# 课程设计——Android四大组件综合使用

——吕锐+201906061609

## 一、基础实验——Android四大组件综合使用

### （一）实验目的

1. 熟悉Android较复杂界面设计；
2. 掌握Android高级控件的使用方法；
3. 掌握Activity布局运用、控件事件编写；
4. 掌握Android四大组件综合使用技术。

### （二）基本知识与原理

1. Android应用采用MVC界面设计模式，通过布局文件、资源文件与程序逻辑代码的分离，实现界面设计与逻辑开发的并行进行。
2. 分析播放器APP界面交互特点，选择合理的第三方控件和布局方式，设计并实现通讯录APP功能。

### （三）具体要求

1、选题要求：APP选题可根据自己兴趣进行选题，独立完成。确定题目前，建议跟老师确认，以保证所选题目是否满足课程作业要求。

2、技术要求：所开发的APP应采用原生Android开发技术，应用的技术应至少涵盖Activity、Service、SQLite、Broadcast Receiver、Handler消息机制、GPS定位（地图接入）、第三方控件中的四种技术。

3、界面设计要求：所完成的APP界面应美观、操作便捷、具有屏幕适配性。

4、文档要求：提交的文档应包含需求分析、界面设计、关键技术、完成效果展示等内容。要求文档格式规范、排版美观、内容饱满。

5、验收要求：一对一验收过程中，应如实回答老师的验收提问，对于弄虚作假的行为，情节严重，成绩将做作弊处理。

6、评分标准：

优秀：APP独立开发，具有原创性，技术应用得当、功能完整，具有一定创新性，文档撰写内容详实，格式标准。验收过程回答正确。

良好：APP独立完成，具有原创性，技术应用得当、功能较为完整，文档撰写内容详实，格式标准。验收过程回答正确。

中等：APP独立完成或部分参考已有资料，技术应用得当、功能较完备，但必须体现工作量，文档撰写内容合格，验收过程回答基本正确。

及格：APP参考已有资料，但不允许完全抄袭，需对参考内容做一定修改，文档撰写内容合格，验收过程回答基本正确。

不及格：APP参考较多，文档撰写不合格，验收过程回答错误。

### （四）实验内容及步骤（以包含需求分析、界面设计、关键技术分析）

#### 需求分析

**我在大学生活遇到的问题**

在大学生活中，不少同学因为没有了体育课的监督和自由散漫的寝室生活，以及学业的压力，运动时间急剧减少。不少同学也尝试着改变，比如我们软工02班的同学，据我统计，光是西13，5楼的15名男同学中就有11名同学在学校健身房办理了一年及以上的健身卡，另外因为我热爱健身，从入学浙江工业大学以来已经教过12名同学如何健身。看似大家对运动和健身十分积极但是遗憾的是到目前为止除我外只有一名同学坚持训练一年以上，并且他也在最后选择了放弃。

虽然大家最后没能坚持，但是健身十分有益。我作为爱好者也经常鼓舞大家去健身房健身。首先我们能够明确的一点就是健身房能够锻炼我们的身体，让我们的身体状况得到提高，在这个过程中，我们的心肺功能得到加强，肌肉力量也得到提高。长期坚持力量训练自己有氧训练能够让我们抵抗外界疾病的能力得到提高，降低患病的几率。并且让我们的身材更加有型，保持一种年轻的状态，精神面貌也得到改善，健身时产生的内啡肽也能够让我们在工作和生活上更加有激情。另外健身房相对于一般健身场所最大的优点就在于更加全面、专业的器材设备。没错，一般健身房都会有有氧区以及力量区，并且还有着瑜伽区等等，我们想要练习一块肌肉，都能在健身房找到3种以上的相应器材，不得不说健身房里器材的多样性是任何地方都无法比拟的。对于刚刚到健身房里健身的朋友虽然看到如此繁杂的器材摸不着头绪，但是只要简单的了解一遍，就能够轻松的上手。并且健身房里的健身器材还能够让我们开阔视野，了解更多关于健身的知识。

