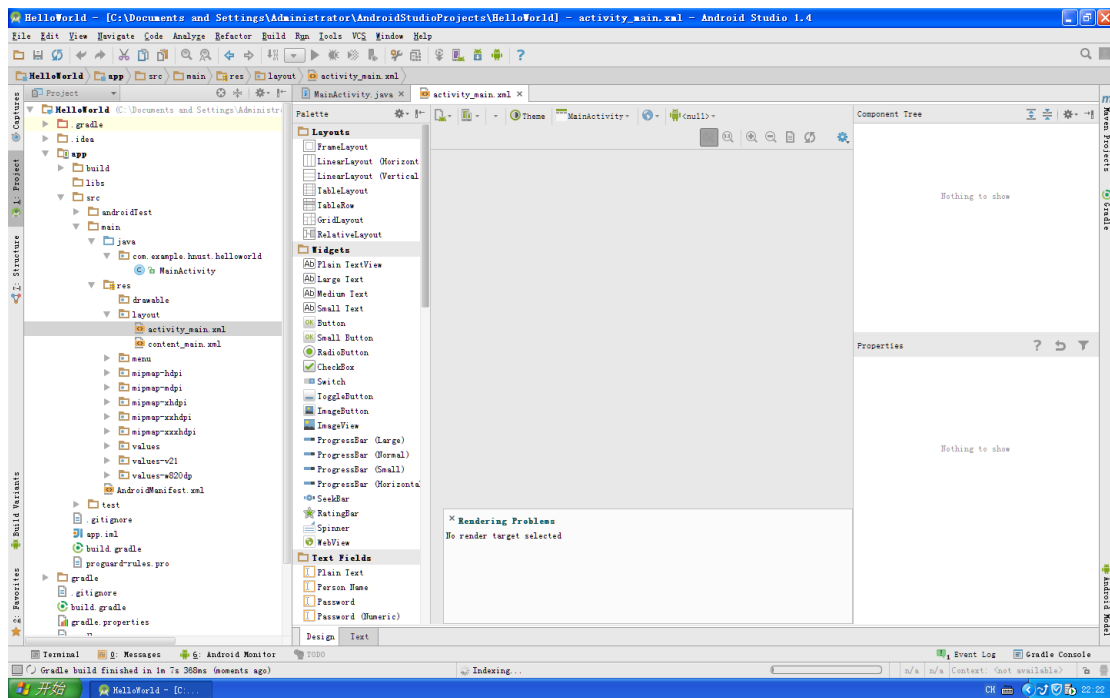
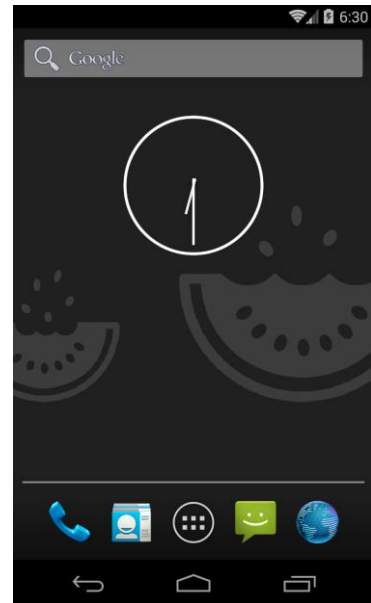
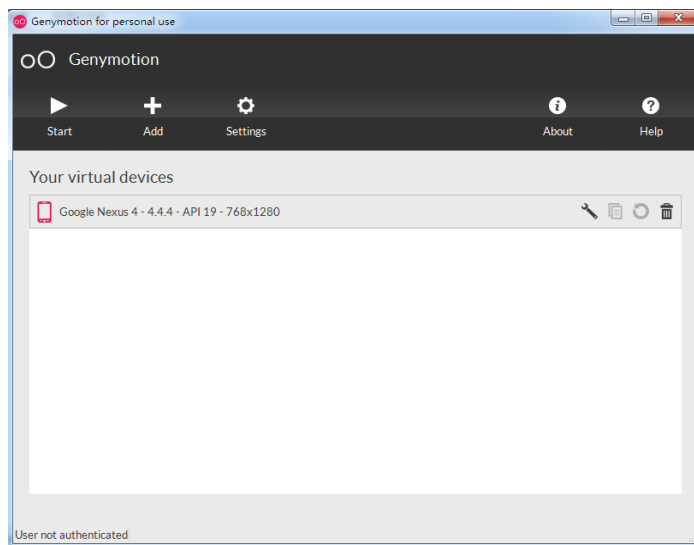


4. 其他的选择:

如果采用 Genymotion 模拟器，它是一个基于 VirtualBox 的 Android 设备模拟器，运行速度较快，首先在 <https://www.genymotion.com> 网站注册一个新的账号，选择 virtual devices，然后需要联网下载镜像。

现在 Google 官方推荐使用 Android Studio 进行开发。

下载网址 <http://tools.android-studio.org/>。请同学们根据硬件配置来选择使用 AS。



5. 创建项目的步骤

在 Eclipse 或者 Android Studio 中，利用向导一步步创建新项目。

New Android Application

The prefix 'com.example.' is meant as a placeholder and should not be used

Application Name: HelloHnust

Project Name: HelloHnust

Package Name: com.example.hellohnust

Minimum Required SDK: API 8: Android 2.2 (Froyo)

Target SDK: API 18: Android 4.3 (Jelly Bean)

Compile With: API 18: Android 4.3 (Jelly Bean)

Theme: Holo Light with Dark Action Bar

The application name is shown in the Play Store, as well as in the Manage Application list in Settings.

< Back Next > Finish Cancel

New Android Application

Create Activity

Select whether to create an activity, and if so, what kind of activity.

