

# Days APP软件设计文档

2017年7月3日

专业：数据科学与计算机学院

课程：软 件 设 计

项目：Days App

成员：14331128 黎睿德

14331140 李旭东

1433xxxx 黄嘉豪

14331112 江中诚

目录

[Days APP软件设计文档 1](file://localhost/Volumes/LXD/软件设计文档.docx" \l "_Toc486173852)

[一、 应用概述 3](#_Toc486173853)

[二、 技术选型 3](#_Toc486173854)

[三、 架构设计 3](#_Toc486173855)

[四、 设计技术 6](#_Toc486173856)

[1. Structure Programming 6](#_Toc486173857)

[2. Object-Oriented Programming 8](#_Toc486173858)

[3. Design Patterns 9](#_Toc486173859)

[a． 视图 9](#_Toc486173860)

[b． 模型 9](#_Toc486173861)

[c． 控制器 10](#_Toc486173862)

[五、模块划分 10](#_Toc486173863)

[六、 具体设计 10](#_Toc486173864)

[Widget： 10](#_Toc486173865)

# 应用概述

Days是我们四人在大三上的安卓开发课程所做的课程项目，是一款用于记录重要日子，并提供倒数日提醒的app。包括的主要功能有：创建和管理日子；日子距离天数提醒；摇一摇传感器应用；桌面控件提醒等。

# 技术选型

1. 客户端： Android （JAVA 技术）

理由：由于我们设计的应用功能是一个计算日期差的工具，作为一款小工具，适合在移动端实现，对用户来说也比较方便。

另外，选择Android技术也因为技术开发人员均为学安卓出身，对安卓技术较为熟练，可减小开发难度，节省成本。

最后，选择Android开发也是因为我们使用MVC设计模式，而Android本身就支持MVC设计框架，这样可以更容易地实现项目规划的内容。

1. 数据存储：SQLite 数据库

理由：SQLite是一个开源的嵌入式关系数据库，支持SQL92语法，更加适用于嵌入式系统，嵌入到使用它的应用程序中，并且占用非常少，运行高效可靠，可移植性好，而且屏蔽了数据库使用和管理的复杂性，程序仅需要最基本的数据操作，其他操作可以交给进程内部的数据库引擎完成。

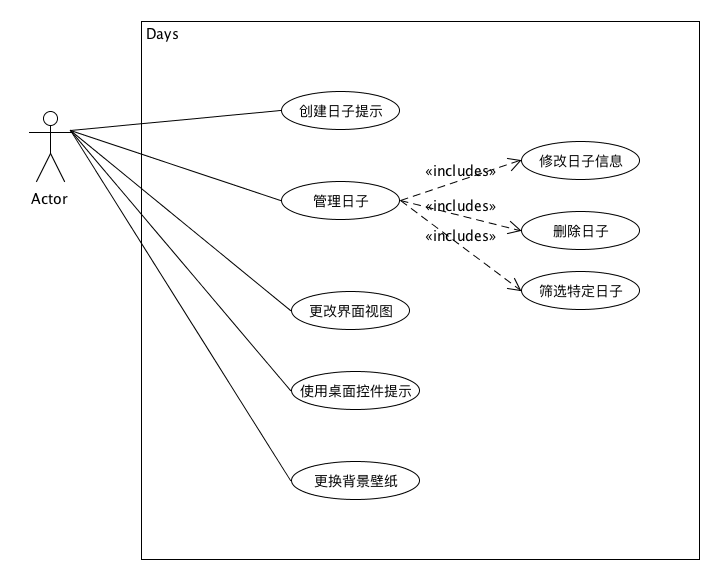
此外，由于此应用所需要存储的数据，只是日期等信息，因此使用Android本身内置的轻量级数据库SQLite最合适。

