**利用爬虫理论，实现12306抢票小助手**

**目录**

1. 引言
2. 系统分析报告

2.1项目背景

2.2可行性分析

2.2.1编写目的

2.2.2经济可行性分析

2.2.3技术可行性分析

2.2.4操作可行性分析

2.2.5时间可行性分析

2.3需求分析

2.3.1功能需求

2.3.2数据需求

三、系统设计说明

3.1总体设计

3.1.1需求概述

3.1.2软件结构

3.2程序描述

3.2.1UI界面与输入项目

3.2.2数据爬取

3.2.3登录功能

3.2.4输出功能

四、系统实现说明

4.1编写目的

4.2详细设计

4.2.1 UI界面与输入项目

4.2.2数据爬取实现

4.2.3登录功能实现

4.2.4预定功能实现

五、系统开发进度

5.1每日工作汇报

5.2测试工作汇报

5.2.1系统测试计划

5.2.2系统测试分析

1. 项目开发总结报告

6.1对项目开发效率的评价

6.2对项目开发质量的评价

6.3对项目开发方法的评价

6.4经验与教训

1. **引言**

每年春运时节，微信群中就充斥着抢票求赞，助力加速的链接。微信中失联已久的人突然弹出一则消息：能帮我点下抢票加速么。但即使加速包加破进度条，在一些热门的线路中，也未必能买到想要的火车票。

在现如今抢票难的背景下,我们决定开发一款12306抢票助手小插件，旨在帮助用户提高抢票效率，增加抢到所需火车票的概率。

以下为本小组的成员介绍：

|  |  |  |
| --- | --- | --- |
| 夏海峰 | 组长 | 统筹安排 |
| 熊诗驰 | 组员 | 代码编写 |
| 杨亮 | 组员 | 代码编写 |
| 虞含怿 | 组员 | 文档编写 |
| 鄢阳天 | 组员 | PPT制作与展示 |

1. **系统分析报告**

**2.1项目背景**

本次开发的插件名称为“12306抢票小助手”，由本次课程的第七组“606团小组”作为项目提出者并负责项目开发工作。该插件仅用作交流学习使用，为学习实践成果，不用于任何商业用途。

**2.2系统开发需求**

该系统需要方便地提供给各行各业及各年龄段，对购买火车票有需求的用户，所以应尽可能方便更多人的操作。用户在登录12306网站后只需输入起始地、目的地与所需车票时间段便可快速查询有无空余车位，若有则会告知用户，用户可以选择相应的列车，点击预定按钮即可一键完成预定，并通知用户尽快完成支付。另外，该系统还应有供用户选择的各类选项：如是否需要学生票，是否限定高铁动车等。

除此之外，该系统最需满足的一点是用户使用该系统应尽量保证其使用体验好，在最短时间内帮助用户抢到车票。

**2.2可行性分析**

**2.2.1编写目的**

编写该可行性分析报告旨在说明该插件开发项目的实现在技术、经济和社会条件方面的可行性；评述为了合理地达到开发目标而可能选择的各种方案；说明并论证所选定的方案。

**2.2.2经济可行性分析**

本次系统设计不用于商业用途，，只用于学习交流使用。开发过程中无任何实际经费支出，故本次项目开发从经济上来讲是可行的。

**2.2.3技术可行性分析**

* 功能：用户登录后，输入所需车票的起始地与目的地，并确定所需时间段，提交后系统应查找所有满足要求的列车信息并确认是否有余票，若有则帮助用户快速预定车票并通知用户付款。该系统还应有选择学生票，是否限定高铁动车等选项。
* 性能：该系统应具有较快的处理速度来帮助用户完成抢票，尽量缩短系统运行所花费的时间。
* 完成时间：该系统需在6月29日前完成所有编写和测试

工作，确保在6月29日系统能够正常工作。

* 系统使用寿命：因12306网站更新速度较快并反爬机制较为完善，导致抢票脚本的使用寿命可能较短。应采取相关措施，尽量延长该系统的使用寿命。
* 实现方式：用brython编译的可以实现chrome插件，并且在GitHub上创建建立项目库。但12306网站反爬技术较为完备，爬取网站数据难度较大。我们将采用反反爬操作从12306网站爬取列车信息，但因预定方面的反反爬操作过于困难，我们决定采用通过webdrive使用selenium模块去模拟人在浏览器直接跳转到购票的网页，并帮助用户实现车票的预定。

