**开源软件基础大作业**

**NBA历年MVP表现变化趋势图**

**Trend chart of NBA MVP performance over the years**

学 生 姓 名： 周子康(组长)

学 号： 201892231

学 生 姓 名： 李修华

学 号： 201892498

学 生 姓 名： 吴宇嘉

学 号： 201892178

学 生 姓 名： 宋济廷

学 号： 201892417

完 成 日 期： 2021.1.9

大连理工大学

目 录

[1 项目背景以及意义 1](#_Toc59392650)

[2 项目创新点](#_Toc59392657) 2

[3 项目的设计](#_Toc59392681) 3

[4 如何实现](#_Toc59392688) 6

[5 测试](#_Toc59392692) 7

[6 项目分析](#_Toc59392688) 11

[7 不足以及可改进之处](#_Toc59392688) 12

[8 成员贡献](#_Toc59392688) 13

[9 成员感想](#_Toc59392688) 14

1 项目背景以及意义

项目背景：这是一个从http://www.stat-nba.com/award/item0.html这个网页上爬取数据，利用urllib获取网页数据，然后利用bs4的beautifulSoup进行数据的解析和获取想要的数据，再利用re正则表达式结合bs4得出想要的数据，利用sqlite3存储数据，把数据从数据库中取出来，然后利用matplotlib.pyplot进行可视化的一个项目。

生存了以下这些图：

NBA历年MVP得分变化趋势（折线图）

NBA历年MVP得分条形图

NBA历年MVP助攻变化趋势（折线图）

NBA历年MVP助攻条形图

NBA历年MVP篮板变化趋势（折线图）

NBA历年MVP篮板条形图

项目意义：一个怎么样的球员才能获得MVP，最好球队中的最好球员才有机会获此殊荣。每一个赛季都会产生一支总冠军球队，也就是说最少会有15个球员获得总冠军戒指，而MVP只有一个。换句话说总冠军决定了一个球员的高度，而MVP则是体现一个球员的宽度。MVP比总冠军更难得，拥有数量更多的总冠军能提升你的历史地位，而MVP是在评价球员伟大层面的指标。

纵观NBA历年MVP的得分，助攻，篮板，从一个横向的角度来分析MVP球员的总体水平，很好的体现了一个球员的宽度，有着非凡的意义。

总冠军是所有球员的追求，mvp是梦想。这代表着你是最顶尖的球员，最重要的球员。

MVP的实际意义就是说你很伟大。

2 项目创新点

一：用到了matplotlib进行可视化，matplotlib是python中很强大的一款绘图工具包，里面提供了类似于条形图、柱状图、折线图等一系列常见的图形的绘制方法。在绘图的时候，经常要对图中的相关数据进行标识以及添加图例、标题等。

二：利用课堂上讲过的sqlite3存储数据

SQLite是目前最流行的开源嵌入式数据库，和很多其他嵌入式存储引擎相比(NoSQL)，如BerkeleyDB、MemBASE等，SQLite可以很好的支持关系型数据库所具备的一些基本特征，如标准SQL语法、事务、数据表和索引等。事实上，尽管SQLite拥有诸多关系型数据库的基本特征，然而由于应用场景的不同，它们之间并没有更多的可比性。下面是SQLite的主要特征：

    1). 管理简单，甚至可以认为无需管理。  
    2). 操作方便，SQLite生成的数据库文件可以在各个平台无缝移植。  
    3). 可以非常方便的以多种形式嵌入到其他应用程序中，如静态库、动态库等。  
    4). 易于维护。

    综上所述，SQLite的主要优势在于灵巧、快速和可靠性高。SQLite的设计者们为了达到这一目标，在功能上作出了很多关键性的取舍，与此同时，也失去了一些对RDBMS关键性功能的支持，如高并发、细粒度访问控制(如行级锁)、丰富的内置函数、存储过程和复杂的SQL语句等。正是因为这些功能的牺牲才换来了简单，而简单又换来了高效性和高可靠性。

3 项目的设计

一：数据爬取部分

数据的爬取分为如下几个步骤：

1. 利用urllib获取网页数据

urllib.request可以用来发送request和获取request的结果：

包括urllib.request.urlopen()基本使用

urllib.request 模块提供了最基本的构造 HTTP 请求的方法，利用它可以模拟浏览器的一个请求发起过程，同时它还带有处理 authenticaton （授权验证）， redirections （重定向)， cookies (浏览器Cookies）以及其它内容。