1. 桌面控件：Widget开发技术

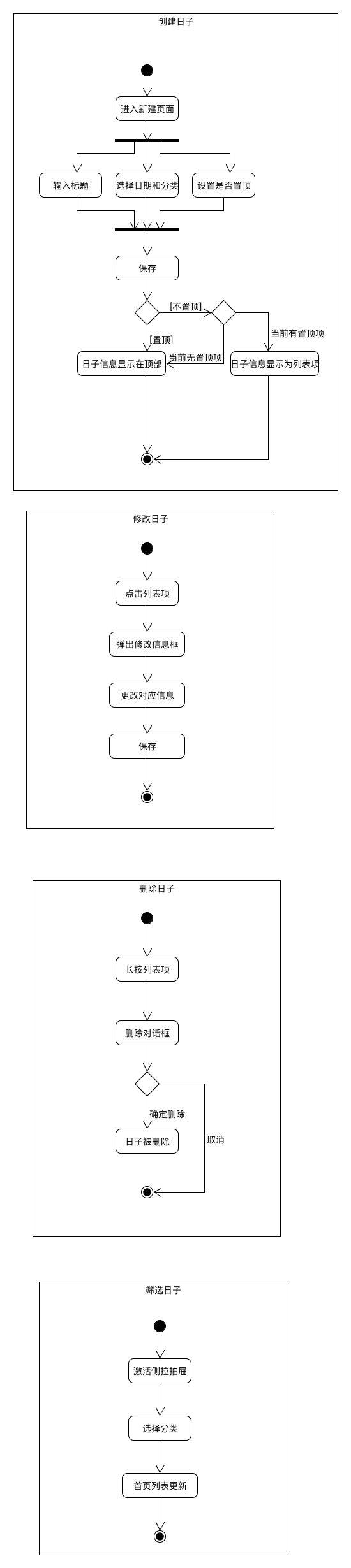
理由：由于设计的APP是工具类应用，我们计划在桌面放置小型控件以显示应用的信息，这样用户无需进入应用就可以看到提醒。要实现这样的控件，我们自然就选择Widget开发。

