

**信息与软件工程学院**

学 号： 2013220402019

姓 名： 陈秋远

专业方向： 大型主机

企业名称： 华西期货

实习岗位名称： 金融软件开发工程师

企业指导教师： 杜野

院内代管教师： 刘峤

**企业实习中期报告**

目 录

[1. 企业实习的进展情况 1](#_Toc449422068)

[1.1. 实习工作完成情况 1](#_Toc449422069)

[1.2. 知识技能学习情况 1](#_Toc449422070)

[1.3. 职业素养学习培养 1](#_Toc449422071)

[1.4. 工程协作与交流情况 1](#_Toc449422072)

[2. 存在问题、应对措施与后续解决方案 1](#_Toc449422073)

[2.1. 存在的主要问题与应对措施 1](#_Toc449422074)

[2.2. 针对问题的后续解决方案与可行性研究 2](#_Toc449422075)

[3. 前期任务完成度与后续实施计划 2](#_Toc449422076)

[参考文献 2](#_Toc449422077)

**说明:**

1. **报告要求6000字以上。**
2. **本模板仅为内容基本参考，请各位同学根据个人情况进行目录结构扩展。**
3. **封面和目录单面打印，从正文第1页开始双面打印。**

## 企业实习的进展情况

### 实习工作完成情况

（详细介绍已完成的工作情况，包括对复杂软件工程问题的推理分析,针对复杂软件工程问题设计满足特定需求的总体设计和详细设计。对于已经完成工作，需对实施结果进行分析和解释，并通过信息综合得到合理有效的结论。）

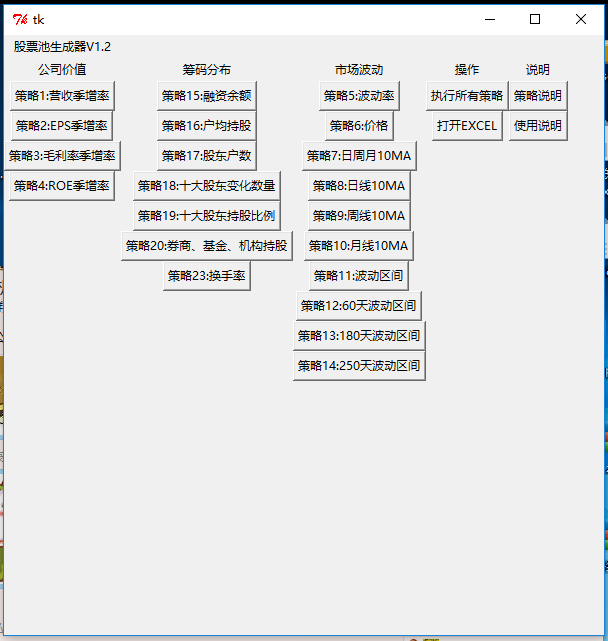
在前期的工作之后，有了一个框架。之后我面临的是更多的需求，在1.0版本我注重的是实现一个功能，所以代码是面向过程的，但是可拓展性差了很多，所以我对之前两千行代码进行了重构，采用的是MVC的方式迭代出2.0版本，在这期间还陆续加了很多指标，工作量大概翻了一倍。之前的提取数据、存储数据、处理数据、输出数据的所有功能都放在一个代码文本里面，在这个代码里面耦合性太高，冗余的代码也多，很多注释也不完整，于是花了很多的功夫来重构，将之前的文件按照数据模型（M）、视图（V）、控制（C）分成了三个模块，用面向对象的方法来重新写了一遍。

在做这件事情的时候我意识到，写一套代码需要从一开始就有很好的框架，不然前面的一个偷懒在后期会有几何数量上升的工作量。

新代码要有新气象，我之前做的非常简陋的UI也被公司的人吐槽了，所以我顺便重做了UI，但是我再一次低估了一个看似简单的“界面”的复杂性，在1.0版本我的UI就是一个对话框加几个按钮，但是新的版本需要更好的布局，更复杂的按钮、输入框以及动态调整大小，还要能够显示帮助文档，于是我放弃了1.0版本所使用的python自带的tk库，转而从头学习pyqt库，pyqt库是从C++的qt库移植过来的，内容驳杂，我主要是需要一个“好看能用”的界面就行了，但是还是在动态调整方面费了一些功夫。在这个过程中我还接触到一个问题，那就是界面和后端的异步问题，在线程交给后端运行的时候界面就会卡死。

