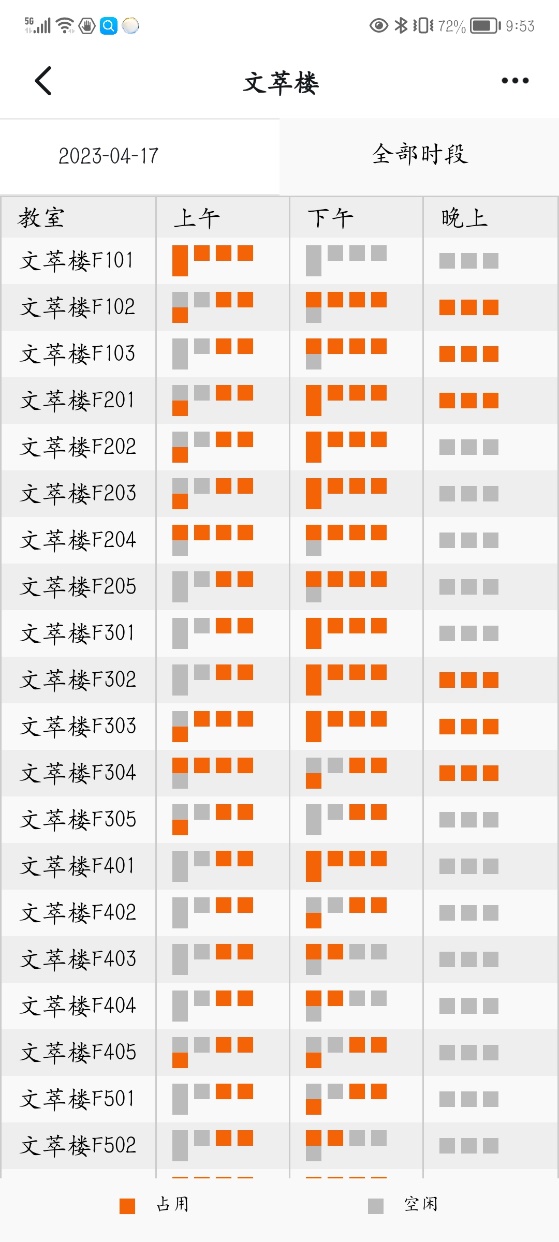
《Android技术开发基础》结课设计

班级\_\_\_07112101\_\_\_学号\_\_\_1120210623\_\_\_\_\_\_\_\_姓名\_\_\_赵宇轩\_\_\_\_\_\_\_\_\_\_\_

当某个时间段没课时，很多同学会去找间空教室自习。而同学们都更喜欢去文萃楼的教室，主要是由于那里教室数量多，并且插座很多。那如何找到空教室呢，最傻的方法是看每间教室门口旁边的电子屏，它显示该教室一天不同时间段的课程，如果没课显示“空闲”。相对聪明的办法，是i北理提供的空闲教室查询功能，如下：



我们可以找其中的灰色方块，它代表这间教室对应时间段是空闲的。但是它有几个不足之处：

1.它只是罗列了每个教室的课程情况，用户不能第一时间得知哪个是空教室，必须一个一个翻找，分辨橙色快和灰色块（尽管这种形式很简便了，但我想使得知道哪间教室现在没课的过程更加简单）

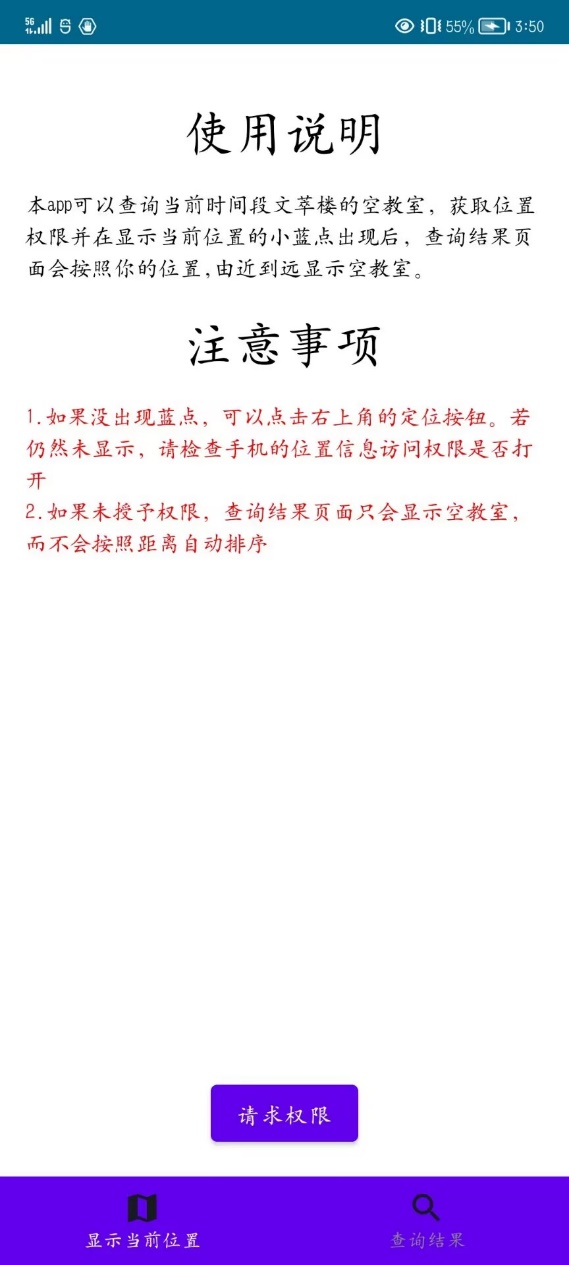
2.它只是机械地罗列课程情况，唯一智能的是它会显示这一天的情况。不能根据用户当前位置和时间段智能筛选出用户最想去的空教室（比如我9：35在文萃楼M下课，那我最想去的是9：55【即第三节课】之后空闲的，最好还在文萃楼M的空教室【毕竟这样最近】。