☒ Create Activity

Blank Activity

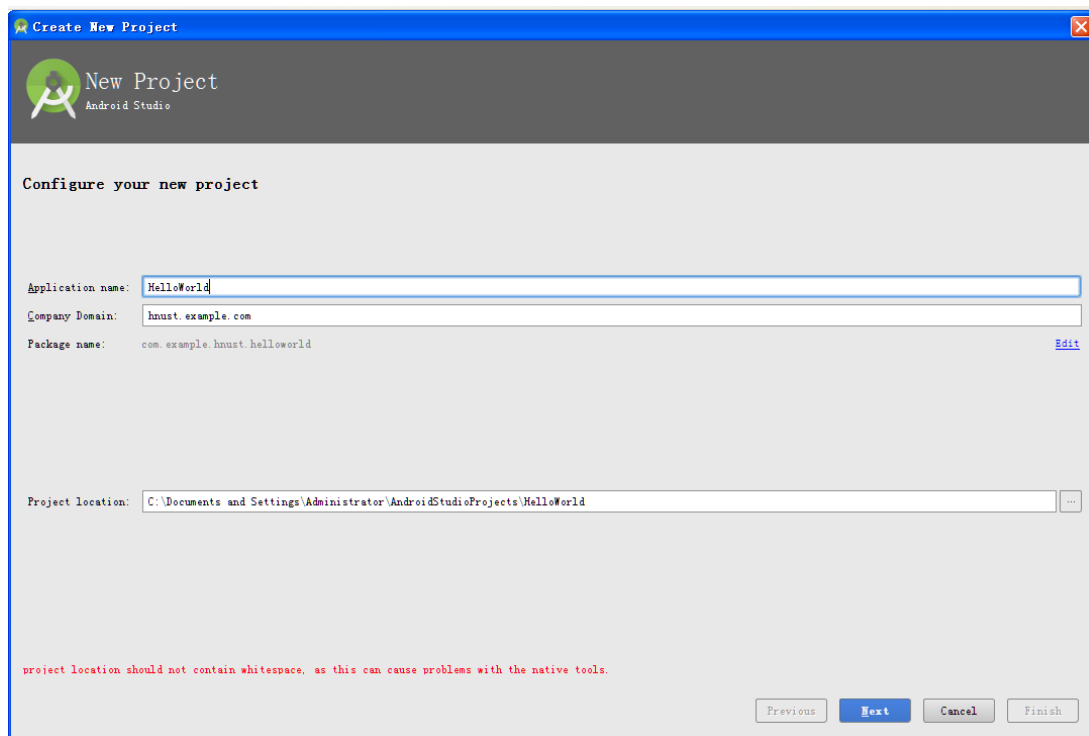
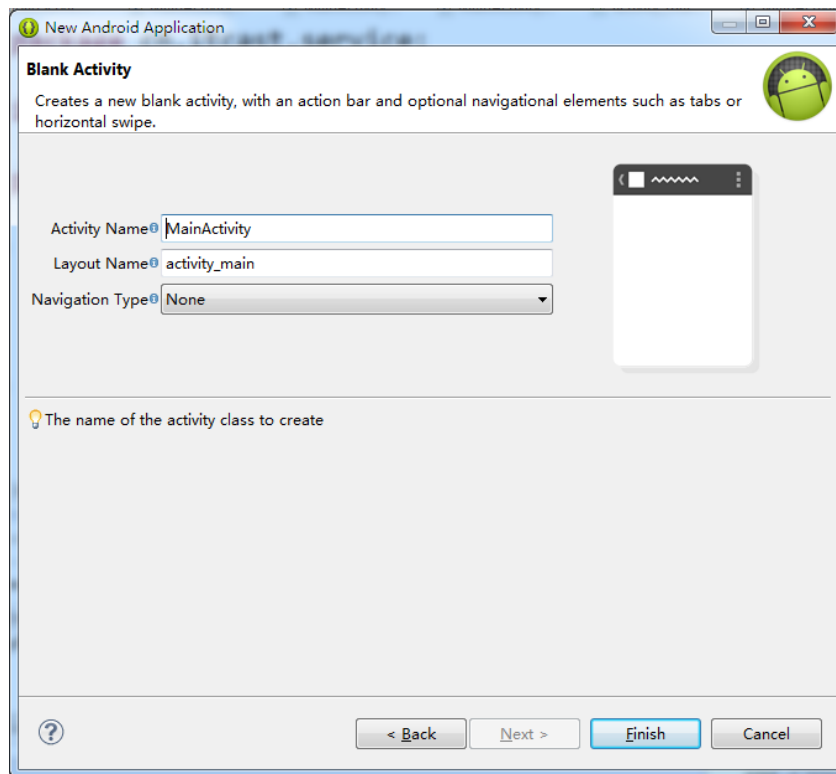
Fullscreen Activity

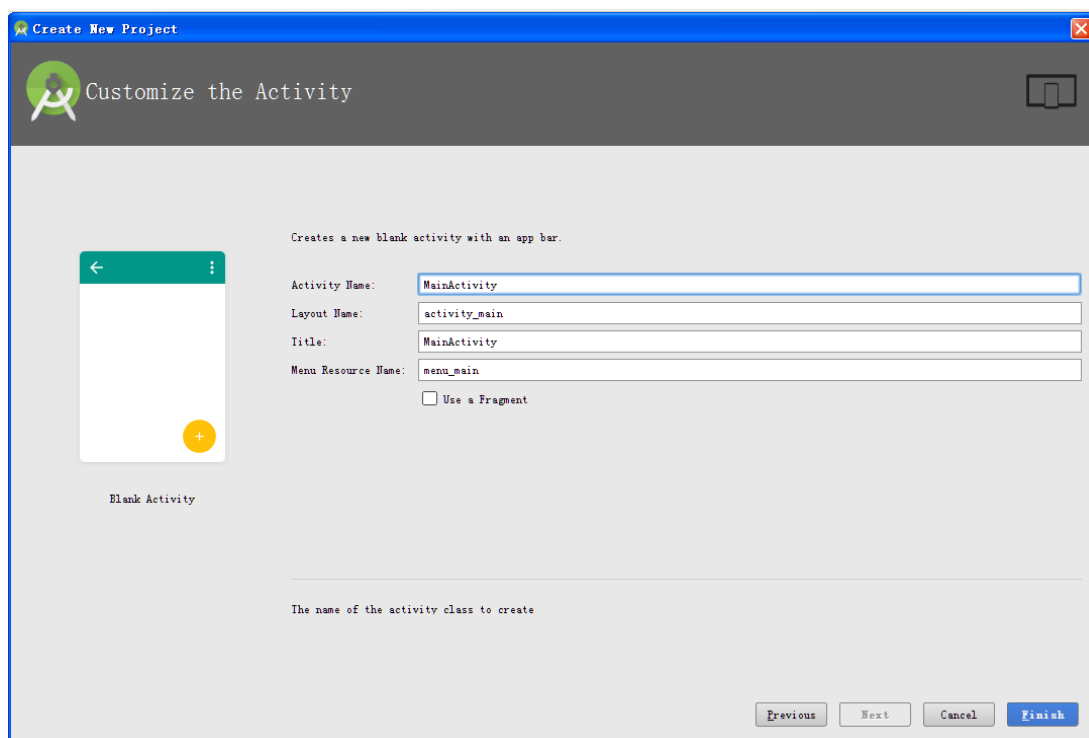
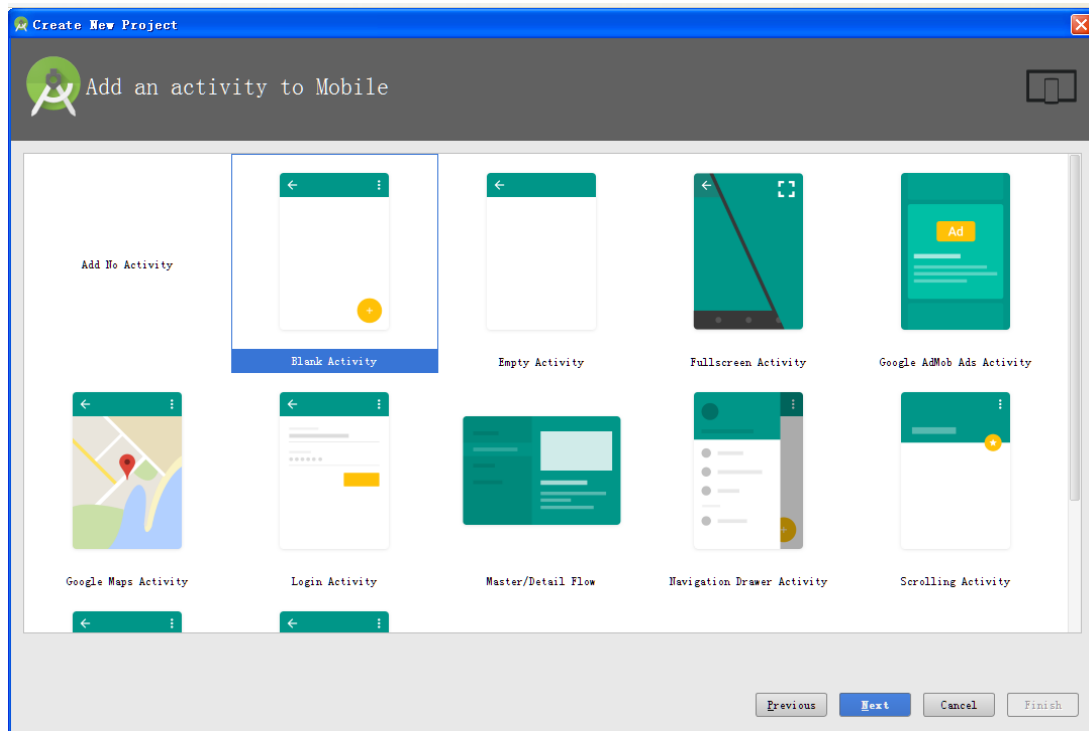
Master/Detail Flow