这时候我才明白，到这个时候就体现出web技术的优越性了，web的前后端分离的思想几乎不会存在这个问题，而我如果要解决这个问题的话还要进行多线程编程，大大提高了复杂度。

然后重构之后添加了更多的指标和各种各样的说明，然后将性能优化了一下，主要是网络连接方面的，这样就完成了2.0。之后陆续有了更多的功能，将版本陆续迭代到了2.2



**第一版UI**



**第一版UI和第二版UI的对比**

在将版本2交付之后又有了新的任务，就是将现有的数据可视化，这里我就使用了python的一个名为matplotlib的库，这个库有很棒的视觉呈现。最开始它的一些东西是嵌入到tk库里面的，但是考虑到我用的是matplotlib，所以我进而参考了一本《Matplotlib for Python Developers》，将matplotlib嵌入到了qt中，然后增加了一些控件。之后碰到的问题就是画出K线图，用python直接画K线图会有一个问题，那就是会出现日期的间隔，这样就会非常丑，但是他的横坐标又比较麻烦，所以我参考了很多地方的代码，最后发现了他本身其实可以写一个回调函数来将横坐标和日期一一对应，这样就成功解决了这个说大不大但是非常麻烦的问题。之后就是熟悉这个库和qt库的交互了，matplotlib和qt的交互比较麻烦，但是都是一个熟悉的过程，在这个方面弄好之后就是将数据导出的工作。在这里我着重考虑到了软件开发时的低耦合的问题，所以完成的还是比较好，基本上没有出现太大的问题。画出来的图形形状还是不错的，也能够很好地体现一支股票的趋势和推荐的程度，得到了满意的回馈，但是存在的不足在于一些可以切换图表的功能还需要在这个软件中加以成型才能用于更多的应用当中。



**股票可视化界面**

交互完成之后就是网络的问题，因为一部分数据源在本机，但是这个软件需要放在其他电脑上来运行，所以需要远程连接本地数据库。开始我以为这是一个很复杂的问题，但是后来发现其实没有那么复杂，因为直接提供一个IP和端口号就能连接了，问题在于这只是内网的IP，而从外网连接到这台主机还需要端口映射，而这里用的是电信的网，外网IP都在运营商那里，所以很难搞定，但是能在内网中使用也将就了。

之后做的一件事情就是将选股选出来的股票进行一个统计，我使用的方法是直接在excel表中用openpyxl来读取然后在内存中运算统计之后交付到数据库中，很简单的一个工作，但是让展示出来的图更加饱满了。

在做完股票的东西之后，依葫芦画瓢，写了很多的胶水代码然后将期货的类似的功能也写出来了，之后将他们整合到现有的代码中，就迭代出了版本3.0



**期货的页面**

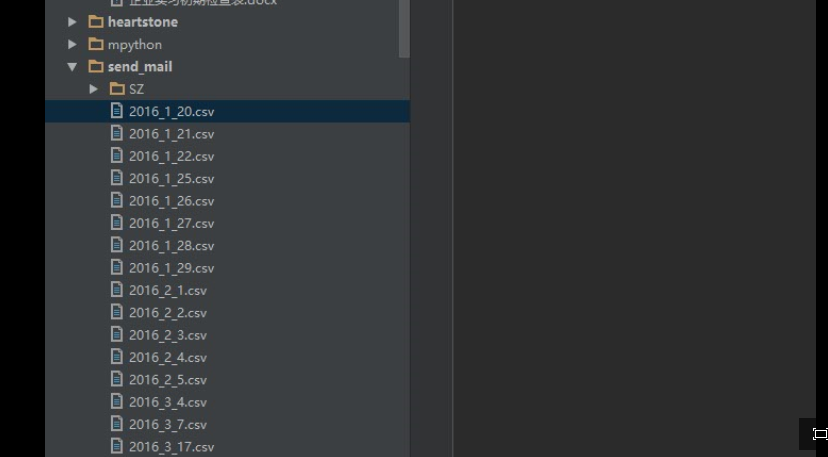
### 知识技能学习情况

（根据课题应用场景，选择开发环境、工具与技术标准；说明工具的学习和使用情况，文献和书籍的查阅、分析、总结和收获；阐述针对个人或职业发展的需求，进行的其它学习等。）