**2.3.4操作可行性分析**

* 从插件管理的角度考虑，我们必须认识到非专业人员很难对内部系统进行调试，出现问题时也很难解决，那么在设计之初就应该明确，设计的整个系统是封装好的，系统的内部运行和功能模块并不对外展示，但有学习交流需求的也可以前往GitHub进行下载，我们的所有系统内部信息，包括升级记录、功能更改记录都保存在其中可以供所有人查看。
* 因为对插件制作难度的估计不到位，所以我们没有能够在规定时间内完成插件的操作。我组决定通过网页形式代替插件进行抢票工作。
* 在登陆时输入账号密码后需要进行一个滑块验证的步骤，这个步骤通过指定路径的办法非常难通过人机测试，所以需要通过其他办法来绕过人机检测，经过学习我们已经找到相关办法并会在4.2.3登录功能实现板块中着重介绍。
* 在完成购票后可以尝试将页面跳转至支付页面，此过程相较之前的登陆模块无较大区别，比较容易实现。但需考虑的一点是，需要等待多少时间再执行点击操作，这一个难点还需要进行更多测试来验证.

**2.3.5时间可行性分析**

整个系统在开发之前我们已经做了详细的工作计划，整个开发将按照时间表来进行，从而充分的保证整个开发计划的可行性。从2022年6月24日到2022年6月29日，我们将要完成系统的需求分析，同时也要学习相关知识技术，积累一定的经验；逐步完善系统各个模块功能，并进行相应的测试；最后还需要总结工作并对系统设计的不足与缺陷提出展望。在2022年6月29日下午展示相关功能，在30日做最后的汇报总结。

**2.3需求分析**

**2.3.1功能需求**

本次系统的功能需求主要基于抢票用户，用户输入相关数据即可查看所有有余票的相关列车信息并提供抢票服务。在插件界面登录后，填写所需列车相关信息后即可为用户展示可以预定的列车信息，用户点击预定按钮后即可帮助用户完成一键登录和购票功能。

**2.3.2数据需求**

在本次抢票助手系统设计中，需要设定的数据有起始站点、目的站点、购票者身份和所需车票的日期等。系统需要从网站爬取所有符合要求的列车信息，并将所得数据整理后返回给用户。

1. **系统设计说明**

**3.1总体设计**

**3.1.1需求概述**

该系统需要方便地提供给各行各业及各年龄段，对购买火车票有需求的用户，所以应尽可能方便更多人的操作。用户在登录12306网站后只需输入起始地、目的地与所需车票时间段便可快速查询有无空余车位，若有该系统则会帮助用户快速预定并以邮件方式告知用户已抢到车票，并通知用户尽快完成支付。另外，该系统还应有供用户选择的各类选项：如是否需要学生票，是否限定高铁动车等。

除此之外，该系统最需满足的一点是用户使用该系统应尽量保证其使用体验好，在最短时间内帮助用户抢到车票。

**3.1.2软件结构**

软件形态为一个使用brython编译的Chrome插件，用户只需安装该插件便可直接在12306网站页面使用，无需安装任何程序。点击插件会显示UI界面，在UI界面输入指定内容便可开始抢票。我们将通过webdrive使用selenium模块去模拟人在浏览器直接跳转到购票网页的方式来完成车票的抢购工作，在完成车票的预定后，用户在规定之间内完成付款即可完成整个购票工作。

**3.2程序描述**

**3.2.1UI界面与输入项目**

用户提前完成12306网站的注册和登陆工作，登陆完成后，打开插件，在UI界面输入所需列车的起始地点和目的地点购票人身份和所需班次时间，点击查询便可看到所有满足要求的列车信息。

**3.2.2数据爬取**

爬取12306网站内相关列车的数据，并将数据存入列表中，通过数据的解析实现从输入时间、始末地等信息到输出所需列车信息的工作。

在UI界面返回所有符合要求的列车信息,展示部分列车信息，若想查询剩余列车信息，可点击“查看更多”按钮查看。

**3.2.3登录功能**

用户的账号密码可提前输入，在点击预定按钮后会自动实现登陆环节，但需要实现包括对滑块验在内的功能。

**3.2.4预定功能**

当用户想要预定某一车次的车票时，用户点击所对应列车的预定按钮即可完成一键抢票功能，无需进行多余操作，待抢到票后页面将会跳转至支付页面，用户在规定时间内完成支付，购票即宣布完成。

1. **系统实现说明**

**4.1编写目的**

编写本系统实现说明的目的主要是详细说明本12306抢票系统的结构和内部设计，在系统设计说明的基础上进一步细化系统，提供详细的程序和接口设计，为程序开发提供直接的支持。