Blank Activity

Creates a new blank activity, with an action bar and optional navigational elements such as tabs or horizontal swipe.

< Back Next > Finish Cancel





第三部分 课程设计

题目一 手机通讯录开发

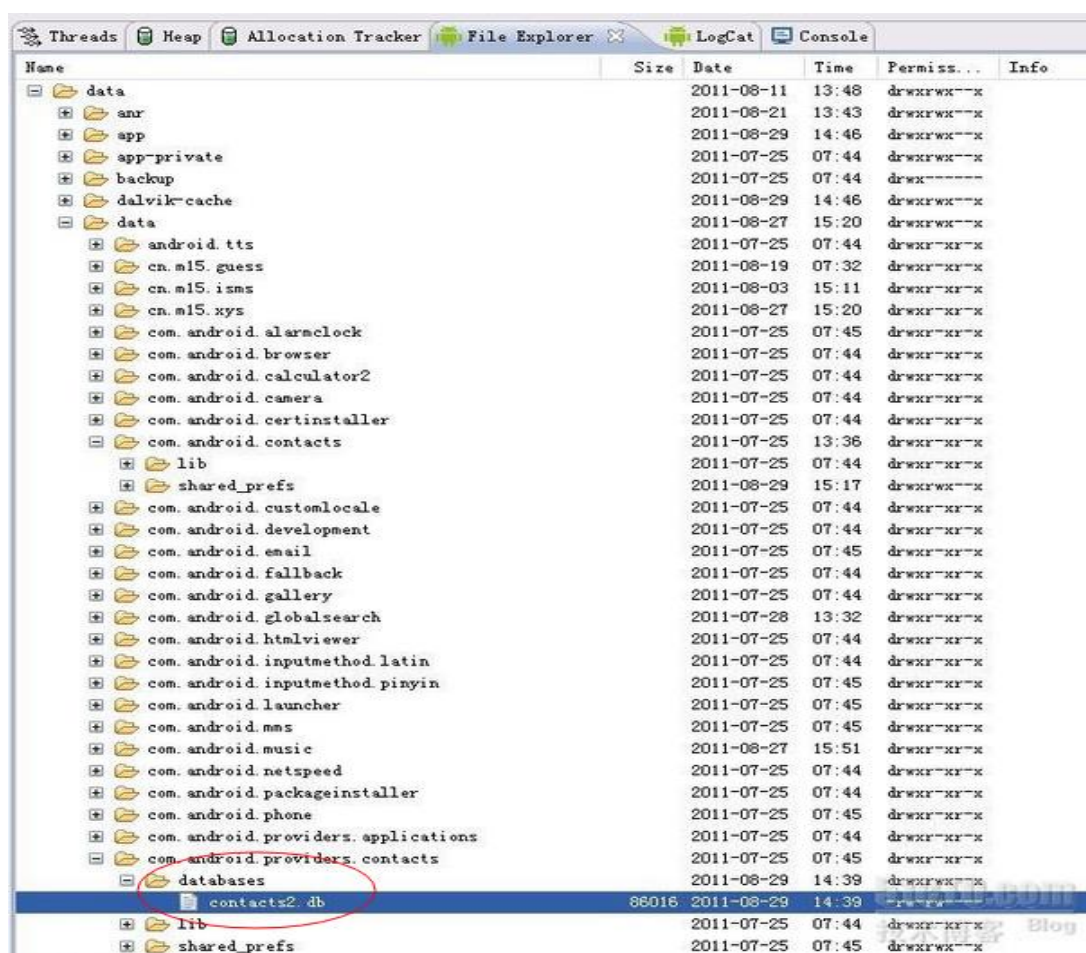
一、目的与要求

1. 掌握手机系统通讯录的结构。
2. 学会往系统通讯录中添加、删除、修改联系人。
3. 学会调用系统的拨打电话、发送短消息功能。

二、实验内容

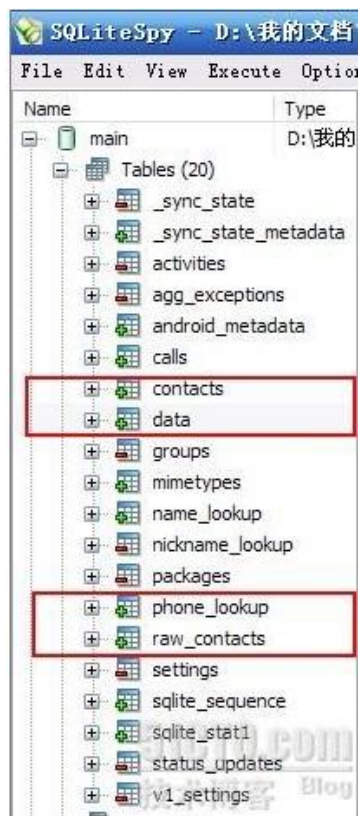
1) Contacts.db

Android 智能设备的联系人存储在 SQLite 数据库中, 名称为 contacts 或者 Contacts2.db; 如果须要获得通讯录联系人的信息就须要访问系统的数据库, 才能将信息拿出来; 真机需要 Root 权限才能查看到; 虚拟机存放在 /data/data/com.android.provider.contacts 目录下:



Name	Size	Date	Time	Permiss...	Info
data		2011-08-11	13:48	drwxrwx--x	
anr		2011-08-21	13:43	drwxrwx--x	
app		2011-08-29	14:46	drwxrwx--x	
app-private		2011-07-25	07:44	drwxrwx--x	
backup		2011-07-25	07:44	drwx-----	
dalvik-cache		2011-08-29	14:46	drwxrwx--x	
data		2011-08-27	15:20	drwxrwx--x	
android.tts		2011-07-25	07:44	drwxr--xr-x	
cn.m15.guess		2011-08-19	07:32	drwxr--xr-x	
cn.m15.isms		2011-08-03	15:11	drwxr--xr-x	
cn.m15.xys		2011-08-27	15:20	drwxr--xr-x	
com.android.alarmclock		2011-07-25	07:45	drwxr--xr-x	
com.android.browser		2011-07-25	07:44	drwxr--xr-x	
com.android.calculator2		2011-07-25	07:44	drwxr--xr-x	
com.android.camera		2011-07-25	07:44	drwxr--xr-x	
com.android.certinstaller		2011-07-25	07:44	drwxr--xr-x	
com.android.contacts		2011-07-25	13:36	drwxr--xr-x	
lib		2011-07-25	07:44	drwxr--xr-x	
shared_prefs		2011-08-29	15:17	drwxrwx--x	
com.android.customlocale		2011-07-25	07:44	drwxr--xr-x	
com.android.development		2011-07-25	07:44	drwxr--xr-x	
com.android.email		2011-07-25	07:45	drwxr--xr-x	
com.android.fallback		2011-07-25	07:44	drwxr--xr-x	
com.android.gallery		2011-07-25	07:44	drwxr--xr-x	
com.android.globalsearch		2011-07-28	13:32	drwxr--xr-x	
com.android.htmlviewer		2011-07-25	07:44	drwxr--xr-x	
com.android.inputmethod.latin		2011-07-25	07:44	drwxr--xr-x	
com.android.inputmethod.pinyin		2011-07-25	07:45	drwxr--xr-x	
com.android.launcher		2011-07-25	07:45	drwxr--xr-x	
com.android.mms		2011-07-25	07:45	drwxr--xr-x	
com.android.music		2011-08-27	15:51	drwxr--xr-x	
com.android.netspeed		2011-07-25	07:44	drwxr--xr-x	
com.android.packageinstaller		2011-07-25	07:44	drwxr--xr-x	
com.android.phone		2011-07-25	07:45	drwxr--xr-x	
com.android.providers.applications		2011-07-25	07:44	drwxr--xr-x	
com.android.providers.contacts		2011-07-25	07:45	drwxr--xr-x	
databases		2011-08-29	14:39	drwxrwx--x	
contacts2.db	86016	2011-08-29	14:39		
lib		2011-07-25	07:44	drwxr--xr-x	
shared_prefs		2011-07-25	07:45	drwxrwx--x	

打开 Contacts2.db 数据库, 里面有一堆表, 其中用到的主要是 4 张表 Contacts、Data、phone_lookup、raw_contacts 表;



➤ contacts 表

_id : 表的 ID, 关键字。

display_name: 联系人名称

photo_id:头像的 ID, 如果没有设置联系人头像, 该字段就为空

times_contacted:通话记录的次数

last_time_contacted: 最后的通话时间

lookup :是一个持久化的储存 因为用户可能会改名字, 但不能改 lookup 值

➤ data 表

raw_contact_id: 通过 raw_contact_id 可以找到 raw_contact 表中相对的数据;

data1 到 data15 这里保存着联系人的相关信息、联系人名称、联系人电话号码、电子邮件、备注等

mimetype_id:

vnd.android.cursor.item/photo -> 4 该行 data15 代表照片

vnd.android.cursor.item/phone_v2 -> 5 该行 data1 代表电话号码

vnd.android.cursor.item/name -> 6 说明该行 data1 代表联系人名称

➤ phone_look_up 表

data_id : 通过 data_id 可以找到 data 表中相对的数据。

raw_contact_id : 通过 raw_contact_id 可以找到 raw_contact_表中相对的数据。

normalized_number: 这个字段就比较有意思了, 它是将每个电话号码逆序排列

➤ raw_contact 表

_id	contact_id	account_name	account_type	sync1			
1	1	Yulei0619@gmail.com	com.google	http://www.google.com/m8/feeds/contacts/yulei0619@gmail.com/base2_property-android/1ac2a39e89c441cc			
2	2	Yulei0619@gmail.com	com.google	省略			
3	3	pcsc	com.htc.android.pcsc	省略			

contact_id: 引用 contacts 表的_id 字段

account_name: 指明该联系人是从哪个账户上同步下载下来的。

account_type: 邮件帐户的类型

sync1: 说明该联系人信息是与哪个网址进行交互（同步联系人信息）

2) 获取手机通讯录联系人信息

```

/**得到手机通讯录联系人信息**/
private void getPhoneContacts() {
    ContentResolver resolver = mContext.getContentResolver();
    // 获取手机联系人
    Cursor phoneCursor = resolver.query(Phone.CONTENT_URI, PHONES_PROJECTION, null,
    null, null);
    if (phoneCursor != null) {
        while (phoneCursor.moveToNext()) {
            //得到手机号码
            String phoneNumber = phoneCursor.getString(PHONES_NUMBER_INDEX);
            //当手机号码为空的或者为空字段 跳过当前循环
            if (TextUtils.isEmpty(phoneNumber))
                continue;
            //得到联系人名称
            String contactName = phoneCursor.getString(PHONES_DISPLAY_NAME_INDEX);
            //得到联系人 ID
            Long contactid = phoneCursor.getLong(PHONES_CONTACT_ID_INDEX);
            //得到联系人头像 ID
            Long photoid = phoneCursor.getLong(PHONES_PHOTO_ID_INDEX);
            //得到联系人头像 Bitmap
            Bitmap contactPhoto = null;
            //photoid 大于 0 表示联系人有头像 如果没有给此人设置头像则给他一个默认的
            if (photoid > 0) {
                Uri uri
                =ContentUris.withAppendedId(ContactsContract.Contacts.CONTENT_URI,contactid);

```

```

        InputStream input = ContactsContract.Contacts.openContactPhotoInputStream(resolver,
uri);
        contactPhoto = BitmapFactory.decodeStream(input);
    }else {
        contactPhoto = BitmapFactory.decodeResource(getResources(),
R.drawable.contact_photo);
    }
    mContactsName.add(contactName);
    mContactsNumber.add(phoneNumber);
    mContactsPhonto.add(contactPhoto);
}
phoneCursor.close();
}

    mContactsName.add(contactName);
    mContactsNumber.add(phoneNumber);
    mContactsPhonto.add(contactPhoto);
}

    phoneCursor.close();
}
}