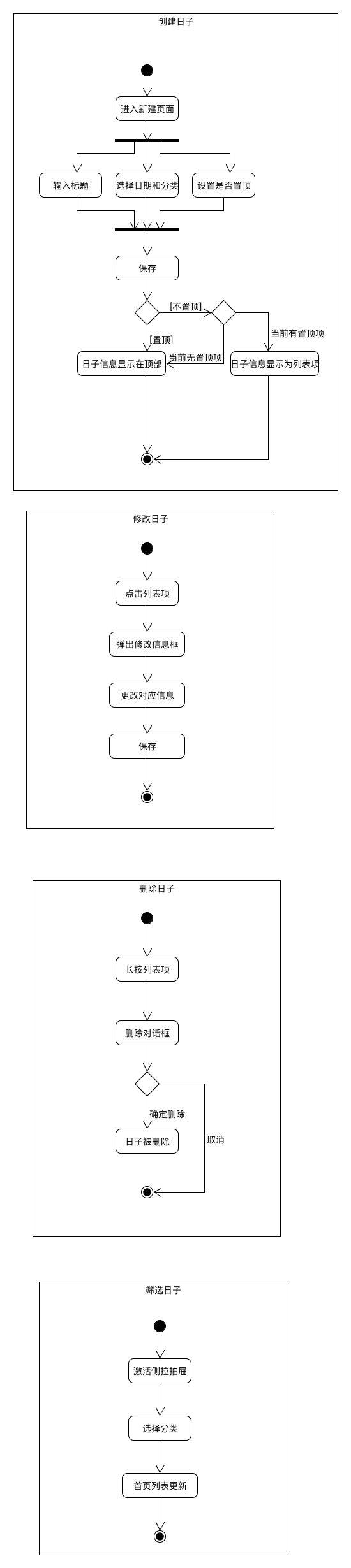
# 架构设计

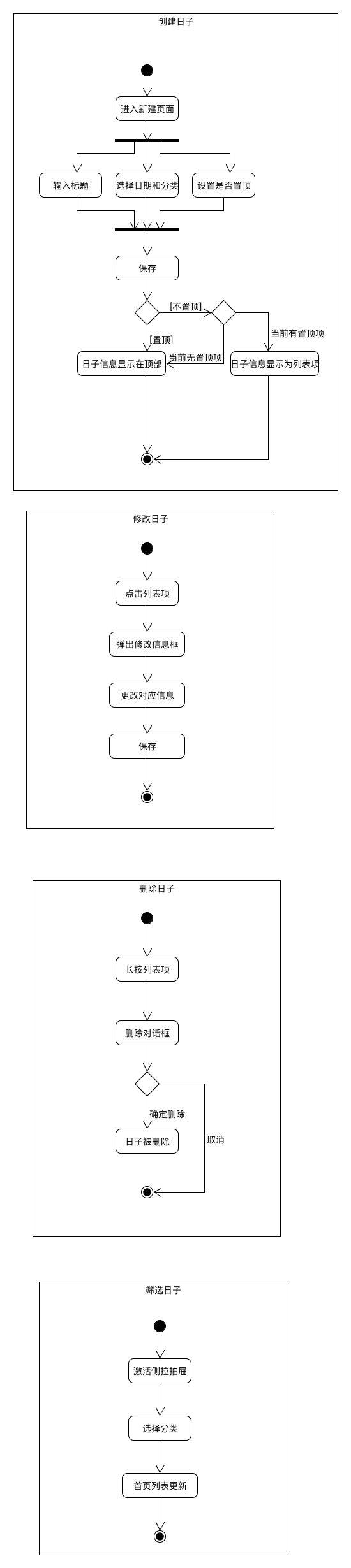
1.用例图

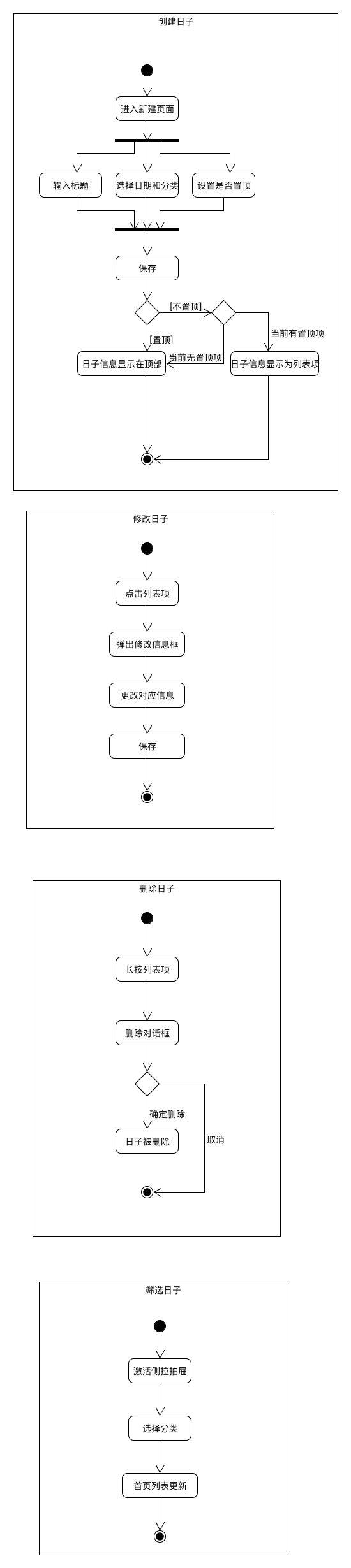


2.活动图

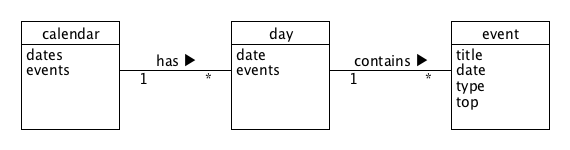






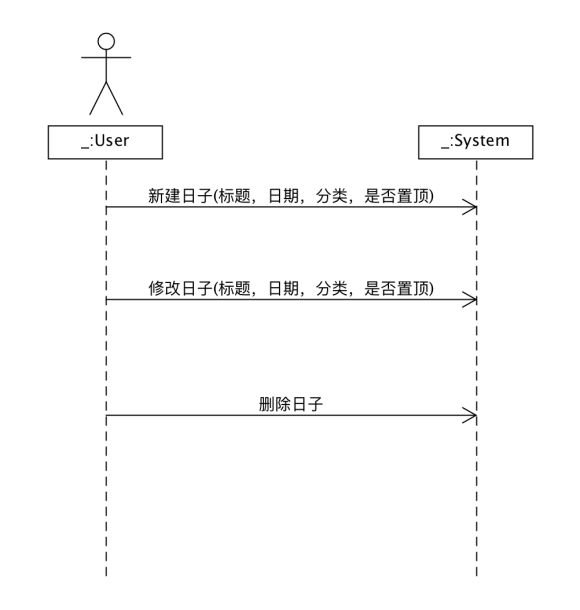


3.领域模型

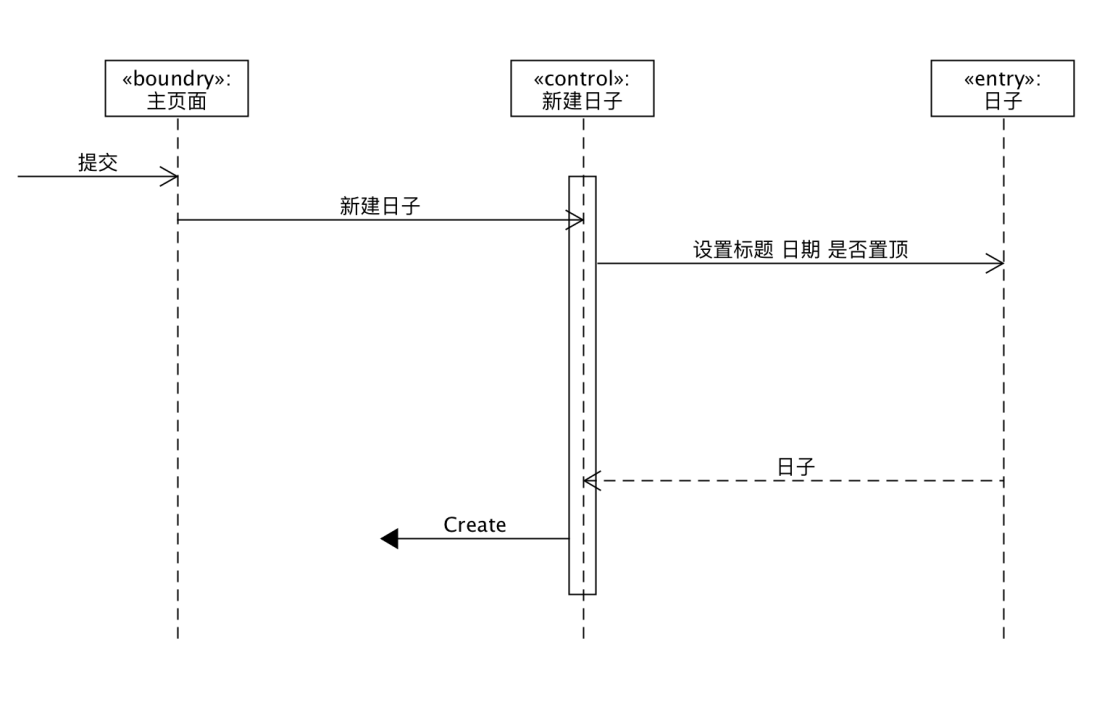