以上不足其实可以理解，我个人认为i北理这个功能设计的初衷，主要是为了方便那些老师，学生组织找到空教室，来进行调课或者组织活动等。相比它，我想站在一个想找空教室去自习的学生的角度，开发出能第一时间显示空教室，且最先出现的空教室是我“最想去”【这里以距离最近为衡量】的app。

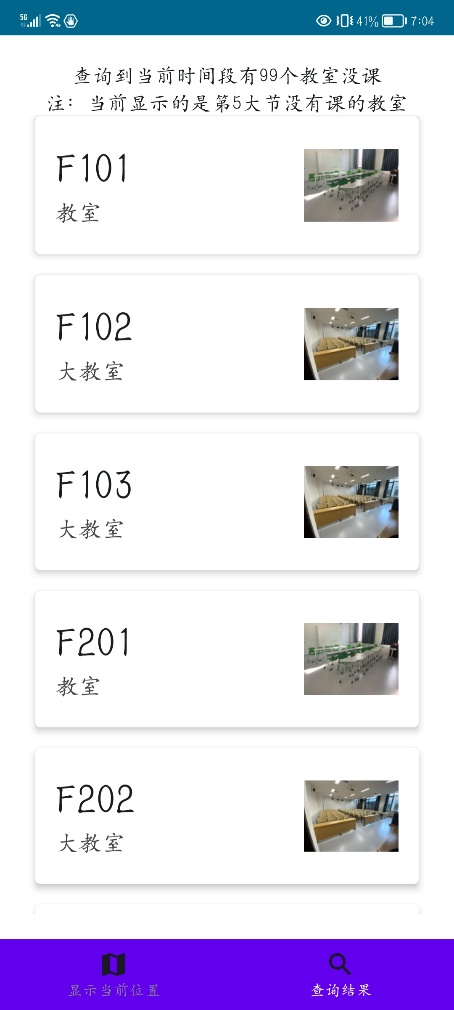
#### 1 App的运行与开发环境

1. **运行环境**： 7.1.1 以上版本Android的Android手机/平板，JDK17
2. **部署方法：** 直接安装Search.apk即可（名称是app-debug.apk，但最后安装完是Search.apk）
3. **开发环境：** Android Studio 2022.2.1，Pycharm（仅用作数据预处理）
4. **手写代码行数：**手机端约1500行（Kotlin），数据预处理100行（Python）

#### 2 App功能说明：

本App一共两个页面：

第一个页面如上，根据用户是否授予访问位置权限而定（没有的话是左边，有的话是右边），我在这里借助高德地图api，来显示用户当前位置



第二个页面如图所示：显示所有空教室。根据用户当前时间不同，文本也不同。例如在早上8：00之前以及8：00-9：35的时间段，会显示第1大节没有课的教室。如果在20：55之后，会显示文本“当前所有教室都没课”

#### 3 App架构设计及技术实现方案

（1）对用到的算法的说明

我需要针对用户的当前位置与六栋教学楼的位置的距离远近，显示由近到远的教室。例如我当前距离文萃楼B只有30m，距离文萃楼F有100米，那程序会先显示文萃楼B的空教室，然后再显示文萃楼F的空教室。

具体算法即排序+哈希表：

val distanceB by distanceViewModel.distanceB.observeAsState(0.0)  
val distanceF by distanceViewModel.distanceF.observeAsState(0.0)  
val distanceG by distanceViewModel.distanceG.observeAsState(0.0)  
val distanceH by distanceViewModel.distanceH.observeAsState(0.0)  
val distanceI by distanceViewModel.distanceI.observeAsState(0.0)  
val distanceM by distanceViewModel.distanceM.observeAsState(0.0)  
val dataList by val dataList by databaseViewModel.courses.observeAsState(*emptyList* ())

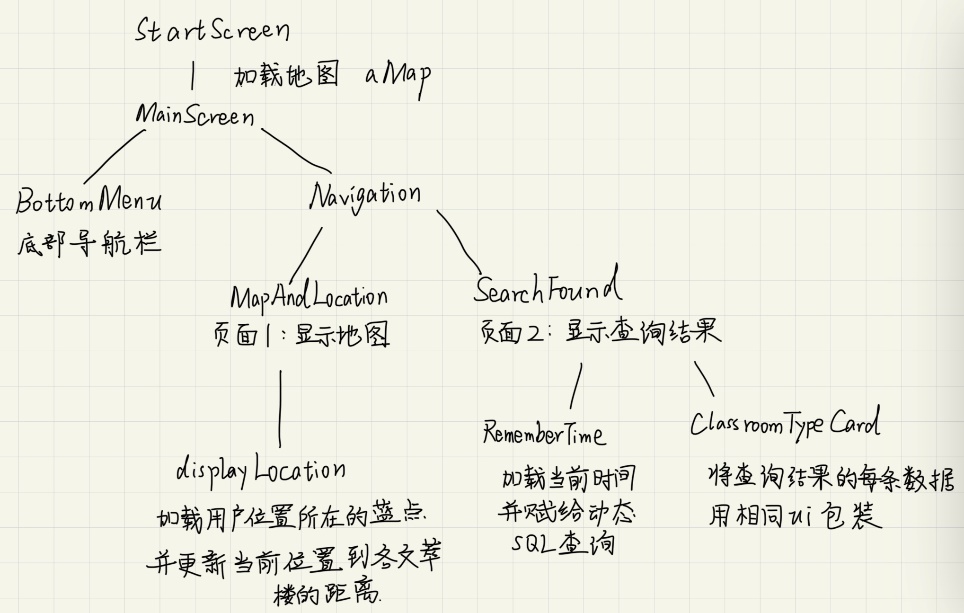
val distanceMap = *mapOf*(  
 "B" *to* distanceB,  
 "F" *to* distanceF,  
 "G" *to* distanceG,  
 "H" *to* distanceH,  
 "I" *to* distanceI,  
 "M" *to* distanceM  
)  
  
val sortedDataList = dataList.*sortedWith*(*Comparator* **{** o1, o2 **->** val distanceO1 = distanceMap[o1.id[0].toString()]  
 val distanceO2 = distanceMap[o2.id[0].toString()]  
 when {  
 distanceO1 == null || distanceO2 == null -> 0  
 distanceO1 < distanceO2 -> -1  
 distanceO1 > distanceO2 -> 1  
 else -> 0  
 }  
**}**)