刚开始入职的时候在老师的建议下选择了python作为工具语言进行一系列的操作，具体使用的是python2.7.因为在python平台上的科学计算和网络支持。Python也有很多的库能够调用。在实习中大量使用了库，但是因为很多库的文档写的不是很全，所以解决问题的时候还是要读源码，在读源码的时候碰到了很多的困难，但是最终还是解决了。因为从事的是金融相关的工作，还需要很多金融和数学的知识，为此我读了很多金融相关的书籍，也在准备考取期货从业资格证。数学知识方面我在下班的时候重新复习了微积分和概率论，但是发现学校学习的数学知识并不够，所以还需要具备一些统计学知识，这些学习任务都在我下班之后进行。我希望以后从事量化交易的工作，所以在做好IT之余还是需要尽量掌握更多的金融知识。

另外还管理了一个大小约为4T的数据库，使用的sqlserver。在管理这个数据库的过程中我碰到的问题就是“太大”，在学校学习数据库的时候我们碰到的数据一般都是demo的数据，最多上百条，但是在这里碰到的数据直接就是几百万条，如果想要排序非常困难。就拿排序来说，当数据小的时候，随便在内存中就能排序了，到了这个量级，内存是不可能装下的，这时候就需要外排序了，更别说整个硬盘装不下的时候了。幸好我碰到的工作比较简单，只是将文件读取出来，但是这也是让我吃了很大的苦头，开始我以为直接写几行把一个表拉下来就好了，但是这样老是死机，我查看了它的出错日志才发现，原来是内存溢出。后来我改成按照股票和日期分别读取的办法，分别存取，这样就解决了这个问题。后来我了解到，sqlserver还能实现分页查询，但是这个功能我没用到。

另外学习到的知识是关于爬虫的知识，我没有遇到需要大规模爬虫的问题，但是在某些时候需要爬取一些特定的网站，开始碰到的几个网页非常简单，就是很简单的html，我写几个代码就爬取下来了。之后碰到一个比较麻烦的，是上证的官网，他显示的内容是JS渲染过后的，但是我用正常的爬取没有办法爬取到，后来在查询了各个地方的资料之后终于找到，使用了一个selenium的包，这个包相当于是一个人模拟浏览网页，可以直接暴力抓取渲染完毕后的网页，相当于是一些包括谷歌浏览器在内的元素审查功能，成功抓取到了需要的表格内容。



**用爬虫每天抓取的上证指数**

### 职业素养学习培养

（描述对软件工程系统的质量、环境、职业健康、安全和服务意识的学习和认识，对职业道德和规范的理解和遵守。）

在开发这个软件的过程中我意识到了，开发软件的时候需要一个良好的架构，这就是现有的一些框架比较流行的原因。如果没有良好的架构，那么就只能称之为代码而不能称之为软件，我在刚开始实习的时候只是以实现功能为目标，所以写的只是程序段，但是出现一个需求需要把这些功能整合在一起的时候就会让人特别难受，所以需要一开始就对整个软件有一个概念，在写的时候才能对这个软件有良好的把控。

另外一个认识就是数据量的不同对于人的挑战是不同的，当我写几十行数据的时候和写几百万行数据的时候是两个概念，需要考虑稳定性、性能效率等各个方面的，阿里巴巴的双十一需要顶级技术人才来解决不是没有道理的。这样的经历让我对大规模的程序有了更深的理解。

在工作闲暇之余我还用PHP写了一个自己的网站，网站的网址为https://chenqiuyuan.vipsinaapp.com/,虽然只是一个展示页面，但是还是花费了不小的功夫，后面会尽量增加功能的。

### 工程协作与交流情况

（阐述在实习执行过程中，针对特定的目标或问题，与工程项目组成员，包括与其他学科的成员合作并开展工作的情况。）

在公司实习的这段时间，我主要负责的工作是由我一个人完成的，但是在工作中碰到的很多技术问题我也在和公司中的同事沟通。这个软件涉及大量的金融知识，它被设计出来的目的也主要是为金融从业者服务，所以我需要对此有很多的了解，我在开发软件的过程中尽量抽空学习金融相关的知识，在这些方面我也和指导我的上司保持积极的问题，一般流程是他提出一个需求，然后我先初步考虑实现的难度，以及能够实现的程度，然后进行反馈，我的上司比较通情达理，能够在不好实现的地方做出改变，这就让开发工作得以顺利的进行而没有把更多的时间消耗在讨论中。当出现一些很难定义的问题的时候，我们也能很好地解决，我尽量将程序的思维转化为口语然后解释，同时从他金融方面的思维获得解决问题的思路，这样良好的互动让我能够更好地解决工作中的问题。