架构设计—系统顺序图、ECB逻辑架构设计、映射到具体代码】

1. 系统顺序图



1. ECB逻辑架构设计

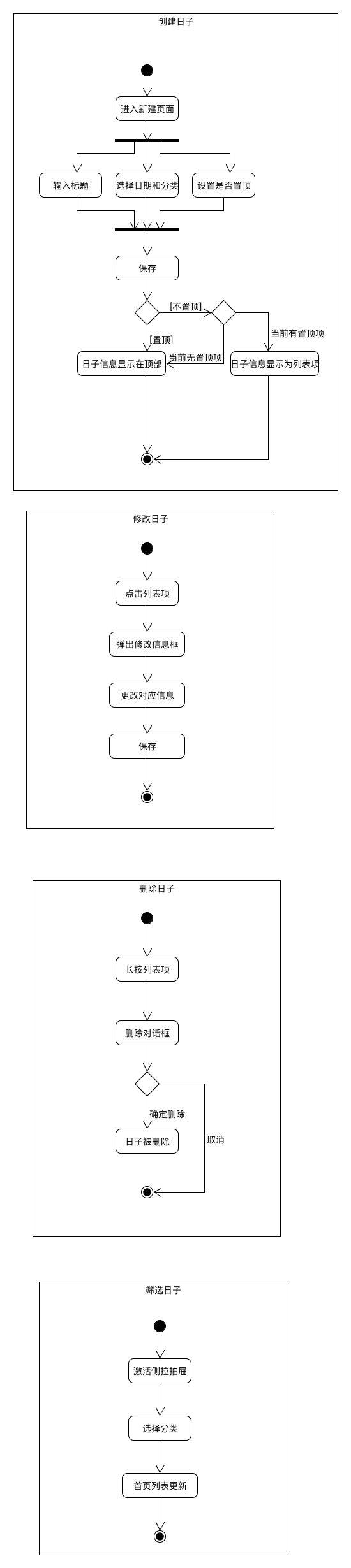


# 设计技术

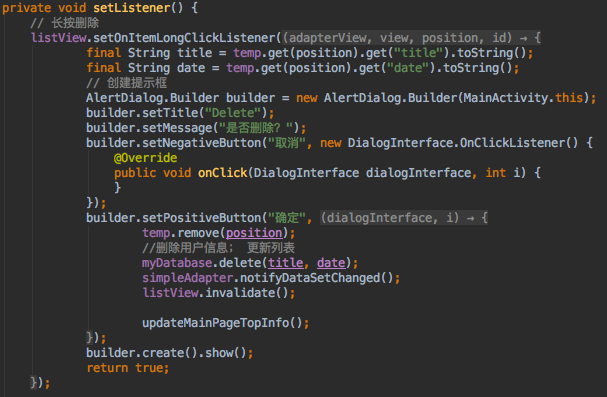
## Structure Programming

结构化程序设计是进行以模块功能和处理过程设计为主的详细设计的基本原则。结构化程序设计是过程式程序设计的一个子集，它对写入的程序使用逻辑结构，使得理解和修改更有效更容易。