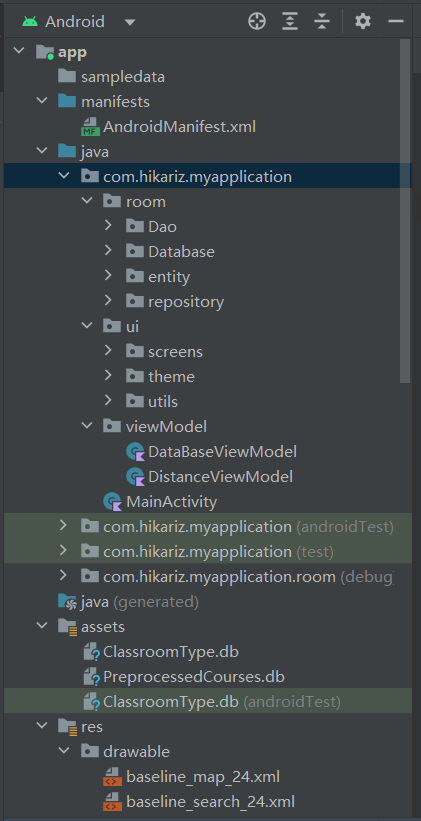
在这里，distanceB表示当前位置到文萃楼B的距离，六个变量都在ViewModel里。这里的databaseViewModel.courses是查询到所有的空闲教室（还未根据距离排序），使用如上排序算法实现从小到大排序，得到sortedDataList后依次在LazyColumn里显示即可

LazyColumn **{** items(sortedDataList.size) **{** index **->** ClassroomTypeCard(sortedDataList[index])  
 **}  
}**

这里引入哈希值的原因是数据库中的教室名都是直接以字母打头的，如文萃楼B的教室都以B开头，建立哈希表distanceMap[distanceB]=“B”

（2）程序架构设计及技术实现方案

我使用了Compose框架，控件树以及每个组合函数实现的功能如下：



程序内的各包划分如上：一共有room，ui和viewModel三个包+MainActivity。

room包存放有关数据库存取的代码，依据职能再划分为Dao，Databse，entity，repository，具体划分依据同课上所讲

ui包存放screen（即各个屏幕），ui（Android stuido自带的），utils（一些功能代码）

viewModel包存放程序要用到的数据，DatabaseViewModel是存放SQL查询后的数据，DistanceViewModel存放

另外，对于程序要用到的数据库db文件，我放到了与java同目录下的文件夹assets中，具体用途我会在后面说明

App导入了以下第三方依赖：高德地图api，导航Navigation，协程，Room，lifecycle，ViewModel

#### 4 技术亮点、技术难点及其解决方案

4.1 相比其它同类app的亮点

（1）对与我有相同目的的学生群体更友好

【我这里对比的“同类app”是我开头提到的i北理内置的相关模块】

如我设计的初衷一致，它能更快，更精确地显示学生最想去的空教室。它操作极其简单，用户只需点击主界面的按钮等待一小会即可显示。而如果使用i北理的那个功能，需要找到这个功能模块，再筛选良乡校区文萃楼，最后再从那个页面寻找信息，并不方便。

当然，我只是对单一模块进行优化，用我的app“专”的部分对比i北理无数模块中的1个有些欺负人。毕竟i北理是全校师生都用的，要兼顾全部校区全部教学楼。并且如同开头所说，这两个app针对的用户群体不一致，我的app仅方便学生。而i北理那个功能适用群体——老师，想找教室开设活动的组织负责人等，他们并不适合用。毕竟我的app只显示实时空教室，他们很难在这么短的时间内召集要参加活动的其他人

（2）推荐方案智能

向用户推荐空教室的顺序是按照用户与教学楼的距离结果排序的（如果要再对每一个教室的距离进行排序的话太麻烦了，我这里记录用户位置到六个教学楼文萃楼B,F,G,H,I,M的距离进行排序。

（3）界面简洁，用户无需进行过多交互，针对不同尺寸屏幕都能适配，健壮性强

由于大量使用协程，以及我个人对程序的诸多优化（在后面会提到），而且本程序无需联网（如果我能直接连接上学校所有教室占用信息的数据库的话可能需要联网，但是我连不上），高德地图可以直接加载本地缓存。仅需要切换一下页面，大约1s即可得到结果。

由于所有控件都是动态尺寸（除了一些间隔为8dp等为固定值），在各种尺寸的手机上都能ui都很美观。而且，由于所有重要的变量都用ViewModel动态观察，即使遇到如手机旋转，扔到后台再点击回来，突然来电话等各种情况都不会影响程序。

4.2 技术亮点

（1）我的代码使用了Compose框架，并且遵循AAC的架构，维护多个ViewModel，使用协程读取后台数据库，使用LazyColumn来返回数据库显示到的多条记录等，性能高