## 存在问题、应对措施与后续解决方案

### 存在的主要问题与应对措施

（分析、总结和归纳实习执行过程中存在的主要问题与具体的应对措施，能通过具体事例来说明在心理素质培养以及应对项目风险与挑战的能力锻炼等方面是否有所收获。）

在软件开发过程中碰到了很多的问题，首先就是一些关于金融数学的公式在实现的时候会遇到一些问题，而金融行业有较强的保密性，很多东西网上都找不到，需要自己去摸索解决，这时候就需要一些数学功底了。

第二个问题是软件本身的，我第一次系统写一个软件，经验不足，很多可重用的东西都没有封装好，代码的效率也不高，程序的结构也不是非常规范。

第三个问题是数据源的问题，现在用的数据源是直接从网上拉取下来，这样虽然短时间看非常方便，但是潜在的问题在于如果远程连接出现问题，那么整个软件都会面临崩溃。

最后一个问题在于现有的PC端软件几乎已经不能适应需求了，需要一个网络版的软件来进行使用，也方便进行身份认证、展示等各个功能。

### 针对问题的后续解决方案与可行性研究

（针对当前尚未解决的复杂工程问题，分析文献寻求解决方案，并在安全、环境、法律等约束条件下，通过技术经济评价对设计方案的可行性进行研究，说明方案的合理性。）

对第一个问题，在实习中我复习了很多的数学知识，明白了在学校中学习的很多的都只是基础而已，更多的实践还是要在实际工作中来解决。但是另一方面，工作中的应用都是非常浅显的，也就是说数学知识这种看似一般程序员用不到，但是在碰到的时候却是最难的一点，这就需要更多的厚积薄发。我在闲暇之余复习概率论和自学了我们专业没有学习的统计，这让我虽然比较累，但是学习了很多知识。

对于第二个问题我认为更多的是经验不足导致的，必须经历的一步就是多写，另一方面在同学的建议下也可以看一看其他人的开源代码，来学习构件一个软件的时候的种种经验。

第三个问题是个很复杂的问题，我至今都没有想到一个很好的办法，全部是很麻烦的事情，如果我建立一个数据库，首先是没有一个好的服务器，另外就是存入数据的脚本非常多，让我想想就觉得很难实现。然后就是存储格式的问题，有时候存进去容易但是维护以及提取就变成了很麻烦的事情，在这样的情况下需要对于一个数据库更深的理解才行。