应用这种结构化程序设计，我们先绘制了活动图，将应用涉及的操作弄清楚后，相应地去实现代码，以下面的活动图为示例：



对应在代码里，截图如下：

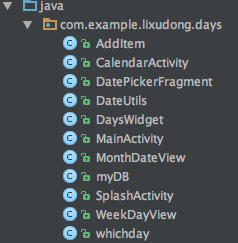


## Object-Oriented Programming

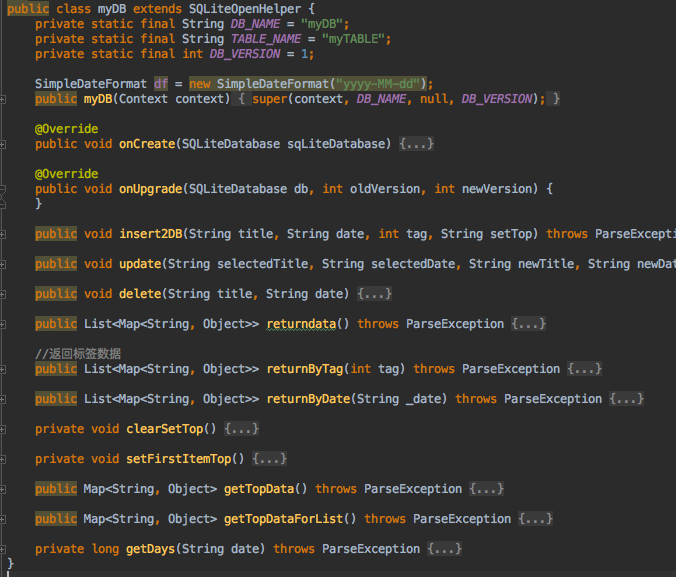
面向对象编程是一种计算机编程架构。OOP的一条基本原则是计算机程序是有单个能够起到子程序作用的单元或对象组合而成的。OOP达到了软件工程的三个主要目标：重用性、灵活性和扩展性。我们的项目是Android项目，使用JAVA语言，本身就是一种面向对象编程语言。

在设计中，我们主要设计了Activity类、日历类、记录日期的Day类还有数据库类，然后各自封装起来。各个类设计接口以便其他对象调用，传递消息则使用Intent来实现不同Activity的结构化信息。

如下图，是我们项目中设计的各个类：



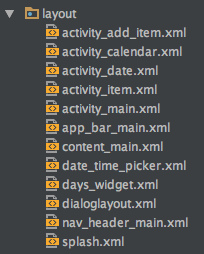
下图是对数据库类myDB的封装示例，其他类的详细封装细节就请看代码文件：



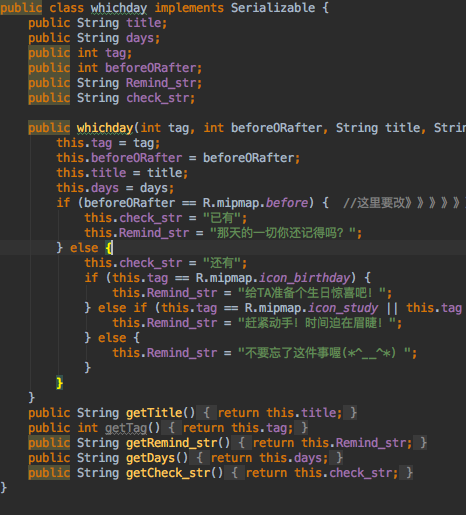
## Design Patterns

本应用采用了MVC设计模式。MVC全名是Model View Controller，是模型(model)－视图(view)－控制器(controller)的缩写，一种软件设计典范，用一种业务逻辑、数据、界面显示分离的方法组织代码，将业务逻辑聚集到一个部件里面，在改进和个性化定制界面及用户交互的同时，不需要重新编写业务逻辑。MVC被独特的发展起来用于映射传统的输入、处理和输出功能在一个逻辑的图形化用户界面的结构中。Android开发本身就是遵循MVC设计模式，体现在代码中的细节如下：