（2）我导入了高德地图api，在用户同意授予app访问位置信息访问权限后，可以用蓝点显示当前位置以及根据当前位置与六栋教学楼的距离远近智能推荐最近的楼里的教室

（3）代码可读性高，各组合函数功能明确且不臃肿（一个函数一般不超过60行）

4.3 技术难点

（1）如何获取全校的课程信息表并转化为程序方便读取的形式

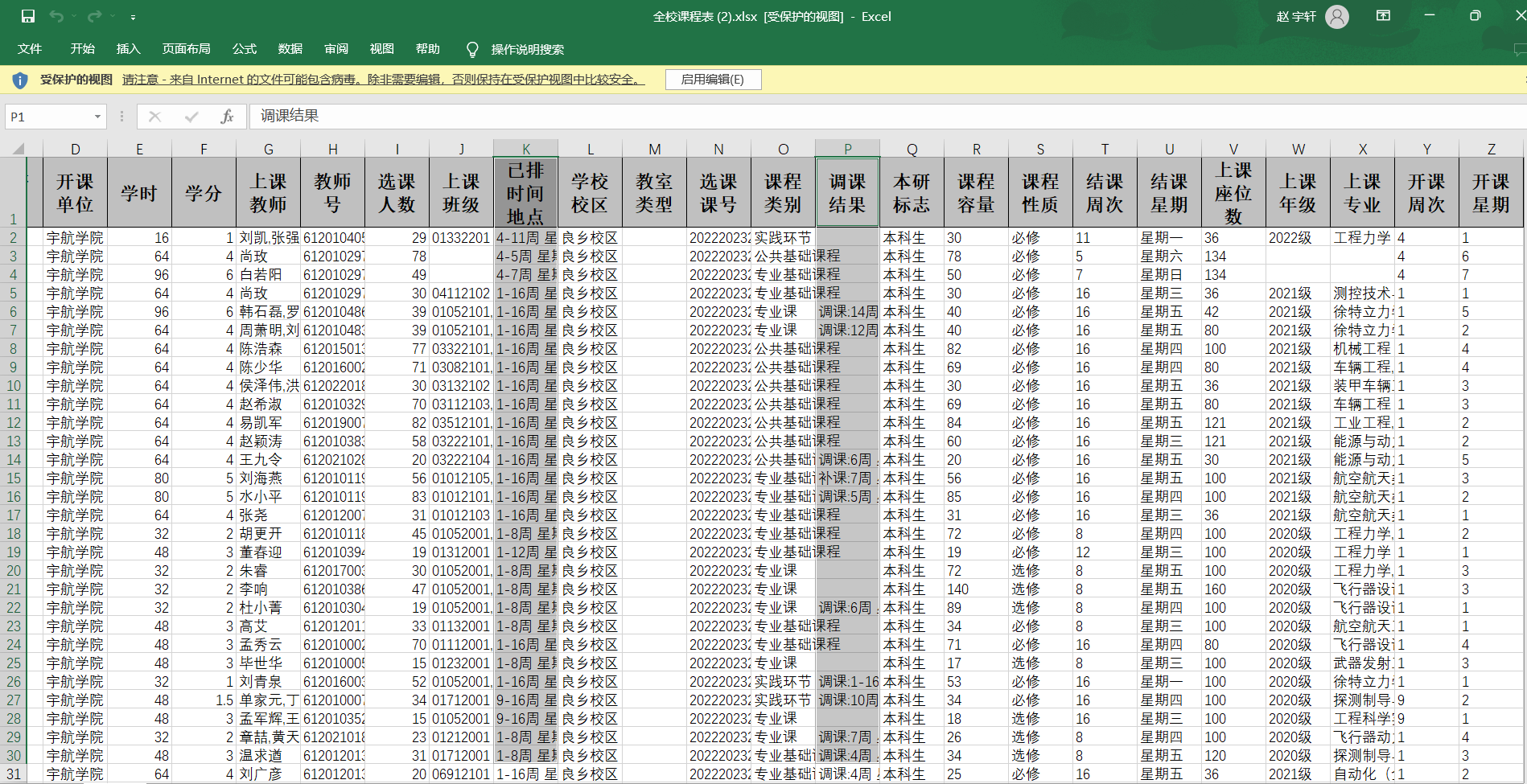
作为一名普通的学生我目前没有发现能让我的app直接连上全校课程信息数据库的途径，于是我采用以下方法作为代替：

在学校的本硕博一体化系统，我可以在“全校课表查询”——“教室课程表”里看到每间教室的排课情况，但是页面提供的“下载”功能只会显示一张jpg格式的图片，如下：

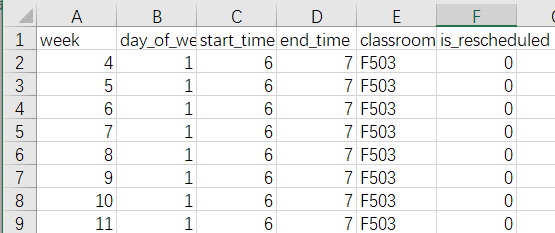


我暂时找不到能将该jpg快速转化为csv等可读取文本的文件类型的有效方法。这条路不可以，于是我采取其它的办法，发现在“全校课表查询”——“全校课程表”中，筛选条件为“学校校区==良乡校区 && 教学楼==文萃楼”，它可以导出我的搜索结果，并让我得到了一张xlsx表，在这个xlsx中，一共有26列，我这里只取两个属性列，即“已派时间地点”和“调课结果”，其余24列对我的程序来说并不需要，再把xlsx文件转为csv文件。（见下面的图片）

既然系统在“教室课程表”只会导出图片，而“全校课程表”会导出xlsx，我猜测学校将所有课程信息都存在了一个数据库，在我导出时，这个数据库我选取满足条件的数据并生成xlsx表给用户。而教室信息表应该是根据这个数据库的数据形成的可视化界面。



为了配合我的软件功能，我将Courses.csv通过python进行了一定预处理，预处理的程序可见提交得文件夹dataProcess中的data-processing.py。我这里说一下这段代码的逻辑：