可以发现，response = urllib.request.urlopen(request)我们依然是用 urlopen() 方法来发送这个请求， urlopen() 方法的参数是一个 Request ，通过构造这个这个数据结构，一方面我们可以将请求独立成一个对象，另一方面可配置参数更加丰富和灵活。

headers 参数是一个字典，你可以在构造 Request 时通过 headers 参数传递，也可以通过调用 Request 对象的 add\_header() 方法来添加请求头。请求头最常用的用法就是通过修改 User-Agent 来伪装浏览器，默认的 User-Agent 是 Python-urllib ，可以通过修改它来伪装浏览器，比如要伪装火狐浏览器，可以把它设置为 [Mozilla](https://www.baidu.com/s?wd=Mozilla&tn=24004469_oem_dg&rsv_dl=gh_pl_sl_csd" \t "https://blog.csdn.net/qq_43546676/article/details/_blank)/5.0 (X11; U; Linux i686)Gecko/20071127 Firefox/2.0.0.11  
 urllib.error包含了urllib.request产生的异常

 urllib.error可以接收有urllib.request产生的异常。urllib.error有两个方法，URLError和HTTPError。URLError是OSError的一个子类，HTTPError是URLError的一个子类，服务器上HTTP的响应会返回一个状态码，根据这个HTTP状态码，我们可以知道我们的访问是否成功。200状态码，表示请求成功，再比如常见的404错误等。

urllib.parse用来解析和处理URL

使用 urllib.parse.urlparse将url分为6个部分，返回一个包含6个字符串项目的元组：协议、位置、路径、参数、查询、片段

使用  urllib.parse.urlsplit 将url分为5个部分，返回一个包含字符串项目的元组：协议、位置、路径、查询、片段

urllib.parse.parse\_qs 返回字典

urllib.parse.parse\_qsl 返回列表

urllib.robotparse用来解析页面的robots.txt文件

1. 利用bs4的beautifulSoup进行数据的解析和获取想要的数据

[Beautiful Soup](http://www.crummy.com/software/BeautifulSoup/) 是一个可以从HTML或XML文件中提取数据的Python库.它能够通过你喜欢的转换器实现惯用的文档导航,查找,修改文档的方式.

1. 利用re正则表达式结合bs4得出想要的数

Re正则表达式**优点**：只要熟练应用正则表达式，而且匹配的目标是纯文本，那么相比于写分析器来说，正则可以更快速的完成工作。还有在捕获字符串的能力，正则也可以很好的完成工作，比如截取url的域名或者其他的等等

**缺点**：1.正则表达式只适合匹配文本字面，不适合匹配文本意义：像匹配url，email这种纯文本的字符就很好，但比如匹配多少范围到多少范围的数字，如果你这个范围很复杂的话用正则就很麻烦。或者匹配html，这个是很多人经常遇到的，写一个复杂匹配html的正则很麻烦，不如使用针对特定意义的处理器来处理（比如写语法分析器，dom分析器等）。

2.易引起性能问题：像.\*这种贪婪匹配符号很容易造成大量的回溯，性能有时候会有上百万倍的下降，编写好的正则表达式要对正则引擎执行方式有很清楚的理解才可以。

3.正则的替换功能较差：甚至没有基本的截取字符串或者把首字母改变大小写的功能，这对于url重写引擎有时候是致命的影响。

1. 利用sqlite3存储数据

SQLite是一个进程内的轻量级嵌入式数据库，它的数据库就是一个文件，实现了自给自足、无服务器、零配置的、事务性的SQL数据库引擎。它是一个零配置的数据库，这就体现出来SQLite与其他数据库的最大的区别：SQLite不需要在系统中配置，直接可以使用。且SQLite不是一个独立的进程，可以按应用程序需求进行静态或动态连接。SQLite可直接访问其存储文件。

缺点：1。并发访问的锁机制。SQLite在并发（包括多进程和多线程）读写方面的性能一直不太理想。数据库可能会被写操作独占，从而导致其它读写操作阻塞或出错。2。SQL标准支持不全。在它的官方网站上，具体列举了不支持哪些SQL92标准。不支持外键约束。3。网络文件系统（NFS）有时候需要访问其它机器上的SQLite数据库文件，就会把数据库文件放置到网络共享目录上。这时候你就要小心了。当SQLite文件放置于NFS时，在并发读写的情况下可能会出问题（比如数据损坏）。原因据说是由于某些NFS的文件锁实现上有Bug。