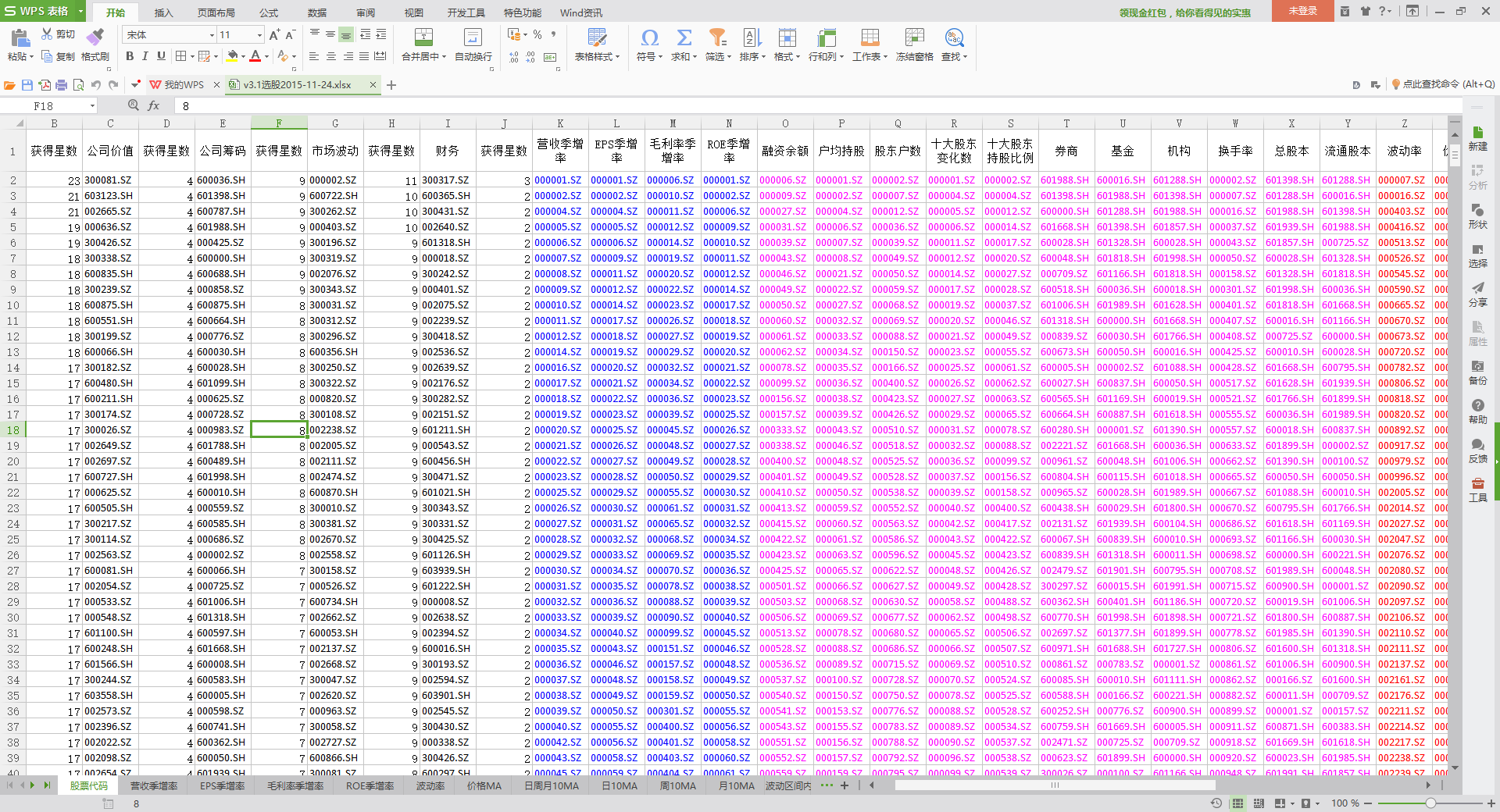
对于最后一个问题来说分为两个子问题，第一个子问题就是现有的软件在UI的地方需要多线程设计，这样才能在将功能交付给后台的时候前台不至于失去响应，这是一个比较麻烦的工作，但是我会尽量去做的。另外一个工作就是将现有的一切移植到web版本上，我想到的办法是用django来实现，虽然之前我有过用php写网页的经验，但是我不能保证php和python有很好的兼容性，所以利用django这个框架将现在的这一套体系全部搬到web版本是很好的一个选择。

## 前期任务完成度与后续实施计划

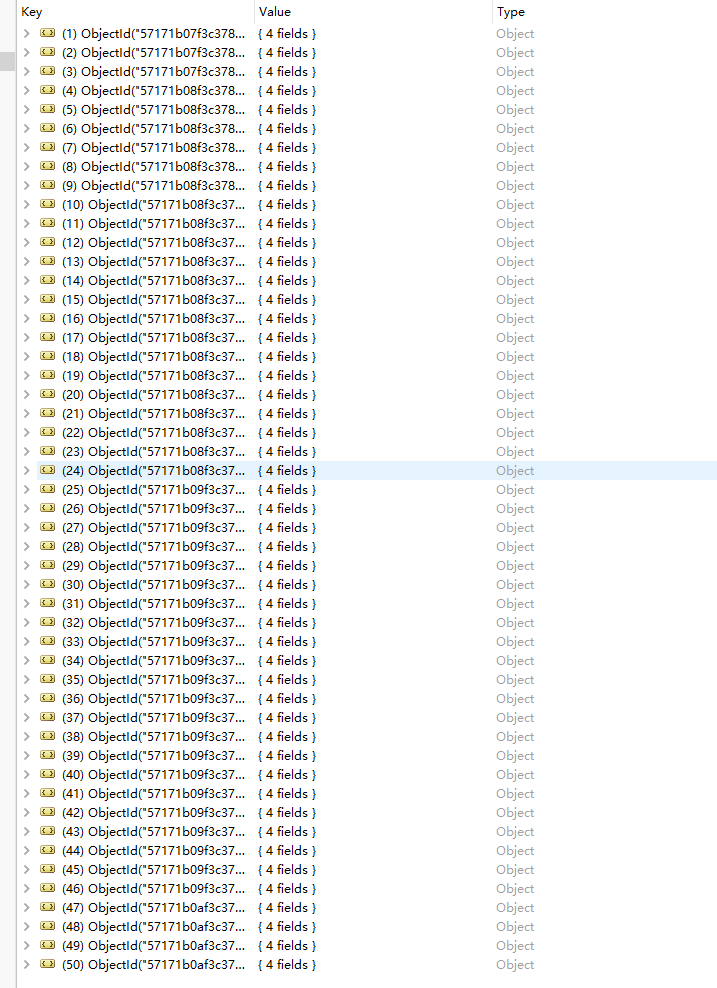
（前期任务完成度自我评价及后续工程环节的实施计划安排。除针对复杂工程问题的实施方案执行计划，还包括针对个人或职业发展需求的知识技能学习计划。）