抱着对大家放弃健身的不解我开始了自己的观察得出了大家放弃健身的几个重要原因：

1. 没有制定严格的训练计划。万事开头难，逼迫自己每周固定时间健身房打卡羊城习惯至关重要。
2. 虽然上述半数男生接受过我的指导，但一旦作为教练的我不在大家还是经常忘记动作。举那个和我健身一年多的男生为例，他和我一起健身一年所有动作里都还有部分记不清，反而健身房有哪些女生记得一清二楚。
3. 饮食习惯不知道如何改变，不会计算热量亏损和盈余。健身开始时饮食和健身一样重要，很多增肌者不知道自己一天应该吃多少导致没有热量盈余长肌肉；减脂者不知道热量是否亏损运动再多也是徒劳。

**我的解决方案**

综上所述，我希望能自己设计开发一款Android app，让大家在手机上就能制定自己的计划、记录健身动作、完成热量记录和计算。

#### 开发环境

### 2.1硬件环境

操作系统：Win10

数据库：SQLite

JDK1.7 及以上

兼容 Android 版本 4.3 及以上版本

目标设备：android 手机

### 2.2软件环境

开发平台：Android Studio

开发语言：Java

#### 软件设计

### 3.1软件名称

基于Android的健身房助手——GymKeeper

### 3.2软件详细描述

GymKeeper软件是用来帮助健身者制定自己的计划、记录健身动作、完成热量记录和计算的运动软件。让大家不仅能保持健身习惯不放弃，还能提高效率，让自己更加专业。

### 3.3功能要求

1. 实现健身计划添加；
2. 实现健身计划时间拟定和有氧无氧种类显示；
3. 实现健身计划同步到手机自带日历并给予提示；
4. 实现健身计划的完成和删除功能；
5. 实现从动作库中添加训练动作到目前健身列表中；
6. 运动目标时长记录和计时功能；
7. 实现本次健身消耗热量功能；
8. 实现本次运动和时间占目标占比功能。；
9. 实现自定义动作加入动作库功能；
10. 实现食品热量数据库查询增删功能；
11. 实现从食品热量库添加到我的饮食功能；
12. 实现今日食物饮食、营养和热量记录功能；
13. 实现历史热量营养记录和分析功能；

#### 4.实验体设计与分析

### 4.1功能设计

GymKeeper的具体功能如下：

1.点击软件图标进入程序；

2.训练计划模块

2.1训练计划界面：实现健身计划添加、健身计划时间拟定和有氧无氧种类显示、健身计划同步到手机自带日历并给予提示、健身计划的完成和删除功能。长按列表项可触发放弃、完成对话框2.2；按加号触发添加计划对话框2.3。通过上面的标签或者左右滑动屏幕进入其他页面；

2.2放弃、完成对话框：已过期计划可标记完成或放弃，未进行可标记放弃；

2.3添加计划对话框：在对话框中选择时间并添加描述和训练类型；

3.开始训练模块

3.1开始训练界面：实现从动作库中添加训练动作到目前健身列表中、运动目标时长记录和计时功能以及本次健身消耗热量、本次运动和时间占目标占比功能、自定义动作加入动作库功能。点击从动作库中添加动作，进入动作添加对话框3.2；点击开始训练记录时间；点击加号进入自定义动作对话框3.3。通过上面的标签或者左右滑动屏幕进入其他页面；

3.2动作添加对话框:长按列表项添加动作；

3.3自定义动作对话框：填写动作的各个信息点击确认提交；

4.食物记录模块

4.1食物记录界面：实现食品热量数据库查询增删功能、从食品热量库添加到我的饮食功能。输入视频名会跳出可能符合的列表；点击查询会显示该食物所有信息；点击自定进入自定义食物对话框4.2。通过上面的标签或者左右滑动屏幕进入其他页面；