 例如上表中，第一条记录的课程信息为：4-11周 星期一 6-7节 文萃楼F503，我期望最终的处理结果如下：

这里它会转化成8条记录，每条记录对应一次课程信息。例如第一条记录就表示第4周，周一，开始于第6节课，终止于第7节课，文萃楼F503有课，至于最后一列is\_rescheduled默认为0，它的引入是因为还存在调课，补课，停课的情况，我需要这个变量来讨论这三种特殊情况。可以查看Courses.csv “调课结果”一栏，对于不同情况处理逻辑不同：

第一种：“调课结果”一栏为空，此时无需处理

第二种：调课，例如在第14行，它的格式为“调课:6周 星期5 3-4节 王九令/6120210287 文萃楼F201 调至:14周 星期3 8-9节 王九令/6120210287 文萃楼F303”，此时我会把之前的记录6 5 3 4 F201的is\_reschduled值设为1，再添加一个新的记录表示14周 星期3 8-9节 王九令/6120210287 文萃楼F303有课（即 14 3 8 9 F303 0）

第三种：补课，例如在第40行，它的格式为“补课:9周 星期5 8-10节 文萃楼M222”，此时我直接新增加一条记录9 5 8 10 M222 0即可。

第四种：停课，例如在第64行，它的格式为“停课:10周 星期3 3-5节葛翔宇/6120190144 文萃楼B140”，此时我需要将原来的记录10 3 3 5 B140 0的is\_reschduled值改为1

以上是我的处理逻辑，经过如此处理后，查询空教室便显得非常简单。因为我可以很轻松地得知用户当前的日期以及时间段，我在SQL语句中通过week，day\_of\_week，start\_time，且is\_reschduled值为0（为1的话要么被调课了，要么停课了）的记录，查询用户当前时间段有课的所有教室（即classroom字段），然后再用全部教室减去这个查询结果即可。我另外自己手写了一个数据库ClassroomType.db，记录全部教室内容，同时增加属性type，表示教室是哪种类型（这里我以数字1，2，3分别代表小教室【20-30人】，阶梯教室和大教室【100人】，这样我先在Courses.csv转换后的Courses.db搜索到有课的教室【SELECT classroom …】获得字符串集合List，然后再在ClassroomType.db搜索到NOT IN List，这样就查到了所有空教室的记录。

在这里建议打开Courses.csv来欣赏这个表格多么难处理：（由于这个处理用到大量正则表达式，我使用相对比较熟悉的python来写）

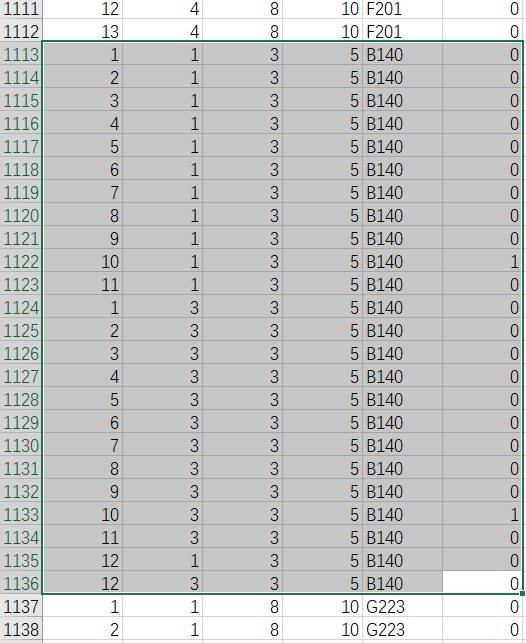
首先，一个格中会有多次课的信息，例如第五行“已派时间地点”为：1-16周 星期一 6-7节 文萃楼I703,1-16周 星期三 1-2节 文萃楼I703，这里一个格有两条记录，我需要先将它们用split函数将它分隔成两个字符串后再分别处理。另外，在调课结果属性列中，会出现调课停课补课三种情况混合的记录，不过无伤大雅，分隔后按照各自的处理逻辑即可。

正当我以为格式也就如上边那样统一时，我遇到了令人头疼的格式不一致的问题：例如在第1240行，记录是“1-6周 星期三[8-10节]文萃楼B223;8-11周 星期三[8-10节]文萃楼B223;12周 星期三[8-9节]文萃楼B223”，这里把时间[8-10节]加上了中括号，另外居然还改成了用；分隔。（之前都是逗号分隔，要不是我在正则表达式匹配前输出句子内容，匹配成功后输出“成功匹配”作为调试的话，我都注意不到！）我不知道相关的工作人员咋整的，估计是快到学期末了，在这块疏忽，忘了之前的规矩。（这些格式不统一的问题集中在后边的记录，显然Courses.csv中越往后的记录应该是这个学期越晚增加的记录）

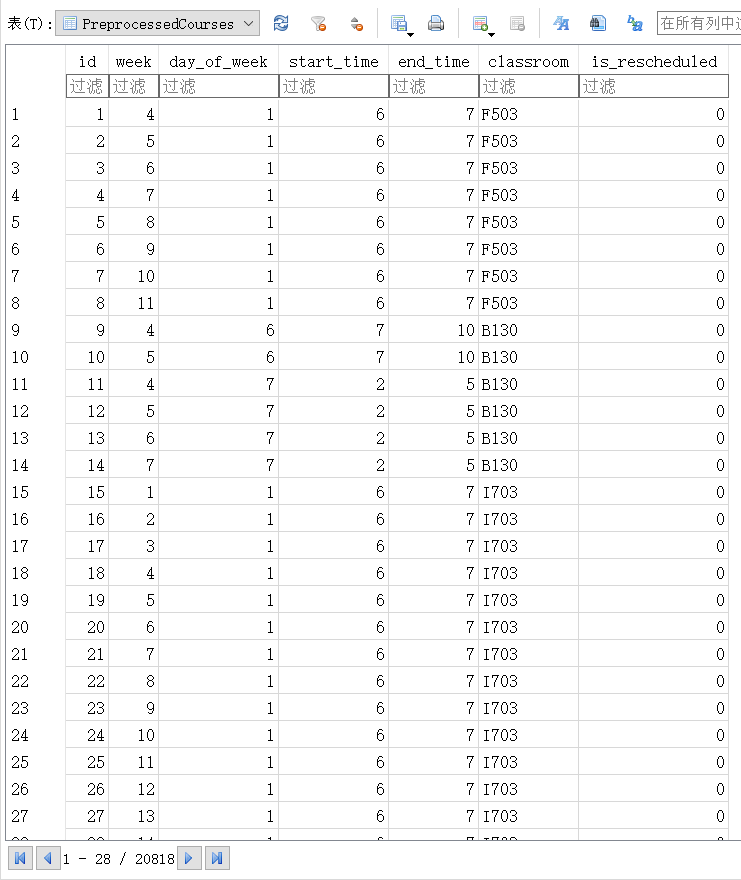
再比如第999行的记录：1-15周(单) 星期二 3-4节 文萃楼I503,2-16周(双) 星期二 3-4节 文萃楼I503,2-16周(双) 星期四 3-4节 文萃楼I503，这里还有单数周和双数周的情况！我这里还需要把1-15周（单）转化成第一个属性为1，3，5，..，15的记录。