```

3) 添加联系人

```

/**
 * 首先向 RawContacts.CONTENT_URI 执行一个空值插入，目的是获取系统返回的
rawContactId
 *
 * 这是后面插入 data 表的数据，只有执行空值插入，才能使插入的联系人在通讯录里
可见
 */
public void testInsert(){
    ContentValues values = new ContentValues();
    //首先向 RawContacts.CONTENT_URI 执行一个空值插入，目的是获取系统返回的
rawContactId
    Uri rawContactUri =
this.getContext().getContentResolver().insert(RawContacts.CONTENT_URI, values);
    long rawContactId = ContentUris.parseId(rawContactUri);
    //往 data 表入姓名数据
    values.clear();
    values.put(Data.RAW_CONTACT_ID, rawContactId);
    values.put(Data.MIMETYPE, StructuredName.CONTENT_ITEM_TYPE);
    values.put(StructuredName.GIVEN_NAME, "zhangsan");
    this.getContext().getContentResolver().insert(
android.provider.ContactsContract.Data.CONTENT_URI, values);
    //往 data 表入电话数据
    values.clear();
}

```



```

values.put(android.provider.ContactsContract.Contacts.Data.RAW_CONTACT_ID,
rawContactId);
    values.put(Data.MIMETYPE, Phone.CONTENT_ITEM_TYPE);
    values.put(Phone.NUMBER, "5554");
    values.put(Phone.TYPE, Phone.TYPE_MOBILE);
    this.getContext().getContentResolver().insert(
        android.provider.ContactsContract.Data.CONTENT_URI, values);
    //往 data 表入 Email 数据
    values.clear();
values.put(android.provider.ContactsContract.Contacts.Data.RAW_CONTACT_ID,
rawContactId);
    values.put(Data.MIMETYPE, Email.CONTENT_ITEM_TYPE);
    values.put(Email.DATA, "ljq218@126.com");
    values.put(Email.TYPE, Email.TYPE_WORK);
    this.getContext().getContentResolver().insert(
        android.provider.ContactsContract.Data.CONTENT_URI, values);
}

```

以上均需在 androidManifest.xml 中添加权限

```

<uses-permission android:name="android.permission.READ_CONTACTS"/>
<uses-permission android:name="android.permission.WRITE_CONTACTS"/>

```

4) 显示联系人列表

```

//      异步查询所有联系人
Uri uri = Uri.parse("content://com.android.contacts/data/phones");
String[] projection = { "_id", "display_name", "data1", "sort_key" };
asyncQuery.startQuery(0, null, uri, projection, null, null,
    "sort_key COLLATE LOCALIZED asc");

private class MyAsyncQueryHandler extends AsyncQueryHandler
{

    public MyAsyncQueryHandler(ContentResolver cr)
    {
        super(cr);
    }

    @Override
    protected void onQueryComplete(int token, Object cookie, Cursor cursor)
    {
        cursor.moveToFirst();
        Log.d("cccc",
            cursor.getString(0) + " 000 " + cursor.getString(1)
            + " 000 " + cursor.getString(2) + " 000 "

```

```

        + cursor.getString(3));

while (cursor.moveToNext())
{
    ContentValues cv = new ContentValues();
    cv.put(NAME, cursor.getString(1));
    cv.put(NUMBER, cursor.getString(2));
    cv.put(SORT_KEY, cursor.getString(3));
    list.add(cv);
}
if (list.size() > 0)
{
    setAdapter(list);
}
}

//List 值通过 ListAdapter 进行绑定
private void setAdapter(List<ContentValues> list)
{
    adapter = new ListAdapter(this, list);
    listView.setAdapter(adapter);
}

private class ListAdapter extends BaseAdapter
{

    private LayoutInflater inflater;

    private List<ContentValues> list;

    public ListAdapter(Context context, List<ContentValues> list)
    {
        this.inflater = LayoutInflater.from(context);
        this.list = list;
        alphaIndexer = new HashMap<String, Integer>();
        sections = new String[list.size()];

        for (int i = 0; i < list.size(); i++)
        {
            String currentStr = getAlpha(list.get(i).getAsString(SORT_KEY));
            String previewStr =
                (i - 1) >= 0 ? getAlpha(list.get(i - 1).getAsString(
                    SORT_KEY)) : " ";

```

```

        if (!previewStr.equals(currentStr))
        {
            String name = getAlpha(list.get(i).getAsString(SORT_KEY));
            alphaIndexer.put(name, i);
            sections[i] = name;
        }
    }
}

@Override
public int getCount()
{
    return list.size();
}

@Override
public Object getItem(int position)
{
    return list.get(position);
}

@Override
public long getItemId(int position)
{
    return position;
}

@Override
public View getView(int position, View convertView, ViewGroup parent)
{
    ViewHolder holder;

    if (convertView == null)
    {
        convertView = inflater.inflate(R.layout.list_item, null);
        holder = new ViewHolder();
        holder.alpha = (TextView) convertView.findViewById(R.id.alpha);
        holder.name = (TextView) convertView.findViewById(R.id.name);
        holder.number =
            (TextView) convertView.findViewById(R.id.number);
        convertView.setTag(holder);
    }
    else
    {

```

```

        holder = (ViewHolder) convertView.getTag();
    }

    ContentValues cv = list.get(position);
    holder.name.setText(cv.getAsString(NAME));
    holder.number.setText(cv.getAsString(NUMBER));
    String currentStr =
        getAlpha(list.get(position).getAsString(SORT_KEY));
    String previewStr =
        (position - 1) >= 0 ? getAlpha(list.get(position - 1)
            .getAsString(SORT_KEY)) : " ";

    if (!previewStr.equals(currentStr))
    {
        holder.alpha.setVisibility(View.VISIBLE);
        holder.alpha.setText(currentStr);
    }
    else
    {
        holder.alpha.setVisibility(View.GONE);
    }

    return convertView;
}

private class ViewHolder
{
    TextView alpha;

    TextView name;

    TextView number;
}
}

```

5) 拨打电话

<!--添加可以向外拨打电话的权限 -->

```
<uses-permission android:name="android.permission.CALL_PHONE"></uses-permission>
```

//点击 List, 获取 List 返回的 object-ContentValues

```
String number = ContentValues .getAsString(NUMBER);
```

//用 intent 启动拨打电话

```
Intent intent = new Intent(Intent.ACTION_CALL,Uri.parse("tel:"+number));
```

```
startActivity(intent);
```