二：数据可视化部分

把所有数据从数据库中取出来

利用了matplotlib.pyplot

matplotlib.pyplot是一个有命令风格的函数集合，它看起来和MATLAB很相似。每一个pyplot函数都使一副图像做出些许改变，例如创建一幅图，在图中创建一个绘图区域，在绘图区域中添加一条线等等。在matplotlib.pyplot中，各种状态通过函数调用保存起来，以便于可以随时跟踪像当前图像和绘图区域这样的东西。绘图函数是直接作用于当前axes（matplotlib中的专有名词，图形中组成部分，不是数学中的坐标系。）

随后输出下面这些图片：

NBA历年MVP得分变化趋势

NBA历年MVP得分条形图

NBA历年MVP助攻变化趋势

NBA历年MVP助攻条形图

NBA历年MVP篮板变化趋势

NBA历年MVP篮板条形图

4 如何实现

一：数据爬取部分

数据的爬取分为如下几个步骤

利用urllib获取网页数据

利用bs4的beautifulSoup进行数据的解析和获取想要的数据

利用re正则表达式结合bs4得出想要的数据

利用sqlite3存储数据

二：数据可视化部分

把所有数据从数据库中取出来

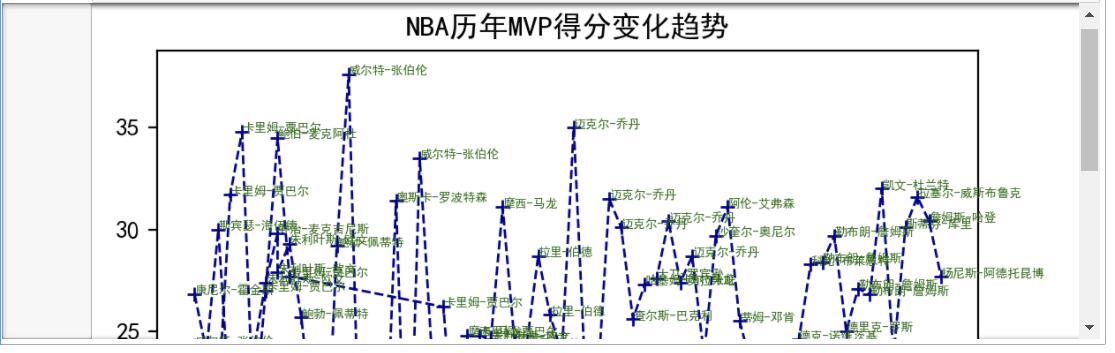
利用了matplotlib.pyplot进行可视化

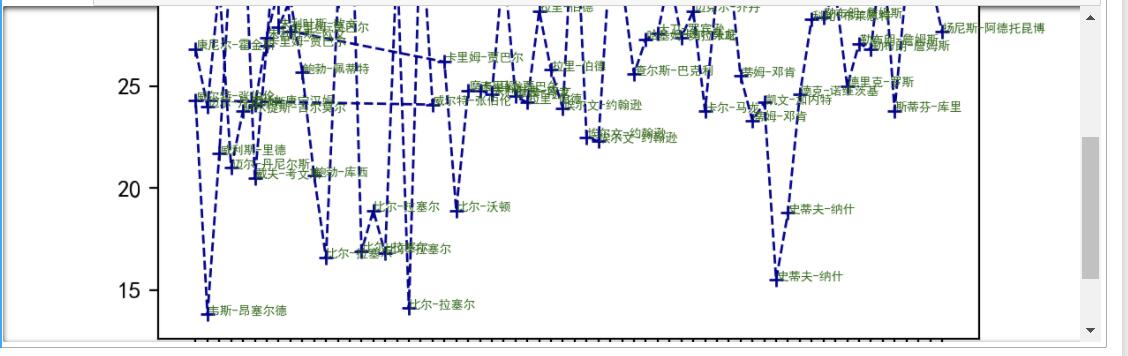
5 测试

下面是在jupyter notebook中的一些测试截图：

# 输出折线图

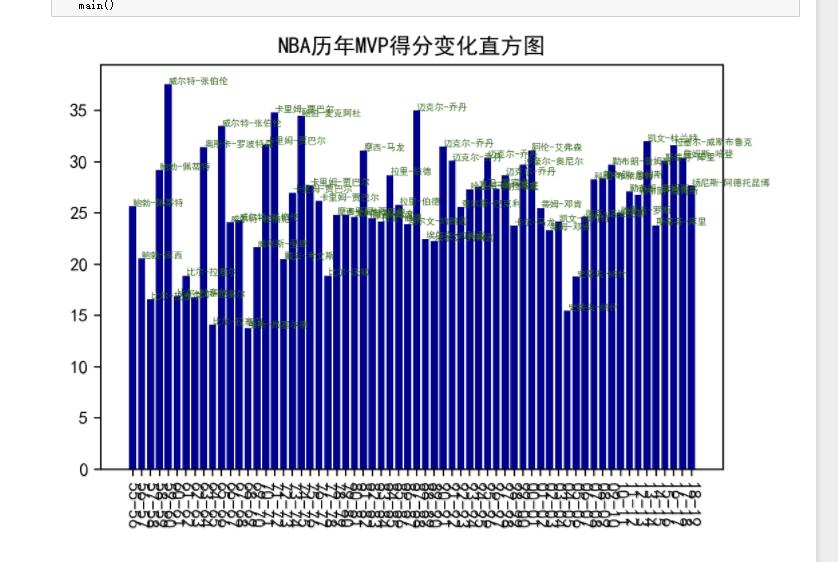
NBA历年MVP得分变化趋势



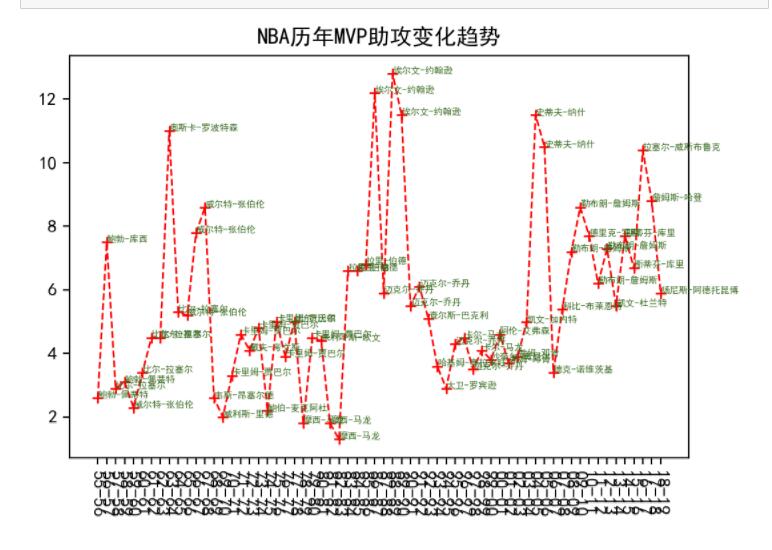


#输出直方图

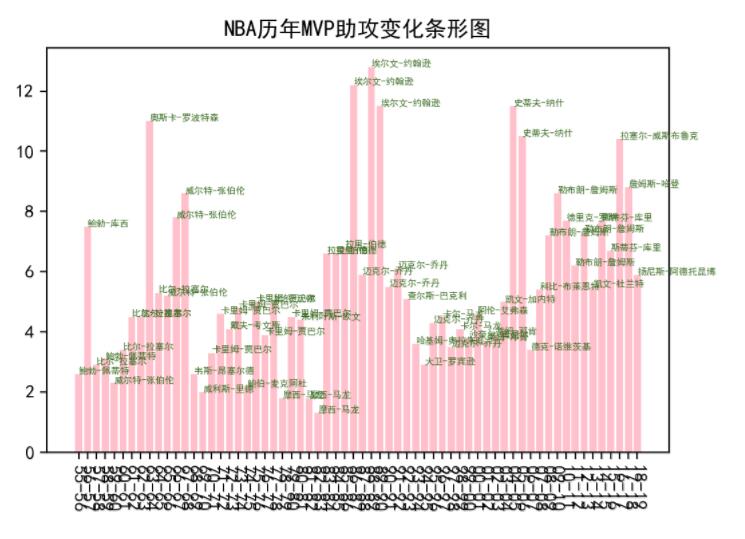
NBA历年MVP得分变化直方图



输出：NBA历年MVP助攻变化趋势折线图

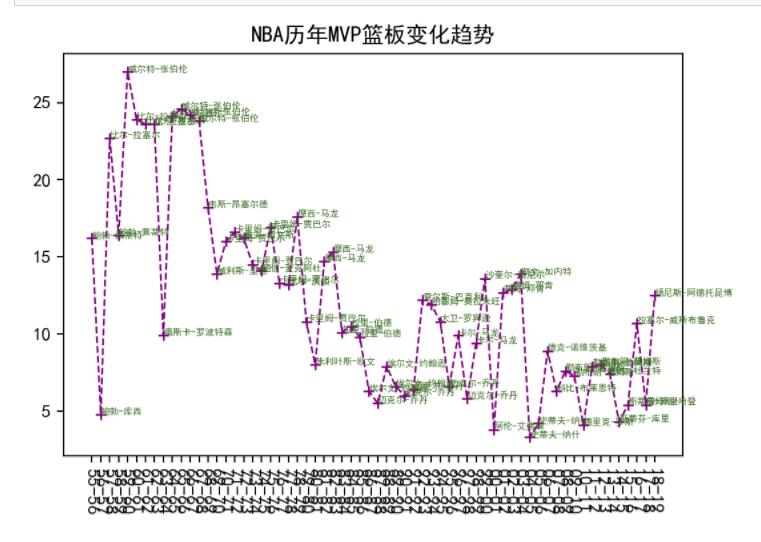


输出：

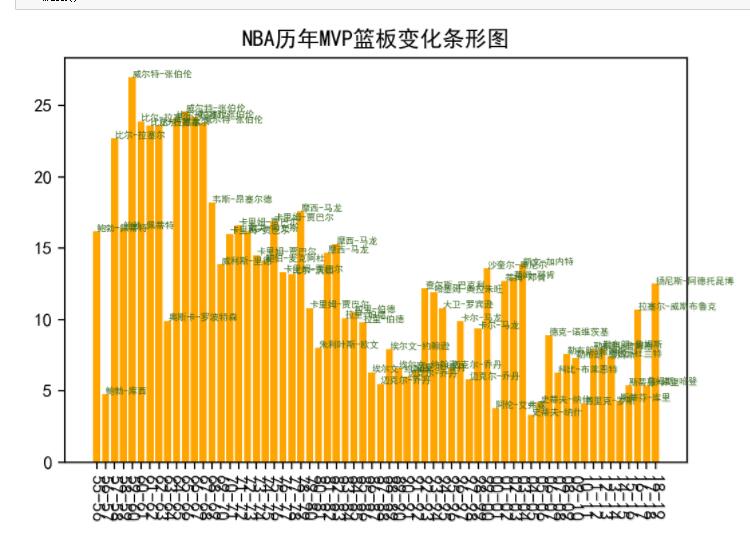
NBA历年MVP助攻变化条形图

# 输出折线图

NBA历年MVP篮板变化趋势



NBA历年MVP篮板变化条形图



6 项目分析

