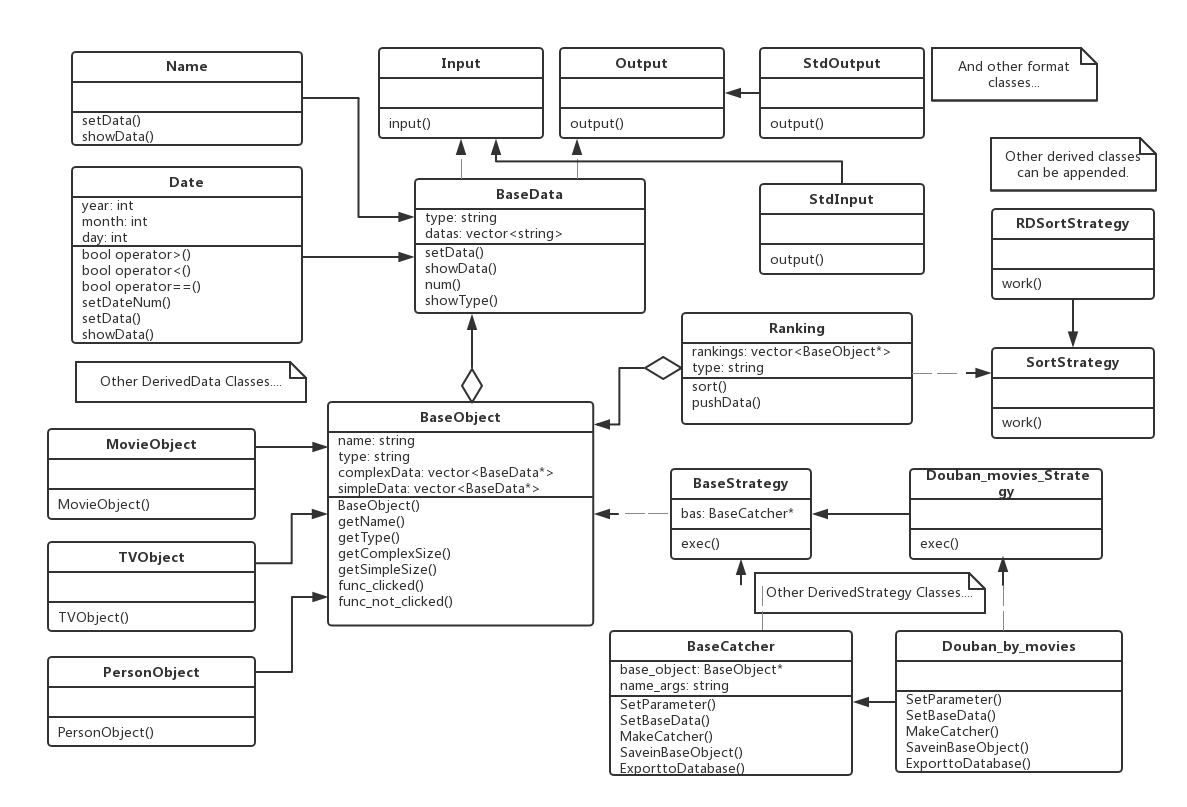
大作业说明文档

名称：电影信息收集系统

作者（字母序）：谭嘉睿，张鸣昊，钟闰鑫

1. 整体思路：本程序希望实现一个类似搜索引擎的收集系统，即在线即时爬取需要的电影信息，如猫眼电影在购票时，实际上就是收集各个影院电影信息的过程，本程序的思想类似，根据用户需求，借助交互界面，实现爬取不同电影网站信息的功能。
2. 功能解析：基本可以分为两大块：在线爬取、离线调库。用户既可以在联网时直接以搜索的形式查询信息，也可以查询现有数据缓存中有无需要的电影信息，如果有便可以直接本地获取。这样的设计，既可以即时收集最新的全面的信息，也可以在网络不佳或有省时需求时离线查询。
3. 文件框架：UML图如下：



（一）事物基类 BaseObject

派生类：电影类 MovieObject/电视剧类 TVObject/人物类 PersonObject

责任：管理、储存不同种类的Data，在Qt中展示一个界面；Data 分为 ComplexData 和 SimpleData，分别对应可以点击跳转页面的数据和不可进一步探索的数据。

（二）爬虫基类 BaseCatcher

派生类：若干。根据网站：IMDB/烂番茄/豆瓣 ；类型：Movie/TV/Person

来适配派生爬虫。

责任：爬取数据并传输

（三）策略基类 BaseStrategy

派生类：若干。根据网站：IMDB/烂番茄/豆瓣 ；类型：Movie/TV/Person

来适配派生策略。

责任：根据不同类型网站/事物，调用相关联的Catcher，并将Catcher返回的原始数据进行标准化处理，得到Data对象

（四）数据基类 BaseData

派生类：若干。包括名称 Name/日期 Date/ …

责任: 将数据高度统一地存储。

（五）排行榜类 Ranking

责任：组合若干 Object，并按照某种规则进行排序。排序用到策略基类 SortStrategy ，这里我们给出一种派生类：RDStrategy，按照上映时间进行排序，后续如果需要其他方法可以进行拓展。

（六）Format 系列

基类：Input/Output 派生类：若干。

责任：定义标准的格式，用于不同种类 Data 的存储。

可以看到，框架主要分为:BaseObject事物基类（派生出电影、电视剧、人物），BaseCacther基类（派生出适配不同网站不同目标的爬虫派生类），Basestrategy基类（派生出储存数据的各种策略），BaseData基类（派生出各种数据，与Format（Input）相互搭配以统一格式），以及一个Ranking类，来使用不同的sortStrategy以实现按不同方式排名。BaseObject里包含一个BaseCatcher，用来爬取信息；又包含一个BaseStrategy，用来储存信息用于将信息储存进BaseData中，这里的BaseObject用vector（complexData）储存，其中一个vector储存可以在UI界面上点击的进一步爬取信息的数据，另一个vector（simpleData）储存不可点击、只用于显示的数据。Ranking里包含BaseObject的vector，来储存参与排行的电影、电视剧等。

4. 工作流程：主程序在Qt中，生成所需类型的派生类对象Object，并配置相关的Strategy。

在Object对象构造时，使用Catcher对象，调用成员方法makeCatcher进行原始数据的爬取，再通过Strategy类进一步解析和存储，将得到的Data 对象导入Object类中，再辅以 Ranking类等实现其它功能。

5.分工：

张鸣昊完成了BaseCatcher部分和Strategy部分的算法，谭嘉睿完成了BaseObject部分，钟闰鑫完成了BaseData部分和Qt制作。

后期Debug由三人共同完成，小组分工明确，进展顺利。

github网站

<https://github.com/Triker-Monell/oop>

编译相关请见github网站上的readme