6) 发送短消息

第一：调用系统短信接口直接发送短信；主要代码如下：

```
/** * 直接调用短信接口发短信  
  
 * @param phoneNumber  
  
 * @param message */  
  
public void sendSMS(String phoneNumber,String message){  
  
    //获取短信管理器  
  
    android.telephony.SmsManager smsManager =  
    android.telephony.SmsManager.getDefault();  
  
    //拆分短信内容（手机短信长度限制）  
  
    List<String> divideContents = smsManager.divideMessage(message);  
  
    for (String text : divideContents) {  
  
        smsManager.sendTextMessage(phoneNumber, null, text, sentPI, deliverPI);  
  
    }  
  
}
```

第二：调起系统发短信功能；主要代码如下：

```
/** * 调起系统发短信功能  
  
 * @param phoneNumber  
  
 * @param message */  
  
public void doSendSMSTo(String phoneNumber,String message){  
  
    if(PhoneNumberUtils.isGlobalPhoneNumber(phoneNumber)){  
  
        Intent intent = new Intent(Intent.ACTION_SENDTO,  
        Uri.parse("smsto:"+phoneNumber));
```

```

        intent.putExtra("sms_body", message);

        startActivity(intent);

    }

}

//增加发送短消息的权限

<uses-permission android:name="android.permission.SEND_SMS" />

```

题目二 手机网络聊天

一、目的与要求

1. 掌握基于 Android 的网络编程的相关知识。
2. 学会如何定义网络协议。
3. 学会如何传输表情、图片等特殊信息。

二、实验内容

1. 建立简单的 SOCKET 通信服务度和客户端。

服务端

```

serverAddress =
    InetAddress.getByName("192.168.1.142");
socket = new Socket(serverAddress, 500);
commsThread = new CommsThread(socket);

public class CommsThread extends Thread {
    private final Socket socket;
    private final InputStream inputStream;
    private final OutputStream outputStream;

    public CommsThread(Socket sock) {
        socket = sock;
        InputStream tmpIn = null;
        OutputStream tmpOut = null;
        try {
            ///---creates the inputstream and outputstream objects
            // for reading and writing through the sockets---
            tmpIn = socket.getInputStream();
            tmpOut = socket.getOutputStream();
        } catch (IOException e) {

```

```

        Log.d("SocketChat", e.getLocalizedMessage());
    }
    inputStream = tmpIn;
    outputStream = tmpOut;
}

public void run() {
    ///---buffer store for the stream---
    byte[] buffer = new byte[1024];
    ///---bytes returned from read()---
    int bytes;
    ///---keep listening to the InputStream until an
    // exception occurs---
    while (true) {
        try {
            ///---read from the inputStream---
            bytes = inputStream.read(buffer);

            ///---update the main activity UI---
            SocketsActivity.UIUpdater.obtainMessage(
                0,bytes, -1, buffer).sendToTarget();
        } catch (IOException e) {
            break;
        }
    }
}

///---call this from the main activity to
// send data to the remote device---
public void write(byte[] bytes) {
    try {
        outputStream.write(bytes);
    } catch (IOException e) { }
}

///---call this from the main activity to
// shutdown the connection---
public void cancel() {
    try {
        socket.close();
    } catch (IOException e) { }
}
}

```

注意在 androidManifest 中添加用户权限:

```
<uses-permission android:name="android.permission.INTERNET" />
```

客户端

```
serverAddress = InetAddress.getByName("192.168.1.142");
socket = new Socket(serverAddress, serverport);
Socket.connect(SocketAddress remoteAddress);
InputStream tmpIn = socket.getInputStream();
OutputStream tmpOut = socket.getOutputStream();

While(bRunning){
    tmpIn.read(byte[] buf,int offset,int length);
    tmpOut.write(byte[] buffer,offset,int length);
}
```

2. 定义聊天的网络交互协议。

当前用于社交的网络聊天协议主要采用 XMPP(可扩展消息处理与出席协议)、SIMPLE(即时通信和出席平衡扩充的进程开始协议);Google 主推是 XMPP 的聊天协议;可以参考其建立聊天协议;

```
<stream>
  <presence>
    <status/>
  </presence>
  <message>
    <body/>
  </message>
  <iq>
    <bind/>
  </iq>
</stream>
//用户状态信息
<presence>
  from=cyber1@jabber.org/contact
  to=cyber2@jabber.org/contact>
  <status>online</status>
</presence>
//聊天信息
<message>
  from=cyber1@jabber.org/contact
  to=cyber2@jabber.org/contact
  Type= "chat" >
  <body>Hello</body>
```


</message>

//IQ 信息，用于与对方进行信息协商，比如采用什么协议加密等等

<iq

from=cyber1@jabber.org/contact

Id= "id1" Type= "result" >

</iq>

iq 的主要属性: type, 包括:

Get: 获取当前域值

Set: 设置或替换 get 查询的值

Result: 说明成功的响应了先前的查询

Error: 查询和响应中出现的错误

以下是完整的登录协议报文示例:

1. Client 初始流给 server

<stream:stream to="example.com"

xmlns="jabber:client" xmlns:stream="http://etherx.jabber.org/streams" version="1.0" >

2. Server 使用一个流标记作为响应发给 client

<stream:stream from='cyber' xmlns='jabber:client' xmlns:stream='http://etherx.jabber.org/streams'

id='a7747277-ad62-4813-b739-63504d6e1246' version='1.0' xml:lang='zh-cn' >

3. Server 发送 STARTTLS 扩展给 client，并带有认证机制与任何其他流特征

<stream:features xmlns="http://etherx.jabber.org/streams">

<starttls xmlns="urn:ietf:params:xml:ns:xmpp-tls" />

<mechanisms xmlns="urn:ietf:params:xml:ns:xmpp-sasl">

<mechanism>PLAIN</mechanism>

<mechanism>DIGEST-MD5</mechanism>

</mechanisms>

<compression xmlns="http://jabber.org/features/compress">

<method>zlib</method>

</compression>

<auth xmlns="http://jabber.org/features/iq-auth" />

<register xmlns="http://jabber.org/features/iq-register" />

</stream:features>

4. Client 选择一个认证机制

<auth mechanism="DIGEST-MD5" xmlns="urn:ietf:params:xml:ns:xmpp-sasl"></auth>

5. 服务端发送挑战

<challenge

xmlns="urn:ietf:params:xml:ns:xmpp-sasl">cmVhbG09Indpbi0yMDE0MDMxNnFudyIsbm9uY2

U9InVUTHhPNmRiSy9Sa1owUUo0ZCspeXJyMTdsOTIJafZqS0dXeDE1QXEiLHFvcD0iYXV

0aCIsY2hhcnNldD1ldGYtOCxhbGdvcml0aG09bWQ1LXNlc3M=</challenge>

6. 客户端给出应答

```
<response
xmlns="urn:ietf:params:xml:ns:xmpp-sasl">Y2hhcnNldD11dGYtOCx1c2VybmFtZT0idGVzdGUi
LHJIYWxtPSJ3aW4tMjAxNDZxMTZxYXNldG5vbmlNPSJ1VEx4TzZkYksvUmtaMFFKNGQrL
3lycjE3bDk5SWWhWaktHV3gxNUFxiYz0wMDAwMDAwMSxjb9uY2U9IjNLSmRHRIRoe
VZnN2ptRm5JbVFjZlVJY0lRUC9IZng5WDIyMFJXNm4iLGRpZ2VzdC11cmk9InhtcHAvd2luL
TIwMTQwMzE2cW53IixtYXhidWY9NjU1MzYscmVzcG9uc2U9Y2JjZDkzYmM3ZTA0NzFk
MGNmZjRmNjY2VhZjdjOTMscW9wPWF1dGg=</response>
```