这是一个从http://www.stat-nba.com/award/item0.html这个网页上爬取数据，利用urllib获取网页数据，然后利用bs4的beautifulSoup进行数据的解析和获取想要的数据，再利用re正则表达式结合bs4得出想要的数据，利用sqlite3存储数据，把数据从数据库中取出来，然后利用matplotlib.pyplot进行可视化的一个项目。

得出：

NBA历年MVP得分变化趋势（折线图）

NBA历年MVP得分条形图

NBA历年MVP助攻变化趋势（折线图）

NBA历年MVP助攻条形图

NBA历年MVP篮板变化趋势（折线图）

NBA历年MVP篮板条形图

这些图。

7 不足以及可改进之处

（1）Sqlite的缺点：1。并发访问的锁机制。SQLite在并发（包括多进程和多线程）读写方面的性能一直不太理想。数据库可能会被写操作独占，从而导致其它读写操作阻塞或出错。2。SQL标准支持不全。在它的官方网站上，具体列举了不支持哪些SQL92标准。不支持外键约束。3。网络文件系统（NFS）有时候需要访问其它机器上的SQLite数据库文件，就会把数据库文件放置到网络共享目录上。这时候你就要小心了。当SQLite文件放置于NFS时，在并发读写的情况下可能会出问题（比如数据损坏）。原因据说是由于某些NFS的文件锁实现上有Bug。