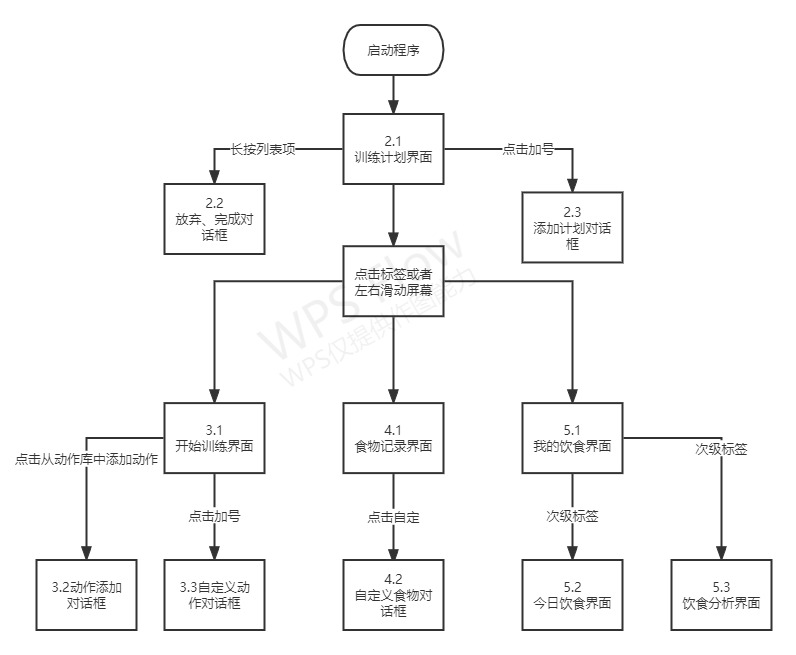
4.2自定义食物对话框：输入食品名和基本信息会加入食品库；

5.我的饮食模块

5.1我的饮食界面：通过次级标签进入今日饮食界面5.2和饮食分析界面5.3；

5.2今日饮食界面：记录今日的食物热量以及各餐食物的营养热量；

5.3饮食分析界面：实现历史热量营养记录和分析功能；

功能总流程图：  


### 4.2架构设计

**训练计划模块：**实现从动作库中添加训练动作到目前健身列表中、运动目标时长记录和计时功能以及本次健身消耗热量、本次运动和时间占目标占比功能、自定义动作加入动作库功能。

**开始训练模块：**实现食品热量数据库查询增删功能、从食品热量库添加到我的饮食功能。

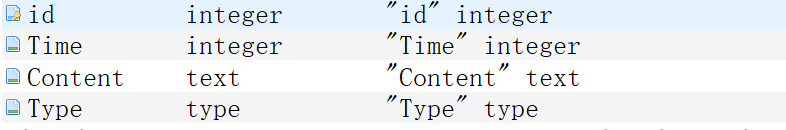
**食物记录模块：**实现食品热量数据库查询增删功能、从食品热量库添加到我的饮食功能。