以上的问题我都解决了，唯一一点遗憾的是我不会处理例如1286行“2周,4-6周,8周 星期五 3-4节 文萃楼F602,3周 星期五 3-5节 文萃楼F602”的记录，我之前说过，不同信息会用逗号或者分号相隔，但是这里它“2周,4-6周,8周”已经用逗号分隔，分开后会依次处理“2周”，“4-6周”，“8周 星期五 3-4节 文萃楼F602,3周 星期五 3-5节 文萃楼F602”，我肯定期望“2周”，“4-6周”也能读取到星期五，文萃楼F602的信息，我之前说我认为courses.csv应该是学校使用的数据库一部分，但我现在认为它可能也是原来数据库进行一部分处理后的结果，比如原数据库有2 5 3 4 F602 0和8 5 3 4 F602 0，在处理后合并为2，8周星期五 3-4节 文萃楼F602。另外，对于那些非课程而占用的教室（比如xx班班会，xx社团活动，xx班自习等等），在本硕博一体化系统上导出的表是不会显示的，因为它系统已经说了只查询全校课程。但是在i北理查询空闲教室的功能中，是可以看到那些教室是被占用的。所以，i北理用到的那个数据库还是另外一个，这个数据库我找不到，本硕博一体化导出的应该也是经过一定处理后得到的。

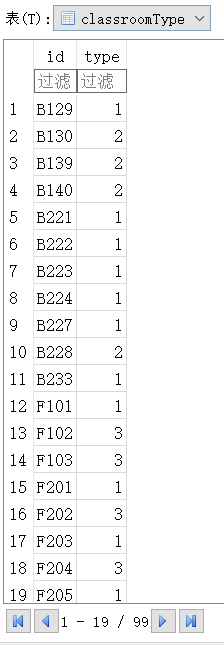
展示一下最终的效果：例如Courses.csv的第61行：1-11周 星期一 3-5节 文萃楼B140,1-11周 星期三 3-5节 文萃楼B140，调课结果是：停课:10周 星期3 3-5节葛翔宇/6120190144 文萃楼B140,补课:12周 星期1 3-5节 文萃楼B140,补课:12周 星期3 3-5节 文萃楼B140,停课:10周 星期1 3-5节葛翔宇/6120190144 文萃楼B140，我处理后的数据为：



5.17更新：在将csv变成db文件并导入app时，发现程序报错数据类必须要有主键，而这里我六个属性都不能当主键，于是我新增了一列属性id作为主键，它的值根据记录位置递增，最终程序读到的数据库为以下：这个数据库有2w+条数据，大小为908KB



在得知用户的当前时间后，很容易生成动态SQL语句在这个数据库中查询出有课的教室，我再用全部教室减去查询结果即可，我在这里又增加了一个数据库ClassroomType.db，它除了实现“减去的功能”（用SQL的NOT IN实现排除）的同时，还可记录教室的类型，让查询结果不再只有名称那么单调。ClassroomType.db见下：



Type=1

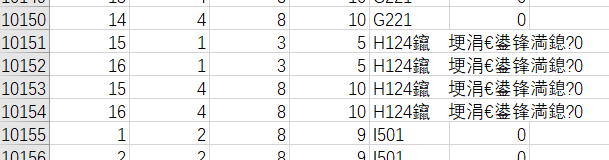
Type=2



Type=3

Type的属性值含义见右。

5.18更新：在后续测试中，发现有乱码产生，查找生成的csv文件发现以下：



这里的H124表示的是文萃楼H的机房之一，乱码应该对应的是机房。由于我在ClassroomType.db没有文H的机房，于是不用处理它。（我没有记录的原因是文H的机房非上课时间段会锁门，同时我不认为有人会选择去机房自习）