1. 视图：layout文件夹里的xml文件都是我们设计的视图文件。

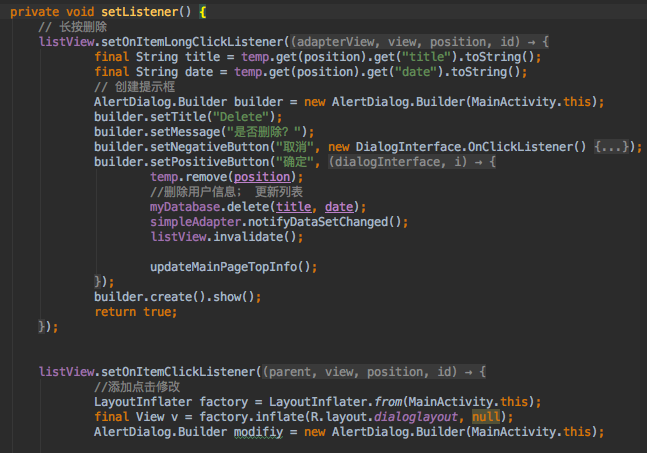


1. 模型：涉及到业务模型，如日期创建与管理、数据库等。



上面的代码是个示例，是我们对记录日的模型定义，包含主题、日期时间、类别等信息，其他的模型详见代码文件。

1. 控制器：JAVA里面的各种监听函数，达到处理事件，响应交互的目的。



上面的代码截图是一个监听函数，如果长按则删除记录，点击则弹出修改对话框。这就是一个典型的控制器。

# 五、模块划分

按界面划分

数据库接口

启动界面

主界面

Widget

Days

日历视图

按实现功能划分

主界面UI

WidgetUI

日历视图UI

删除

修改

添加

事件item操作

数据库接口

UI

摇一摇更换界面

Widget提醒

更换背景

分类

Days

# 具体设计

## Widget：

在桌面实现一个展示置顶信息的小工具，并根据该置顶信息的分类情况显示贴心的提醒语。

运行界面：

①没添加记录时，显示如下：



②当置顶记录的日子在一周之后，显示：



③当距离记录的日子不到一周时间，显示



④当记录的日子已经过去：



⑤当记录的事项是生日时，显示：



使用技术：AppWidget、Broadcast

遇到问题：

①如何根据不同的记录，显示对应的提示语；

②如何在适当地时候更新widget；

解决方法：

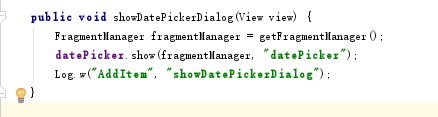
①写一个有关日期记录的类（whichday），在这个类中写返回类型与距离天数的函数，在接收到从mainactivity传来的数据时进行判断。

②根据用户使用场景，应该在退出app界面时更新，所以在onPause()中调用更新widget的函数。截图如下：

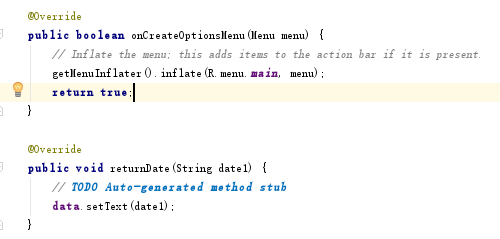


**视图及日历：**

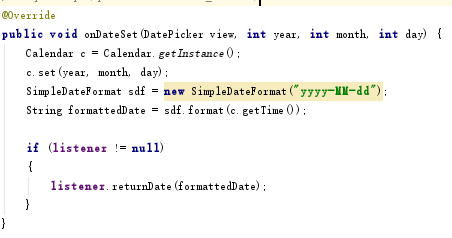
1、首先是完成activity\_add\_item.xml以及修改点击修改界面里的点击日期出现日历对话框，通过网上搜索资料，发现这部分可以调用安卓developers里的API，实现public class DatePickerFragment extends DialogFragment implements DatePickerDialog.OnDateSetListener类，然后当点击日期部分的layout时，然后通过监听事件调用showDatePickerDialog方法创建新的fragmentManager，然后通过调用DatePickerFragment的show方法即可显示出来出日历对话框



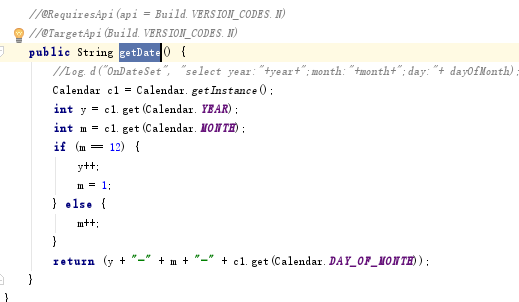
而在此处的关键是如何选择日期点击对话框的确定按钮后以”xxxx-xx-xx”形式返回日期，而后，通过参考codego.net上提供的建议的方法，首先在AddItem中添加以下方法，通过在returnDate里设置addItem里的TextView形如”xxxx-xx-xx”的日期



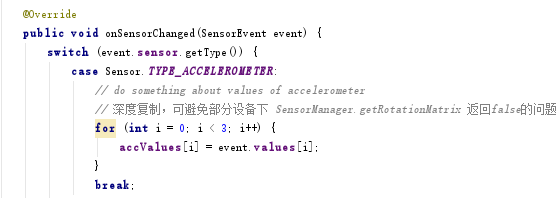
然后在DatePickerFragment类里的onDateSet方法调用returnDate方法