**我的饮食模块：**实现今日食物饮食、营养和热量记录功能实现历史热量营养记录和分析功能；

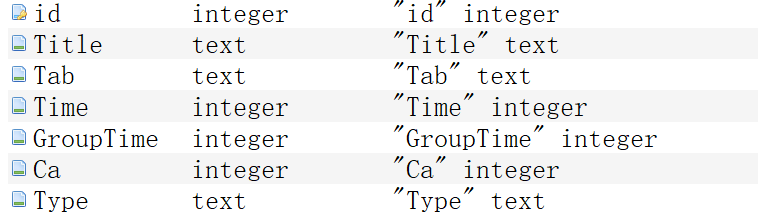
### 4.3数据库设计

**确认所需数据库：**通过功能模块分析可知需要训练数据表、动作表、食物表、饮食表

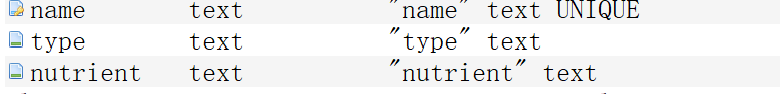
1. 训练计划表Plan



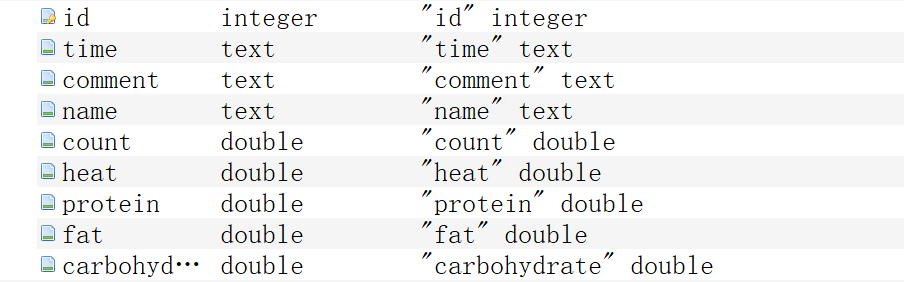
1. 动作表Start



1. 食物表food



1. 饮食表diet



### 4.4界面设计

2.训练计划模块

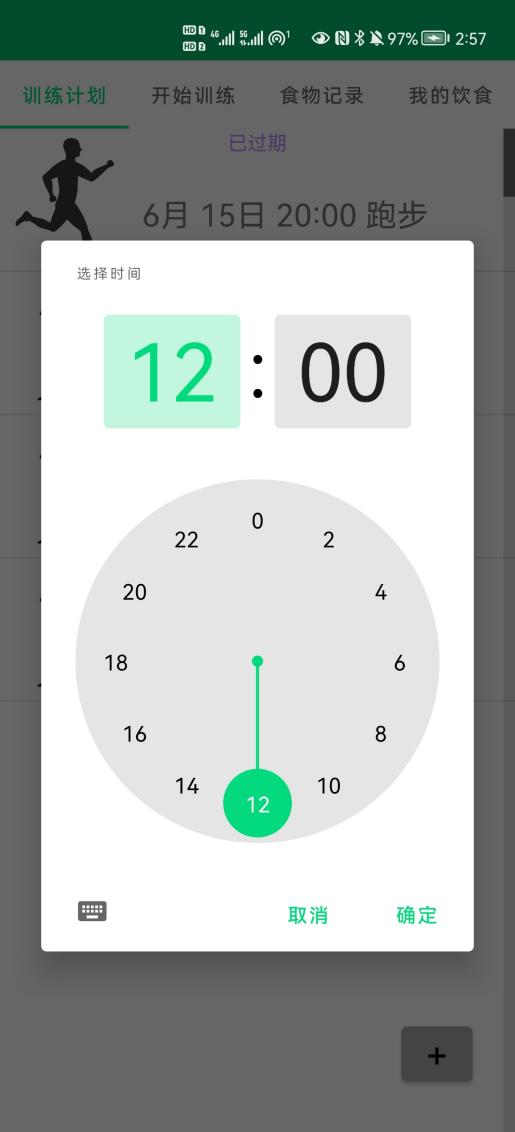
2.1训练计划界面：实现健身计划添加、健身计划时间拟定和有氧无氧种类显示、健身计划同步到手机自带日历并给予提示、健身计划的完成和删除功能。长按列表项可触发放弃、完成对话框2.2；



2.2放弃、完成对话框：已过期计划可标记完成或放弃，未进行可标记放弃；



2.3添加计划对话框：在对话框中选择时间并添加描述和训练类型；



3.开始训练模块

3.1开始训练界面：实现从动作库中添加训练动作到目前健身列表中、运动目标时长记录和计时功能以及本次健身消耗热量、本次运动和时间占目标占比功能、自定义动作加入动作库功能。点击从动作库中添加动作，进入动作添加对话框3.2；点击开始训练记录时间；点击加号进入自定义动作对话框3.3。