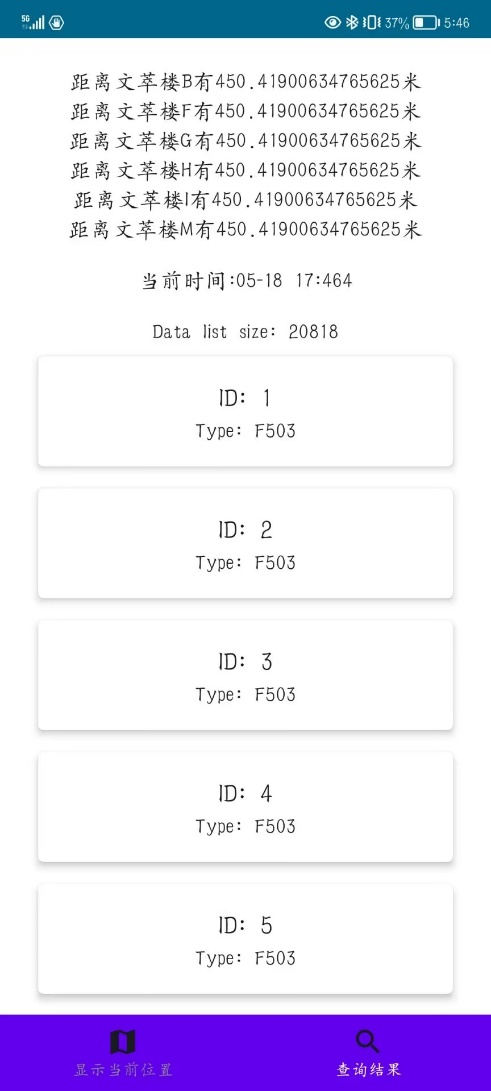
（2）导入高德地图以及实现按距离推荐

导入高德地图api有些麻烦，我首先要申请key，它要用的SHA1码我找了半天。另外，官网文档没有Compose框架的实现。我借助gpt的帮助逐步在Compose框架中实现。但还是遇到了一些问题：

1.获取用户当前位置的蓝点：我原本以为创建高德地图后就能自己显示，但是并没有显示。我以为是导入的过程中哪段代码漏了，但最后才发现没有向设备申请获取位置信息访问权限。。

2.获取当前位置与六栋楼的距离：我最开始是使用文档里的方法，输入终点名称的字符串，

最后查询与它的距离，但出现了以下情况：



这里我不知道为什么总显示六个距离相同，而且我不知道为什么会距离450米（我自己在学校走了走，发现这个点在南校区），这个bug迟迟不能解决。但后来我发现不用查询地点的名称，而是输入它的经纬度坐标：

val destinationLatLngB = LatLng(39.732275,116.174499)  
*updateDistanceWithLatLng*(currentLocation, destinationLatLngB) **{** distance **->** distanceViewModel.updateDistanceB(distance)  
**}**

例如文萃楼B的坐标是39.732275N，116.174499E,我通过这个方法就能更新distanceB，同时这个方法不需要询问网络（当前位置的经纬度也已知，直接两点距离算一下就是），运行速度更快。

#### 5 简要开发过程

（文档很多内容都是随我开发过程写的）

4月16号 确定app选题

4月17号 明确整个app的框架，涉及主要功能/界面交互的草图

4月19号 申请了高德地图的api接口，导入依赖（找个SHA1码找了一小时。。）

4月28号 寻找可以在compose框架下直接创建导航地图的api（高德地图官网只给了对传统基于View的文档，但是GitHub上有对谷歌地图的Compose框架支持库,我在Github上搜了几个人自己编写的库，但它们只支持地图创建，对于其他功能没有代码示例），我暂定从最初设想的Compose框架转为View

4月30号 突然发现compose框架中可以使用View，即通过AndroidView框架创建。我换回Compose框架并用这个控件生成了地图。在我自己手机跑程序时，发现无法显示用户定位出现的蓝点，通过查文档/询问gpt后发现没有申请手机的访问位置信息权限，

5月2号 完成高德地图的显示，用户当前位置的的蓝点显示

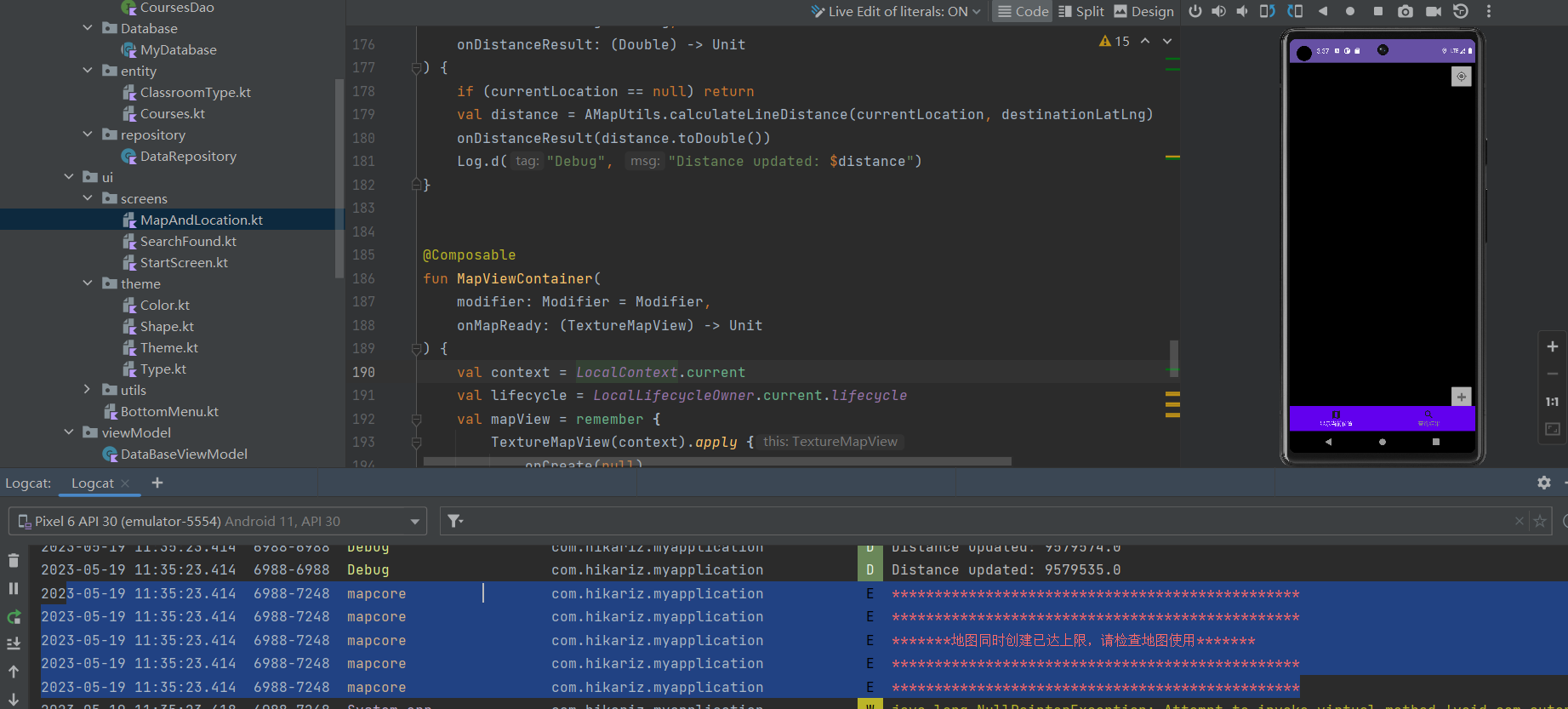
5月14号 完成查询当前位置与某楼的距离，完成教室类型数据库的读取，并对每条数据进行了ui装饰