（2）界面还可以做进一步美化，变得更简洁，一目了然。

（3）可以多找几个网站进行数据收集，提高数据的可信度，收集的数据还不够多。

（4）还可以制作更多种类的图表，有利于使用者从不同的角度来分析数据，方便使用。

8 成员贡献

（1） 201892231 周子康（组长）

**工作占比：40%**

主要工作内容：（1）总负责（2）撰写报告（3）选择爬取数据的网站（4）负责数据爬取代码的编写（5）负责数据可视化的代码编写（6）设计与实现前端界面（7）负责数据的存取与可视化代码的编写。

（2） 201892075 吴宇嘉（组员）

**工作占比：20%**

主要工作内容：（1）撰写报告（2）进行项目测试工作（3）进行数据的分析与前端界面的一些实现。

（3） 201892075 宋济廷（组员）

**工作占比：20%**

主要工作内容：（1）撰写报告（2）进行项目测试工作（3）负责数据的一些存取方面的代码编写。

（4） 201892075 李修华（组员）

**工作占比：20%**

主要工作内容：（1）撰写报告（2）进行项目测试工作（3）负责数据可视化的总体设计。

9 感想

周子康：

首先很庆幸自己选到了这门个性化选修课，可能是我个人比较认为这门课程所用的语言很特别很奇妙，老师也很有趣，不仅能让我们更好的了解Python这门语言真正的用途。在学习开源软件基础这门课程的这段时间以来，并且自己也能认识并且学习到很多知识。

作为本次大作业的组长，看着项目从呱呱坠地到瓜熟蒂落，就像看着自己的孩纸长大一样。心中感慨万千，成长的过程总是快乐与痛苦并行，喜悦与烦恼同在的。其实我对python也有比较大的兴趣，所以自己研究了很长时间的python，从python小白到小有所成吧，期间也花了很大的努力，学习python是一个很累的过程，但是不苦，在开源软件基础这门课中我学到的东西很多，python基础知识、爬虫相关技术以及网站框架的架构和制作。再如实现爬虫需要用到requests以及BeautifulSoup模块和re正则表达式，等等这些都是完成一个项目的思维方式和构架。

通过这次开源软件基础大作业的学习，我感受到想要完成一个项目，不是一蹴而就的，需要耐心的一个一个功能的去完成，每写完一个代码模块的时候，真的会有一种非凡的成就感，不会的就去学习，然后根据自己课上所学到的知识，结合运用从而完成自己的项目。

在我看来，学习python的最大动力其实是兴趣。正如伟大的科学家爱因斯坦所说：“兴趣是最好的老师”。当你对编程具有浓厚的兴趣时，你就会想要自己动手编程出一个小程序，甚至是一个游戏。而编程的前提，是你需要掌握一门编程语言。在python的学习中，每当我从书中或在信安协会课程上学会新的语句或函数，我都会自己进行实验，编程出一个自己的小程序，同时修复各种bug。这对我熟练掌握python是一个很大的帮助，在这个过程中，我慢慢掌握了各自的用法。当遇到自己无法解决的难题时，不要总是想着放弃，向老师或者同学求助也不失为一种好方法。

其实我对python也有比较大的兴趣，所以自己研究了很长时间的python，因为之前也学习过python的一些基础知识，所以这次的大作业对我而言也是一个很好的锻炼。