3.2动作添加对话框:长按列表项添加动作；



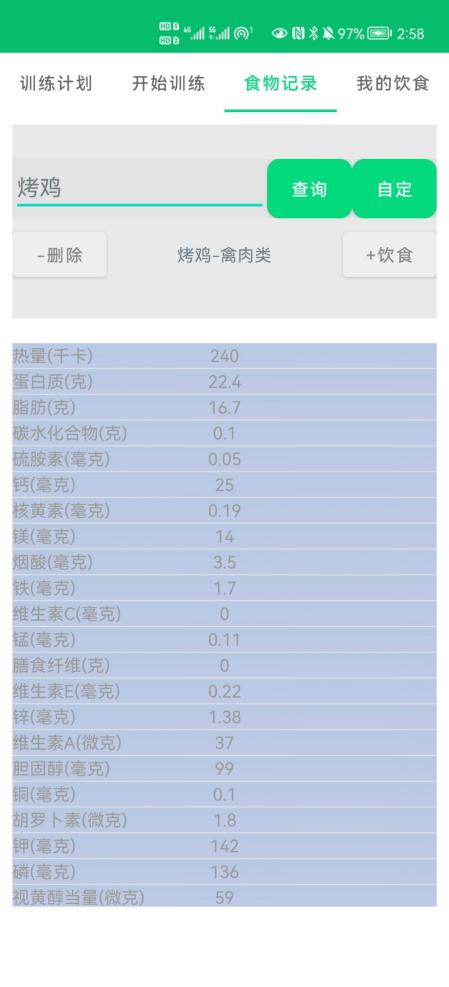
3.3自定义动作对话框：填写动作的各个信息点击确认提交；



4.食物记录模块

4.1食物记录界面：实现食品热量数据库查询增删功能、从食品热量库添加到我的饮食功能。输入视频名会跳出可能符合的列表；点击查询会显示该食物所有信息；点击自定进入自定义食物对话框4.2。





4.2自定义食物对话框：输入食品名和基本信息会加入食品库；



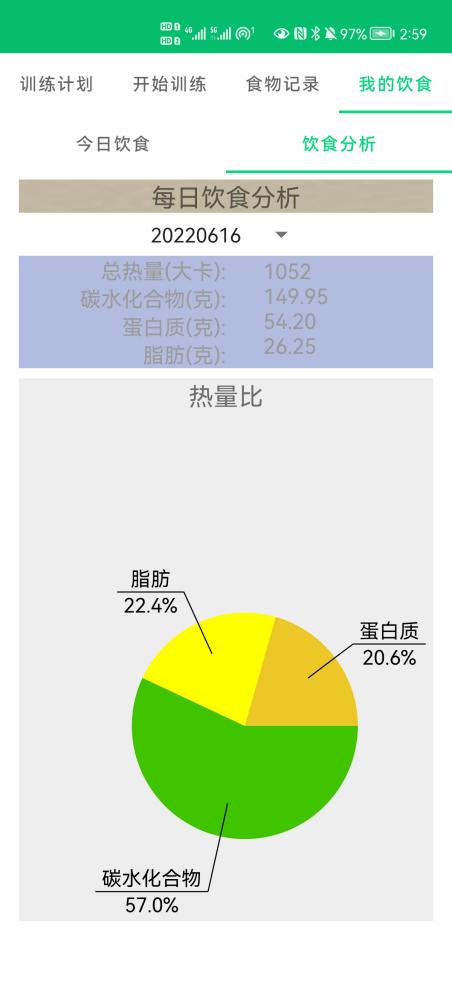
5.我的饮食模块

5.1我的饮食界面：通过次级标签进入今日饮食界面5.2和饮食分析界面5.3；

5.2今日饮食界面：记录今日的食物热量以及各餐食物的营养热量；

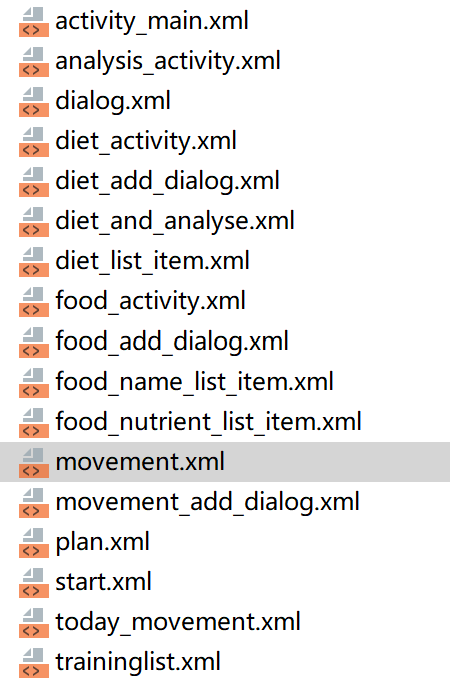


5.3饮食分析界面：实现历史热量营养记录和分析功能；



#### 5 详细设计与实现

**5.1布局实现**

:唯一事件存主标签、pager

：饮食分析界面

：今日饮食界面

：添加饮食对话框

：我的饮食次级标签和pager

：今日饮食表

：食物查询表

：自定食物对话框

：食物查询结果列表

：食物营养列表

：动作列表的项

：动作自定对话框

：计划列表

：开始运动界面

：当前运动对话框

：计划列表的项

**5.2Manifests**

*<?*xml version="1.0" encoding="utf-8"*?>*<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 package="com.example.spotsmusic">  
  