5月15号 完成课程信息的数据预处理（几种情况讨论真麻烦。。）

5月16号 发现六个楼距离相同的问题，修复这个bug

5月18号 对查询结果进行

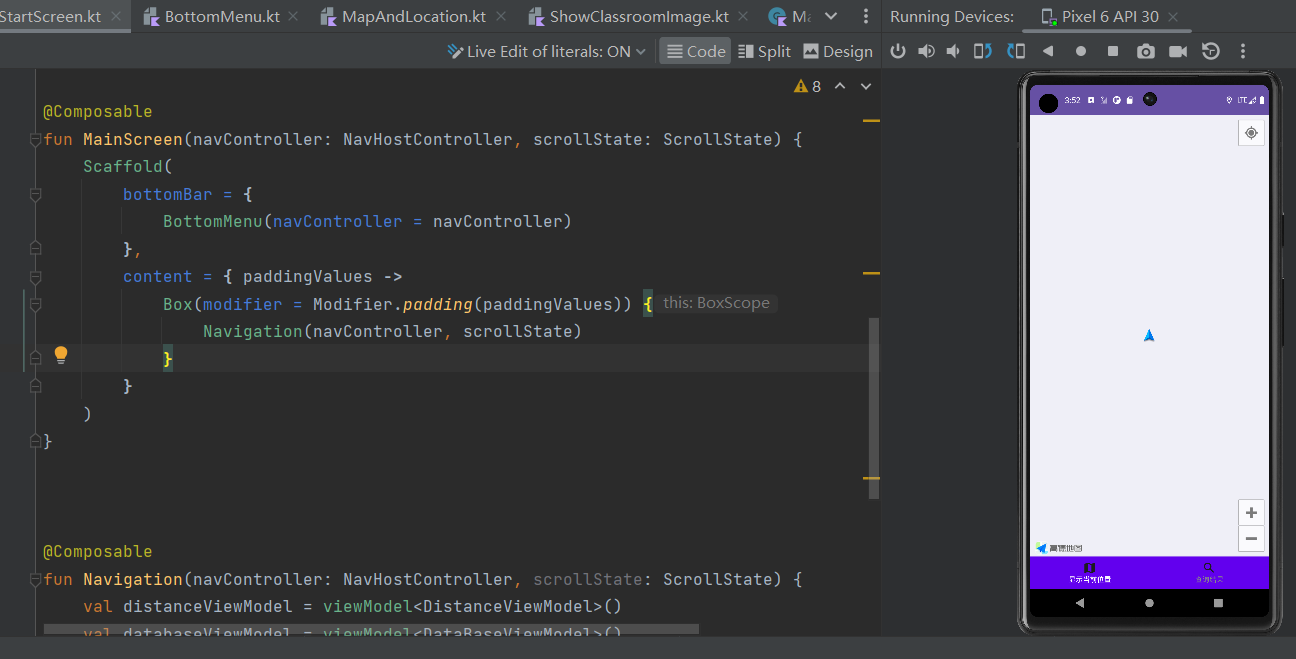
5月19号 修复一个bug：在多次于两个界面切换时（差不多切到创建高德地图页面10次时会显示黑屏），翻看日志得知如下，可见是没有及时销毁之前创建的地图



5月20号 修复一个bug：如下，底部导航栏会覆盖页面内容（比如这里-号就被覆盖）



修改方法：Scaffold空间的content属性做如下修改



5月21号 整理文档（很多地方在开发过程中已经详细记录了）

#### 6 学习感悟及对本课程的建议（可选）

经过一个学期的理论与技术学习和亲自动手开发，你有何感悟？对本课程的教学有何建议？

gpt4对我本次开发的帮助十分巨大。首先代码纠错Android Studio并不能跟上学期Java课用的IntelliJ那样能自动改错，其次经常会出现代码没错但是程序就会闪退的情况。虽然日志能记录错误信息，但是并不是每条报错信息都能让我迅速明白错在哪（例如导入的数据库中的数据必须要有主键，数据空还是非空必须要与db文件完全一致）。对于我不知道怎么修改时，我就把日志错误信息拷贝并复制可能出错的代码，交给chatgpt，在“两人”合作下，一个个bug也逐渐被攻克。

另外我不得不感叹一句Compose框架真好用。我记得有一次课Compose框架里的LazyColumn，用传统View实现就十分麻烦。除了简洁外，我也十分喜欢Compose框架的代码结构。最后一次课说有些东西不能用Compose框架，只能用View，我在尝试导入高德地图的最开始以为这就是不能用Compose的一种，高德地图官网文档只有在View中实现的。但是在github上，我看到Google官方对于Compose框架中应用谷歌地图的支持。所幸我后来使用AndroidView控件等也实现了，希望Compose框架生态能逐步完善，国内几家地图的开发人员能添加对Compose框架的文档。