通过这次的项目开发设计，我感受到了python的强大，大数据统计的功用，以及前端开发的魅力，以后的日子我也会继续努力学习，脚踏实地，与自己的组员一起成长。

宋济廷：

这次的大作业从头脑风暴到前期的精心设计、中期的分工实施再到后期的测试与优化。每一步走的都让人流连忘返，每一步都让人记忆犹新。在这一段时间中，我们不仅得到了知识和技能上的提升，更懂得了队友之间该如何默契地合作，如何与人更好的沟通。“没有人有义务为你做任何事情。”在团队生活中，我们不是他人的boss，跟队友没有利害关系，所以我们没有权利对别人颐指气使，别人做这些事完全是出于自己的一份责任心和对你的信任，所以我们不要把别人的努力当做理所当然，要学会感谢他们的信任，并且自己也应该为了他们的信任付出自己的时间和精力。“己欲立而力人，己欲达而达人。在我们的大作业中，我们也有比较明确的分工，分析、设计、编码、测试、文档，在规定的时间内，一个人很难全都做完，更重要的是很难一个人做好，因为他每个模块都精通。因此，在这个追求双赢的时代，学会在团队中生存不可或缺。眼前的问题往往比长远的bug更吸引人的眼球，我们小组也认识到了这一点，所以就花了较多的时间在设计上面这也提醒我，在今后的开发过程中多注意后期实践跟理论的差距，丰富自己的经验，以后设计的时候就会更明智一点。任何程序都会有bug，团队活动亦然。出了问题，本能的反应是趋利避害，这个时候就会出现推卸责任或者埋怨他人，很少会从自己身上找原因，出了问题最重要的是解决问题，早点将损失降到最小，弥补过失而不是推诿责任。这既不利于团队的团结也不利于问题的解决，所以出了问题，先解决它。在这个有些浮躁的社会，学生也变得浮躁，61分太多，60分正好，完成作业即大功告成，课本束之高阁，很少人真正去把自己正从事的东西做深、做广。自我的驱动力是一个人变得优秀的内因，在一件件小事草草交差之后，自我的要求就会越来越低，所以我们要警惕在小事上断送自己的好习惯，尽力做好一件小事，才能做成大事。锐意进取，追求卓越！

李修华：

通过这个学期的开源软件基础，我对以前了解不太深的开源软件相关知识有了清晰的了解。

首先是开源这个概念的诞生历程，1983 年，AT&T 发布了大名鼎鼎的 Unix System V。同时 AT&T 改变了 Unix 的授权模式，将其变成了收费的闭源商业软件。AT&T 的行为只是当时软件行业的一个缩影，很多企业或个人都一改过去的源码免费分发模式，而转向闭源的商业软件模式。由此催生了自由软件运动（始于 1983 年）。发起人 Richard Matthew Stallman，业界尊之为 RMS。

后来还学习了软件的知识产权和开源软件许可证，也了解到其中的很多趣味小故事。后来也了解了github以及gitlab等开源代码相关仓库，同时也有老师简述了关于python的相关开源知识，例如库的使用以及更多代码的管理和规范

最后在完成我们这份大作业时，采用了python爬虫技术来采集NBA相关数据，并最终呈现出结果。我从中学到了关于爬取数据的方法。Gitlab的使用，以及和队友的密切配合。在这个任务结束后，我想对于整个团队合作的软件的开发有了一定了解。给我许多收获。

吴宇嘉：

通过本次开源软件基础的课程，我对计算机这门学科有了许多新的认识，这次的大作业也让我更加感受到灵活运用所学知识的不易。这次的大作业，我们采用自主选题，分组完成的方式，组内分工明确、有条不紊，程序的编写也是经过数次的分析、思考和修正的，几经波折最终圆满完成了对大作业的编写与测试。  
在完成作业的过程当中，我从刚开始的无从下手到后来的逐渐熟练，使我对python语言有了更深刻的认识，在完成作业时也会遇到许多难题，有些东西感觉自己做的是时候明明没什么错误，偏偏程序运行时就是有错误，让其他同学帮忙看了一下，才发现其实是个很小的错误。在日常学习生活中也会有这样“当局者迷，旁观者清”的情况出现，这时我们应该积极去寻求帮助，在同学的帮助下进行改进，这在以后的生活中都是十分重要的。  
同时，我也深深体会到了积累知识的重要性。在程序的编写过程当中我们遇到了不少难题，但是经过我们大家的讨论后问题都逐一得到了解决。伴随着大作业的逐渐完成，我也收获颇丰，同时也更深刻的认识到要做一个合格的程序员并非我以前想像的那么容易，最重要的还是细致严谨。社会是不会要一个一无是处的人的，所以我们更要更多更快地完成从一个学生向工作者转变。   
我更多学到的是不懂就要问和自己应该尽自己的全力去尝试，哪怕失败，也要尽自己的全力，和身边的同学一起探讨而不是想办法逃避问题，保持团队合作和发挥团队意识，最后在大家的努力下程序终于运行成功，这种感觉美不可言，心情愉悦至极，有很强的成就感。  
最后，这次开源软件基础课程让使我学到了很多东西，我们利用计算机求解问题的能力得到了提升，锻炼了我们团队合作能力， 理论知识也得以在实践中运用。为我以后的学习做了引导，点明了方向。