7. 服务度检查通过

```
<success
xmlns="urn:ietf:params:xml:ns:xmpp-sasl">cnNwYXV0aD1mMjIzYTc1MjhmZTgzNjhkMGUzZ
jQ4MTI5YzE3YWM5Yw==</success>
```

登录完成后，进行前面类似的 Presence、Message、IQ 报文进行信息交互；

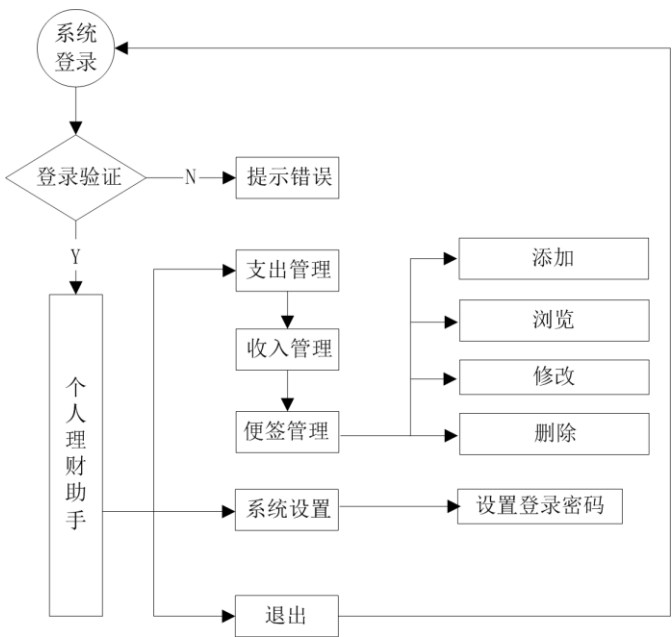
题目三 个人理财小助手

一、目的与要求

- 1. 掌握 SQLite 数据库及其使用。
- 2. 熟练掌握布局及常用控件 Button、ListView、EditText、TextView 等。

二、功能需求

个人理财小助手主要为用户提供 5 个主要模块，包括：登录模块、收入管理模块、支出管理模块、便签管理模块、系统设置模块。对日常的开支进行随时记录，存储到本机的 SQLite 数据库。



1. 使用 SQLite 数据库实现增加、删除、修改；
2. 使用 ListView 进行显示；
3. 增加私密性，验证成功才能进行收支管理；
4. 能提供简单的收支统计（按月、按周）功能，丰富的图标统计（饼图、柱状图、折线图）；
5. 提供数据管理，能对数据进行导出（如将收入/支出明细导出为 txt 或 Excel 文档或云端备份）。

(以下截图仅供参考，请自行发挥)





题目四 幼儿识图 APP

一、目的与要求

1. 掌握 Android 中的菜单。
2. 学会如何实现图片、语音等多媒体应用。

二、实验内容

设计一款为儿童设计的认知应用，帮助学说话，提供丰富的图片、声音资源，配合中英文发音，以及趣味测试题目。

功能包括：

1. 帮助小朋友认识数字、动物、交通工具、水果、颜色、形状这六大类的图片，在设置菜单中进行设置；
2. 为每个图片准备声音文件，在设置菜单中可以设置中/英语发声；提供两个按钮进行上一张和下一张图片切换，同时实现左右滑动进行切换；提供发声的按钮，点击可以语音复读；自动播放（进入下一张图）功能，播放间隔时间可以根据需要设置，可以停止自动播放；
3. 测试的功能，在菜单中选择进入游戏测试模式，随机抽取 4 张图片对幼儿进行测试训练。选择与描述相符的图片，自动播放语音“下面哪幅图是……”，在小孩做出选择之后，判断答案对错，并给出相应的语音提示，然后进入下一题。

(以下截图仅供参考，请参阅教材中声音播放相关资料)



题目五 新闻客户端

一、目的与要求

1. 掌握 Android 中的菜单及导航框架。
2. 掌握自定义布局。
3. 掌握 Android 中的数据存储。

二、功能要求

1. 要求实现体育、财经、娱乐、科技等多个新闻版块（或者自定义），并可以自由通过导航菜单切换。
2. 对每条新闻可以打开，进行阅读详细信息，包含丰富的图片和文字。
3. 对每条新闻可以进行点赞和评论。
4. 新闻数据： 如果动手能力较强，可以尝试自己动手用 PHP、JSP 等搭后台，利用 Android 网络编程，或者通过网络爬虫，爬取相关的新闻素材。（特别提示：通过安卓模拟器访问 PC 搭建的网站，IP 地址请使用 10.0.2.2）

(以下截图仅供参考)



题目六 手机防火墙

一、目的与要求

1. 学会如何使用广播接收者获取 Android 系统广播。
2. 学会如何进行数据保存。

二、功能需求

手机防火墙能够监听手机的来电与短信，对电话、短信进行拦截，按多种规则进行处理，同时可以配合黑白名单，进行拒绝或自动回复等。主要的功能：

1. 短信防火墙：短信防火墙包含按号码过滤和按内容过滤两种方式。通讯录中的白名单，即不受约束的名单，不论其发任何内容的短信，均不加以阻拦。与之对应的黑名单，需要用户另外编辑维护，从黑名单号码发来的短信，不论是什么内容，一律加以阻拦。若不在此二者之列，则启动内容阻拦规则。用户可事先设定阻拦内容，加以阻拦，否则作为正常短信予以通过。规则数量不限，黑白名单内的数量也不限，被阻拦的短信可以在“垃圾箱”这项功能中查看最近的被删除的短信。

2. 电话防火墙：目标手机接收到来电通知后，查询黑名单列表来决定是挂断还是振铃，可以减少手机对您的辐射，屏蔽掉骚扰垃圾电话。被阻拦的来电时间、来电者号码被记录下来，也可以被查看。

题目七 网上商城/外卖小助手

一、目的与要求

1. 掌握 Android 中的菜单及导航框架。
2. 掌握自定义布局。

3. 掌握 Android 中的数据存储。

二、实验内容

1. 要求实现商品展示、商品详细介绍、下订单、购物车。
2. 要求实现用户注册、登录、查看历时订单。
3. 数据：可以采用静态的固定的数据来模拟（如果动手能力较强，可以尝试自己动手搭后台，利用 Android 网络编程）。

(以下截图仅供参考)