前期基本上完成了第一个版本，版本号为1.0，经过几次调整迭代到1.2版本。大概的样子如图所示，界面只是很微小的工作，但是可以以此为窗口来展示所做的工作。首先是对于数据的整理，这里用到了万得的API，万得是一个金融数据的公司，提供了一个能够抓取数据的接口，通过这个接口能够拿到所需要的数据。我的任务就是完成上司的要求，获得相应的数据并进行相应的处理，然后以此来筛选股票。这是一项比较麻烦的工作，因为他涉及到从网络连接，数据抓取，数据存储到数据加工这一系列的过程，而且需要筛选的条件涉及到一些数学

计算。举个例子，一个EMA，在统计学中能找到一串公式，但是却不能找到相关的代码，更不要说现成的库，所以就需要自己来实现这个计算。而且既有的公式算出来的结果也与券商提供的数据不符，所以只有自己来将公式用代码实现，所幸这个公式很容易用递归的办法来完成，在调试多次后成功将数据算出。类似的例子还有很多，就不一一举例了。另一个比较麻烦的事情是数据的输出，因为公司的人不擅长python，所以需要将数据以EXCEL的形式输出，本来只是一些简单的数据的话会非常简单，但是要做成多sheet表，各个指标定制化格式就会显得比较复杂，所以我引用了一个python控制excel的openpyxl库，这个库基本上都是英文的，文档支持也不是很好，一些边边角角的功能都没有写清楚，所以我还需要去读它的源代码，举个小例子，要从python中改变excel文字的颜色的代码，库的源代码是预定义的一些颜色，而我找到了它放RGB的位置，然后加以修改就能通过RGB来画出我想要的颜色了,类似的例子还有很多，按下不表。与此同时，EXCEL其实是一个便于展示但是弱于处理的存储方法，也就是说从追求表现以及简单操作来说EXCEL就足够了（使用EXCEL可以方便地对某些行列排序之类的），MS的VBA也能有一些稍微高级的程序功能。但是从程序高效的角度来说，数据库才是更好的选择，所以我还是要考虑将这些数据存储到一个数据库中，我选择的数据库是mongodb，它是一个nosql类型的数据库，比较方便，我咨询了公司的前辈，数据大概每500万条是一个性能的门槛，我估计了一下我的数据量，大概就几百万的水平，所以暂时直接使用就好了，实际情况不大可能会碰到性能瓶颈的问题。至于搭建数据库的过程没有可以过多叙述的地方，虽然比较麻烦但是还是成功了，其中用到了python支持mongodb的pymong这个库，支持比较好，也没有多去查看源码。这些是我搭建这个平台的一些初期的工作，总共的代码量大约两千行。版本2和版本3的陈述在实习工作中已经陈述。



**每日生成的excel表，下面的sheet是详细数据**



**Mongodb中的数据**

在后续的工作中，还有许多待做的工作，比如将数据库存取到本地、制作出新的模式追踪图，现在还发现既有的代码依然有很多设计不合理的地方需要重构，我需要在后面的实习中更好地运用软件开发的知识来将这个结构设计的更加合理。

另外，传统软件的一些弊端包括并发性和网络联通性能已经在我实践的过程中暴露出来了，让我更深地认识到web技术是目前一个很先进的技术的原因，所以我还有将现在的软件移植到web上面去的想法。

关于既有的股票的回测也是一个问题，因为现在的状况是选出了股票但是没有对它们进行回测的手段，我需要写一些框架和一些基本的策略来测试一下效果。

## 参考文献

1. Openpyxl. http://openpyxl.readthedocs.io/en/default/
2. matplotlib. <http://matplotlib.org/>
3. 《机器学习实战》,Peter Harrington
4. 《统计学方法》， 李航
5. Pyqt4,https://sourceforge.net/projects/pyqt/
6. 我的github地址：github.com/chenqiuyuan