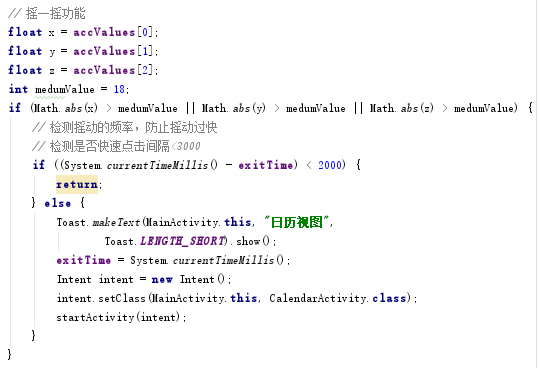


由于Calendar返回的日期与公历有所区别，所以需要在getDate里进行相应的修改（主要是月份从0开始到12）



2、实现摇一摇功能切换，由于最后一次实验已经实现，因此参照最后一次实验完成即可，重写onSensorChanged函数，当手机摇动时，检测加速度传感器，当三个方向上的加速度值大于18时，实现页面跳转





而关于传感器生命周期部分，由于最后一次实验已经实现过，因此按照上次实验部分实现即可，再次不在赘述。

此部分的问题是，由于手机没摇动一次，按照正常使用的习惯，必定出现两次获以上在三个方向上的某个加速度值大于18，因此设置定时器，即检测到第一个加速度值大于18之后的两秒以内，即使仍检测到大于18的加速度值，也不进行页面切换，防止创建多个日历视图的Activity。

1. 实现日历视图部分的布局和自定义日历控件。

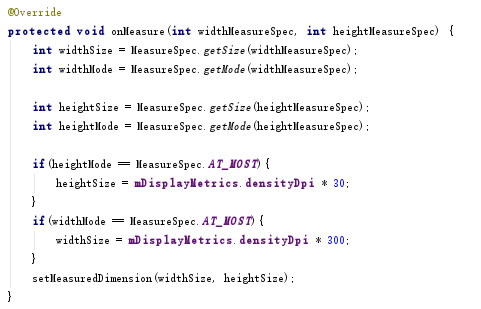
日历的视图布局如下：



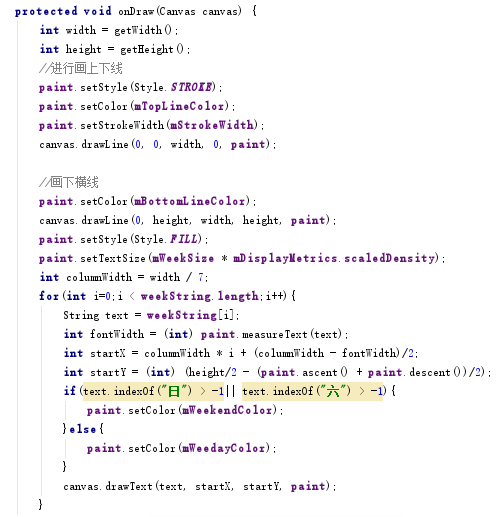
关键在于实现自定义的日历控件。

而我的实现方法是java文件实现相应的WeekDayView和MonthDayView类，然后在XML文件布局时调用，如上图部分。

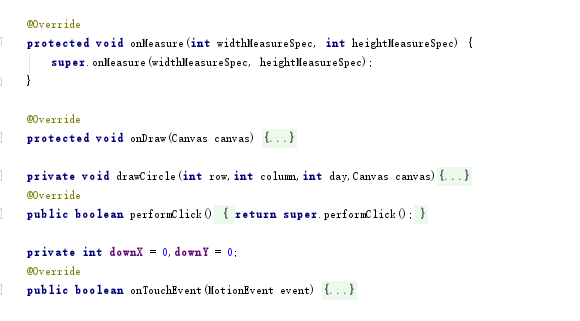
WeekDayView设置日历”一二三四五六日”部分，OnMeasure设置WeekDayView的宽度



Ondraw设置字体，背景色等



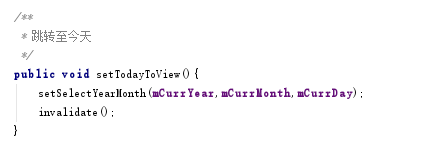
同理，在MonthDayView里也是如此设置



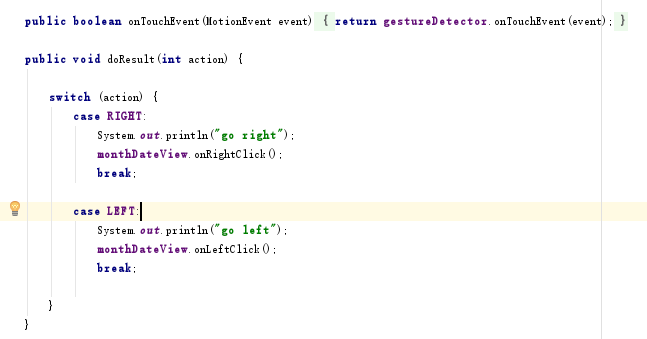
由于此部分参考博客上的代码，因此需要修改的地方较少，参考博客给出的代码结合本次实验设计的样式进行布局，最后布局如下