 <uses-permission android:name="android.permission.READ\_CALENDAR" />  
 <uses-permission android:name="android.permission.WRITE\_CALENDAR" />  
  
 <application  
 android:allowBackup="true"  
 android:icon="@mipmap/ic\_launcher"  
 android:label="@string/app\_name"  
 android:roundIcon="@mipmap/ic\_launcher\_round"  
 android:supportsRtl="true"  
 android:theme="@style/Theme.SpotsMusic">  
  
 <activity  
 android:name="com.example.spotsmusic.MainActivity"  
 android:exported="true">  
 <intent-filter>  
 <action android:name="android.intent.action.MAIN" />  
  
 <category android:name="android.intent.category.LAUNCHER" />  
 </intent-filter>  
 </activity>  
 </application>  
  
</manifest>

#### 6 难点分析

1. 使用修改手机原有日历时，要记得先搜索手机数据库中日历表，并记得使用权限申请弹窗或在手机设置中允许日历修改权限。

public void setCalendar(Calendar calendar,String title, String content){  
 ContentResolver cr = requireActivity().getContentResolver();  
 ContentValues values = new ContentValues();  
 Cursor cur = null;  
 Uri uri = CalendarContract.Calendars.*CONTENT\_URI*;  
 final String[] EVENT\_PROJECTION = new String[] {  
 CalendarContract.Calendars.*\_ID*, *// 0* };  
 cur = cr.query(uri, EVENT\_PROJECTION, null, null, null);  
 cur.moveToFirst();  
 final int PROJECTION\_ID\_INDEX = 0;  
 Long calID = cur.getLong(PROJECTION\_ID\_INDEX);  
 long startMillis=calendar.getTimeInMillis();  
 long endMillis=calendar.getTimeInMillis()+600000;  
 values.put(CalendarContract.Events.*DTSTART*, startMillis);  
 values.put(CalendarContract.Events.*DTEND*, endMillis);  
 values.put(CalendarContract.Events.*TITLE*, title);  
 values.put(CalendarContract.Events.*DESCRIPTION*, content);  
 values.put(CalendarContract.Events.*CALENDAR\_ID*, calID);  
 values.put(CalendarContract.Events.*EVENT\_TIMEZONE*, "America/Los\_Angeles");  
 uri = cr.insert(CalendarContract.Events.*CONTENT\_URI*, values);  
  
*// get the event ID that is the last element in the Uri* long eventID = Long.*parseLong*(uri.getLastPathSegment());  
 }

1. 导入他人写好的数据库时可以采用在第一次启动时将res文件中的数据库读到软件对应app下data文件夹中。

public SQLiteDatabase openDatabase() {  
 try {  
 String databaseFilename = DATABASE\_PATH + "/" + DATABASE\_FILENAME;  
 File dir = new File(DATABASE\_PATH);  
 if (!dir.exists())  
 dir.mkdir();  
 if (!(new File(databaseFilename)).exists()) {  
 *// 获得封装food.db文件的InputStream对象* InputStream is = context.getResources().openRawResource(R.raw.*fitness*);  
 FileOutputStream fos = new FileOutputStream(databaseFilename);  
 byte[] buffer = new byte[BUFF\_SIZE];  
 int count = 0;  
 *// 复制文件* while ((count = is.read(buffer)) > 0) {  
 fos.write(buffer, 0, count);  
 }  
  
 fos.close();  
 is.close();  
 }  
 SQLiteDatabase database = SQLiteDatabase.*openOrCreateDatabase*(databaseFilename, null);  
 return database;  
 } catch (Exception e) {  
  