题目八 学习助理（查成绩、查课表）

一、目的与要求

1. 学会如何使用 Activity 及常见控件。
2. 学会如何进行数据保存（SQLite）。

二、功能需求

学习助理 APP 主要的功能有：

1. 菜单导航：查成绩和查课表两种功能选择菜单，及学期的切换。
2. 查成绩：给定不同的条件，提供多种查询，如：查找指定班级成绩及排名、查找指定学生各门课成绩、查找指定课程的成绩单。
3. 查课表：给定不同的条件，提供多种查询，如：查找指定周次课表、查找指定日期课表、查找指定课程课表。
4. 技术挑战：成绩及课表数据通过网络接口（爬虫）来访问获取。

题目九 音乐播放器

一、目的与要求

1. 学会如何使用 MediaPlayer 等媒体播放组件。
2. 学会如何进行数据保存。

二、功能需求

音乐播放器主要的功能：

1. 播放列表：在设置菜单，读取 sdcard 指定目录下的所有 mp3 文件，建立播放列表。
2. 播放控制：能够对播放进行控制，能够开始/继续播放、暂停播放、停止播放、上一首、下一首。
3. 保存播放记录：能够存储已播放过的历史。
4. 扩展功能：（1）增加歌曲库的概念；（2）给歌曲设置类别标签，根据已播放的历史，在界面上的推荐区域展示推荐歌曲；（3）根据歌名，搜索下载网上的歌曲。

题目十 图书管理

一、目的与要求

1. 学会如何使用 Activity 及常见控件。
2. 学会如何进行 SQLite 数据保存。

二、功能需求

为了简单管理自己的个人图书外借记录，现要求开发一个单机版的图书管理 APP，主要的功能：

1. 图书入库：输入图书名、作者、出版社、单价、出版日期、书号 ISBN 等基本信息，存入 SQLite 数据库。
2. 图书信息维护：当输错图书信息时，提供修改功能；删除图书。
3. 扫码借出：扫图书的 ISBN 码，同时输入借书人信息，自动获取当前日期，产生一条借阅信息。

4. 图书归还：扫图书的 ISBN 码，点击确认，归还图书。
 5. 查询功能：给定条件（图书名）查询图书；查询当前已借出的图书；查询当前所有的图书；查询历史借阅记录。
- 注：可采用 ZXing 等实现扫码功能。

题目十一 儿童安全定位

一、目的与要求

1. 学会如何使用 Activity 及常见控件。
2. 学会使用地图及定位。
3. 学会如何通过网络编程访问后台服务接口。

二、功能需求

本题目要求设计两个 APP 来完成此项目。一个 APP 用于模拟儿童端（以下简称为 APP_CHILDREN），一个 APP 用于父母来监控小孩的位置（以下简称为 APP_PARENT）。

具体的功能有：

1. 位置汇报：APP_CHILDREN 开机自启动，每隔 3 分钟定时向后台服务器汇报经纬度等位置信息（采用 GPS 或网络定位）。
 2. 位置查询：APP_PARENT 从后台服务器获取小孩的位置信息，提供两种查询，（1）查询当前的位置（最近的一次位置汇报）；（2）查询给定时间区间内的小孩历史轨迹，并在百度地图中显示轨迹。
 3. 一键直呼：APP_PARENT 设置并存储小孩的手机号（采用 SharedPreferences 存储），按钮一键拨打小孩手机。与此相似，APP_CHILDREN 设置父亲和母亲的手机号，按钮一键拨打。
 4. 位置报警：APP_CHILDREN 15 分钟没有位置汇报（手机异常关机或者无网络），APP_PARENT 端弹出警报。
 5. 搭建后台服务器及相应的数据库，并做好位置汇报和位置查询的接口。
- 注：请自行研究百度定位及百度地图。

题目十二 校车呼叫系统

一、目的与要求

1. 学会如何使用 Activity 及常见控件。
2. 学会使用地图及定位。
3. 学会如何通过网络编程访问后台服务接口。

二、功能需求

针对校园里小黄车和小绿车的现状，想坐车，但是车子没来，让司机及时知道哪些位置有学生打车需求。借鉴滴滴打车软件，学生发送打车请求，司机查看。本题目要求设计两个 APP 来完成此项目。一个 APP 用于司机端（以下简称为 APP_DRIVER），一个 APP 用于学生端（以下简称为 APP_STU）。

具体的功能有：

1. 位置汇报：APP_DRIVER 每隔 1 分钟定时向后台服务器汇报经纬度等位置信息（采用 GPS 定位）。
 2. 位置查询：APP_STU 从后台服务器获取所有校车当前的位置（最近的一次位置汇报），并在地图上显示所有校车的位置，直观的展示校车当前的分布情况。
 3. 发送打车请求：APP_STU 发送打车请求，获取学生当前的位置作为起点，通过编辑框输入或者下拉列表选择终点，发送至后台服务器。
 4. 打车请求分布图：APP_DRIVER 可以在地图上直观的展示发送请求的学生位置热力图（颜色越深表示此区域当前有较多的学生需要坐车），从而调度车辆，将车子开过来。
 5. 搭建后台服务器及相应的数据库，并做好位置汇报和位置查询等的接口。
- 注：请自行研究百度定位及百度地图。

题目十三 人脸识别考勤

一、目的与要求

1. 学会如何使用Activity及常见控件。
2. 学会使用数据库存储及网络通信。
3. 学会网上提供的现成SDK。

二、功能需求

针对课堂上采用APP来进行考勤这一需求，开发一套系统。

具体的功能有：

1. 人脸录入：通过拍摄学生照片，或者上传现有照片。
2. 人脸识别考勤：通过人脸识别，每个学生来考勤。
3. 考勤统计及数据查询：制定查询条件（按学生、按时间），输出统计结果。

注：请自行研究人脸识别API及各种接口。

题目十四 运动计步及轨迹绘制

一、目的与要求

1. 学会如何使用Activity及常见控件。
2. 学会使用Android手机中的加速度等多种传感器。
3. 学会如何通过网络编程访问后台服务接口。

二、功能需求

设计一个运动计步APP，具体的功能有：

1. 计步开始与计步结束：在计步过程中采集数据。
2. 轨迹保存与绘制：对运行的步数进行估算，利用经纬度等数据对轨迹进行绘制。
3. 云端保存：同步到云端服务器，运动计步数据及轨迹的保存。

注：请自行研究百度定位及百度地图（或高德地图）。

题目十五 股票行情 APP

一、目的与要求

1. 学会如何使用Activity及常见控件。
2. 学会如何通过网络编程访问后台服务接口。

二、功能需求

利用相关 API 或者通过网络爬虫，爬取相关的股票行情数据。对股票数据进行绘制，提供股票查询、数据对比等多种数据可视化的方式。

附录： 课程设计报告封面

湖南科技大学计算机科学与工程学院

综合实践能力创新实训(2)

课程设计报告

题目： _____

学 号： _____

班 级： _____

姓 名： _____

指导老师： _____