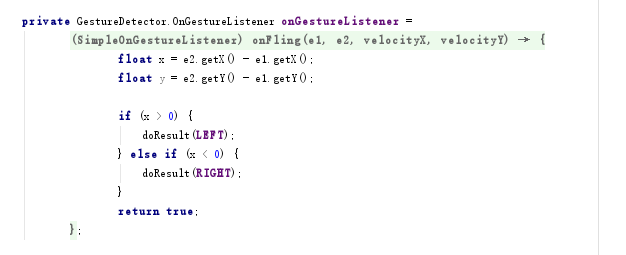
点击”今”回到今天的日期



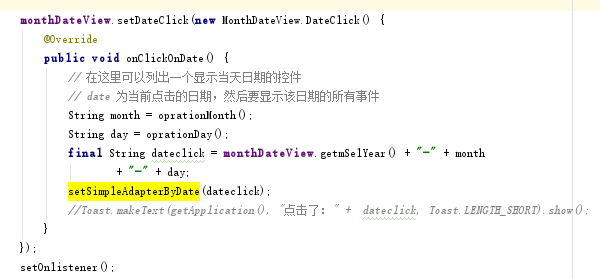
最后在CalendarActivity里整合以及添加相应的事件监听，比如遇到的一个问题首先是该参考代码给出部分的左右移动部分对调，即点击右按钮日历回往前一个月，因此将onTouchEvent里的Right和Left对调即可

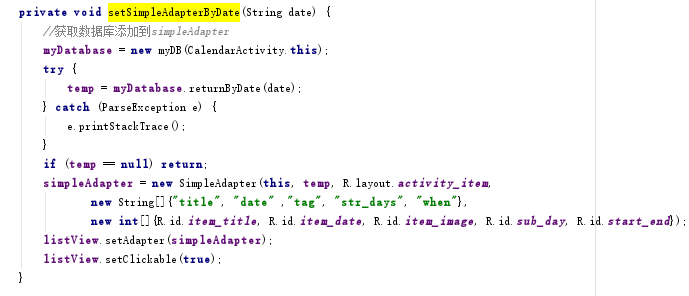


同时实现左划和右划事件监听

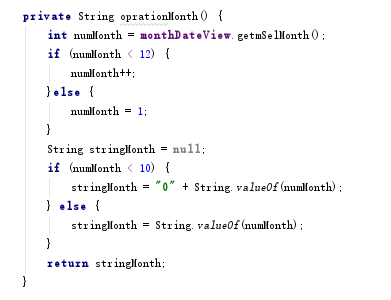


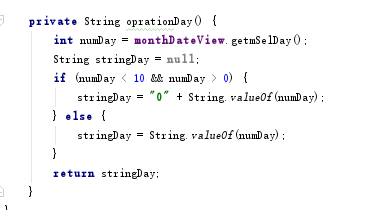
而后需要实现最关键部分就是当点击某个日期时，显示当天的所有事项，通过点击日期获取当天的日期，然后根据当天的日期查询数据库，并在listView中显示出来





不过这里遇到的小问题是由于数据库存储的是形如”xxxx-xx-xx”的日期，而在这里点击日期返回获取到的日期如果月和日是个位数，就会显示”xxxx-x-x”的形式，无法在数据库中实现数据查询，即使当天有时间也无法获取，所以需要对日期格式进行相应的处理，如下：



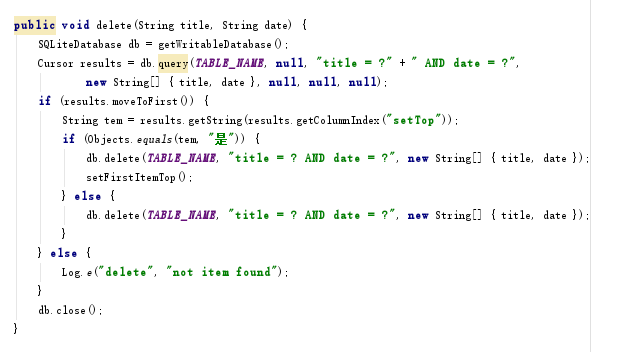


标准化日期格式后即可实现查询。

1. 长按删除Item，并更新列表和数据库，如下：







**实现效果：**



