## 2022年夏季Java小学期大作业

## 实验报告

李骋昊 2021010826

1. **代码结构**

代码主要分为四个部分。

1. 数据类型定义。（/data 文件夹）
2. 数据管理与处理。（/service 文件夹）
3. 界面显示。（/ui文件夹）
4. 全局操作。（MainActivity 以及 MyApplication）。
5. **具体实现**
   1. 数据类型定义。

News，表示一篇新闻。其中包括了其ID（对应被打开的次序）、API接口中给的ID（以检查其唯一性）、标题、发布者、时间、正文内容、其中包含的视频和图片的URL。

NewsResponse和NewsResponse2对应API发来的数据可能的两种格式（image为一个字符串或者数组）。

User类存储了用户选中的分类（以UserPreserence这一枚举类表示）。

* 1. 数据管理与处理。

主要包括以下方面：

SQLite数据库使用。这方面使用了DBManager和MySQLiteOpenHelper两个类。用于在打开应用时加载之前本地保存的新闻，并将阅读过的新闻保存到本地的数据库。实现的重点：使用SQLite数据库的beginTransaction 和 endTransaction， 保证数据的完整性。同时每次添加时只处理数据的新增部分，以减少SQL语句的调用次数。

从提供的API中获取数据。在FetchFromAPIManager中，根据参数（如时间，关键词，分类，以及分页计数）生成URL，利用OkHttp库获取API返回的值，并利用GSON库对其进行解析，得到News类型的对象。

管理News对象。在NewsManager类中，对于已经被打开过的新闻，保存ID与新闻之间，ID与API提供的ID之间的映射。记录历史记录，收藏记录。并在运行时将其与数据库中的数据同步。

加载图片。在PictureLoader中，包装了开源的Glide类，以显示图片，并且借助Glide实现显示图片的缓存。

多线程时回调。TaskRunner类封装了将任务A交给子线程，伺其有返回值，再行交由任务B在UI线程执行的毁掉过程。用以在尝试从API获取数据等耗费可观时间的操作时保正UI不卡死。(参考2022年计算机系暑培的开源代码。)

* 1. 界面显示。

整个界面分成四个部分。上部是分类切换。中部是新闻列表、新闻正文或者是搜索界面。下部是导航栏。此外还有从侧方弹入的用于增删分类的控件。其中，除下部以外用Fragment来管理。

上部的分类切换在TablistFragment 中实现。其中，实现的难点是其中的数据和MainActivity的通信：在Fragment中提供接口，并通过重载onAttach函数，将这一接口的一个实例作为Fragment的成员，从而在Fragment的业务代码中调用。而MainActivity 实现这一接口，在Fragment中更新了列表后，调用MainActivity中的重载方法，进而在其中直接调用中部的Fragment的相应方法。

中部用于显示主要内容（新闻列表、搜索结果、历史记录列表、收藏记录列表、搜索界面等）。由于具体的功能上有一定差异，用NewsListFragment和RecordListFragment两种Fragment来实现。这两个Fragment复用了NewsListAdapter来具体加载数据，以保持一些行为的一致：包括对于已经读过的新闻，其标题以灰色显示（这里有一个难点：由于使用了RecyclerView, 其中的View被循环使用，直接对TextView 设定颜色会导致复用了这个view的其他新闻标题也表现为灰色。通过显示带有颜色标签的Html内容的方式解决了这一问题），以及对于有0张、1张，2张以上图片的新闻，自动适配不同的标题布局。同时，对NewsListFragment中的RecyclerView，配置了EndlessRecyclerViewScrollListener(来自2022年计算机系暑培的开源代码)，以实现对上拉，下拉行为的监听和功能实现。

下部为一个NavigationView，以实现在不同的界面之间的切换。具体实现为，在点击对应分类后，在MainActivity中的对应函数中判断点击了哪一个，并进而调用对应的FragmentManager的方法，以替换Fragment，并实现对于返回栈的支持（参考2022年计算机系暑培的开源代码）。

侧方弹入的控件：TablistFragment上左侧的控件在被点击时将打开SelectPaddleFragment，以提供分类列表的增删。其中实现的难点在于其中对应的Adapter和Fragment之间的通信。同样利用回调接口实现。

* 1. 全局操作

MainActivity中初始化了以上提到的Fragment，并作为Fragment之间通信的中转（在上文已经述及）。

MyApplication中初始化了一些数据管理与处理的类（如数据库相关和User类），并提供了应用程序级别的全局的context，方便在Fragment等不妨便获取Activity的context时使用。

1. **总结与心得**

本工作参考了一些开源资料（主要是2022年计算机系暑培的相关内容），实现了在作业文档中要求的功能和性能。

心得：我采用了类似于“迭代开发”的策略来安排这两周的时间，即先完成一些小的模块（如新闻列表、或者增删分类的面板）。在这些小的模块实现后，对于其依赖的其他模块，先给一个最基础的模拟（如只返回一条固定的新闻的API处理），以提供一个基础的环境用于测试。这样的开发过程中，我可以较容易的确定BUG的位置（重点关心刚修改过或完成的模块），从而节省时间。

建议：提供更多更明确的参考文档，同时增加助教在机房的时间，以方便同学当面答疑。