 }  
 return null;  
}

1. 使用嵌套pager是需要一个fragment来存放pager变量和作为控制它的组件。

public void onViewCreated(@NonNull View view, @Nullable Bundle savedInstanceState) {  
 Bundle args = getArguments();  
 context = getContext();  
  
 ViewPager2 viewPager = view.findViewById(R.id.*dietPager*);  
 DietAdapter dietAdapter= new DietAdapter(requireActivity());  
 viewPager.setUserInputEnabled(false);  
 viewPager.setAdapter(dietAdapter);  
  
 TabLayout tabLayout = view.findViewById(R.id.*dietTab\_layout*);  
 new TabLayoutMediator(tabLayout, viewPager,  
 new TabLayoutMediator.TabConfigurationStrategy() {  
 @Override  
 public void onConfigureTab(@NonNull TabLayout.Tab tab, int position) {  
 if(position==0)  
 tab.setText("今日饮食");  
 if(position==1)  
 tab.setText("饮食分析");  
 }  
 }  
 ).attach();  
  
}

1. deco第三方库使用切记要让最大值大于最小值和当前值，通过+0.01f来解决可能的报错。

arcView.addSeries(new SeriesItem.Builder(Color.*argb*(255, 200, 200, 200))  
 .setRange(0, (float)Ca+0.1f, (float) Ca+0.1f)  
 .setLineWidth(32f)  
 .build());  
  
*//Create data series track* SeriesItem seriesItem1 = new SeriesItem.Builder(Color.*argb*(255, 64, 196, 0))  
 .setRange(0,(float) Ca+0.1f,Cost)  
 .setLineWidth(32f)  
 .build();

1. 使用时钟组件记得判断是否大于目前事件再加入日历。插入时记得对数据库进行查重。

if(date.before(Calendar.*getInstance*())){  
 Toast.*makeText*(context,"请设置晚于目前的时间！",Toast.*LENGTH\_SHORT*).show();  
 return;  
}

int count = 0;  
 *//定义可操作的数据库对象* db =dbHelper.getWritableDatabase();  
 *//设置Cursor对象，用来查看数据库中的信息* Cursor cursor = db.query("Plan",null,null,null,null,null,null);  
 if (cursor.moveToFirst()){  
 do{  
 Long time = cursor.getLong(1);  
  
 *//判断数据库中是否已经存在输入，或者是否存在输入的信息相同的信息* if (time==date.getTimeInMillis()){  
 count ++;  
 }  
 }while (cursor.moveToNext());  
 }  
  
  
*// 如果输入的信息不相同，也就是count没有进行运算* if (count == 0){  
 SQLiteDatabase db = dbHelper.getWritableDatabase();  
 ContentValues values = new ContentValues();  
 values.put("Content",content);  
 values.put("Time", date.getTimeInMillis());  
 values.put("Type",type);  
 long res = db.insert("Plan",null,values);  
 plans.add(new Plan(content,date,type));  
 values.clear();  
 setCalendar(date,type,content);  
 Toast.*makeText*(getContext(),"保存成功！" + String.*valueOf*(res),Toast.*LENGTH\_SHORT*).show();  
 PlanAdapter adapter=new PlanAdapter(context,plans);  
 listViewPlan.setAdapter(adapter);  
 }else{  
 Toast.*makeText*(getContext(),"该时间已有运动！",Toast.*LENGTH\_SHORT*).show();  
 }

#### 7 实验总结

在本次课程设计中利用Android开发了健身房助手，熟悉了Android下各种控件的使用。本次实验中最重要的是数据库第三方库的使用和高级组件的使用。

数据库不光要会自己建表做增删查改，还要会用网上的数据库文件，导入到自己的项目中来。第三方库和组件十分便捷美观，性能优化的同时达到了很复杂的功能让开发过程事半功倍。高级组件不仅美观灵活而且有更好的人机交互和更强大更综合的功能。

在实验中，体会到开发一个工程时，应该先制定好程序的框架，规划好相应的功能模块，使程序模块化，易于日后的扩展和完善。其次是程序的数据结构，良好的数据结构能使程序高效化，功能强大。要严格按照需求分析，设计，编码，优化的顺序进行开发。