本系统实现说明主要面向系统分析人员，程序员和以后的维护人员，作为程序员开发系统的根本依据和在系统维护阶段，维护人员对系统实施维护的参考资料。

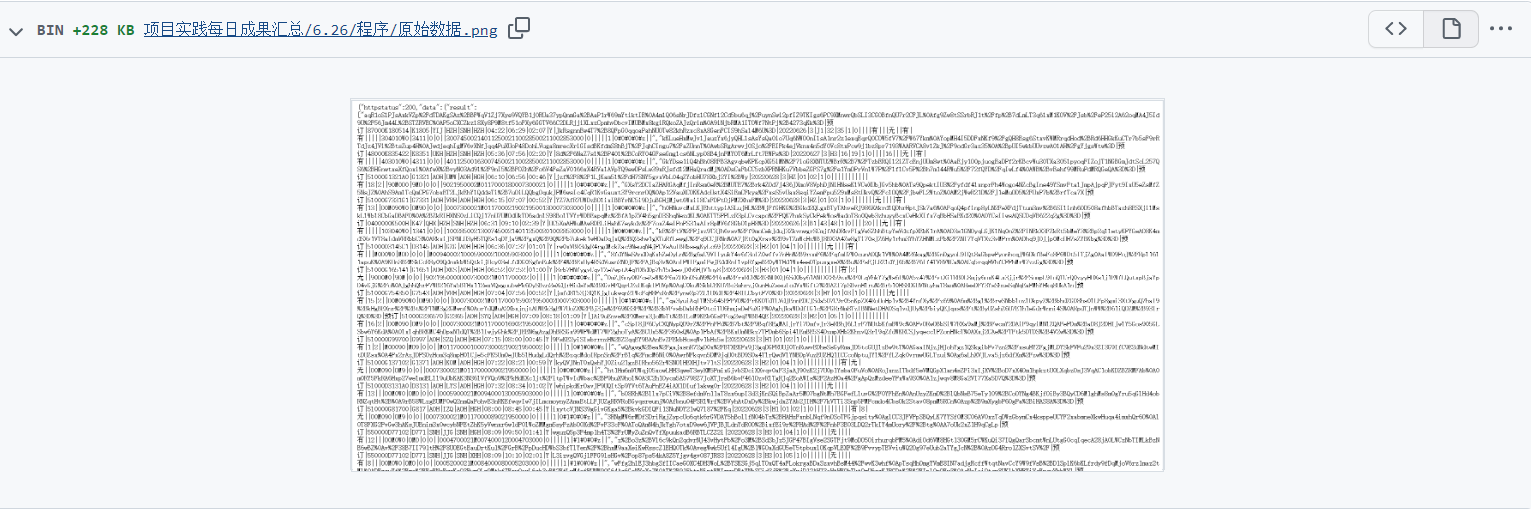
**4.2详细设计**

**4.2.1 UI界面与输入项目**

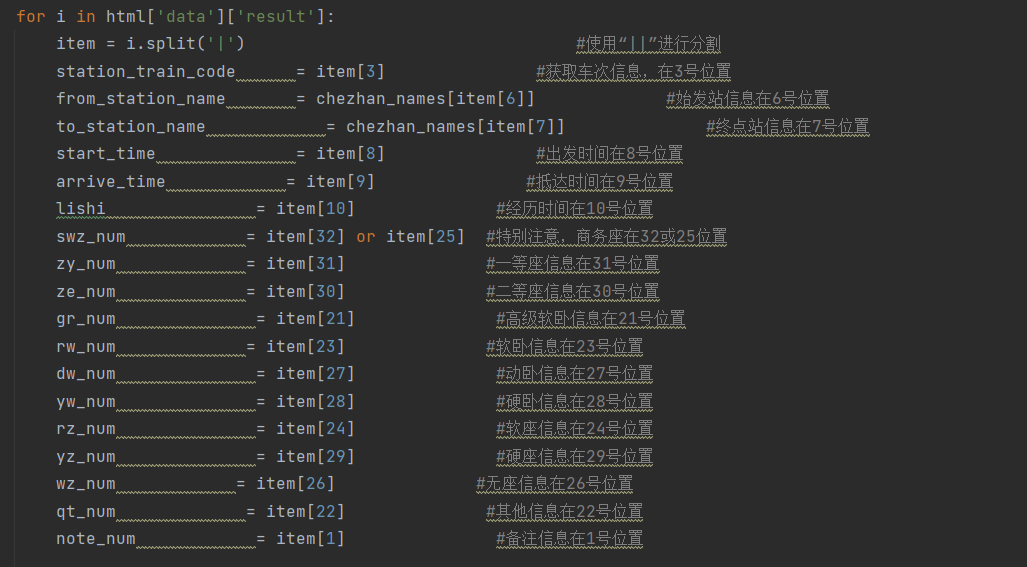
* 第一个界面供用户输入用户名与密码，确定用户的身份为成人还是学生，用户点击“提交”按钮即可实现信息的保存。
* 在第一个界面下方提供跳转按钮，点击跳转后页面会跳转至第二个界面，该界面中供用户输入所需车票的起始地、目的地、车票日期以及支付方式，点击“提交”按钮后界面会显示所有满足要求的列车信息，并在每辆列车后方提供预定按钮，用户可自行选择所需的列车班次。

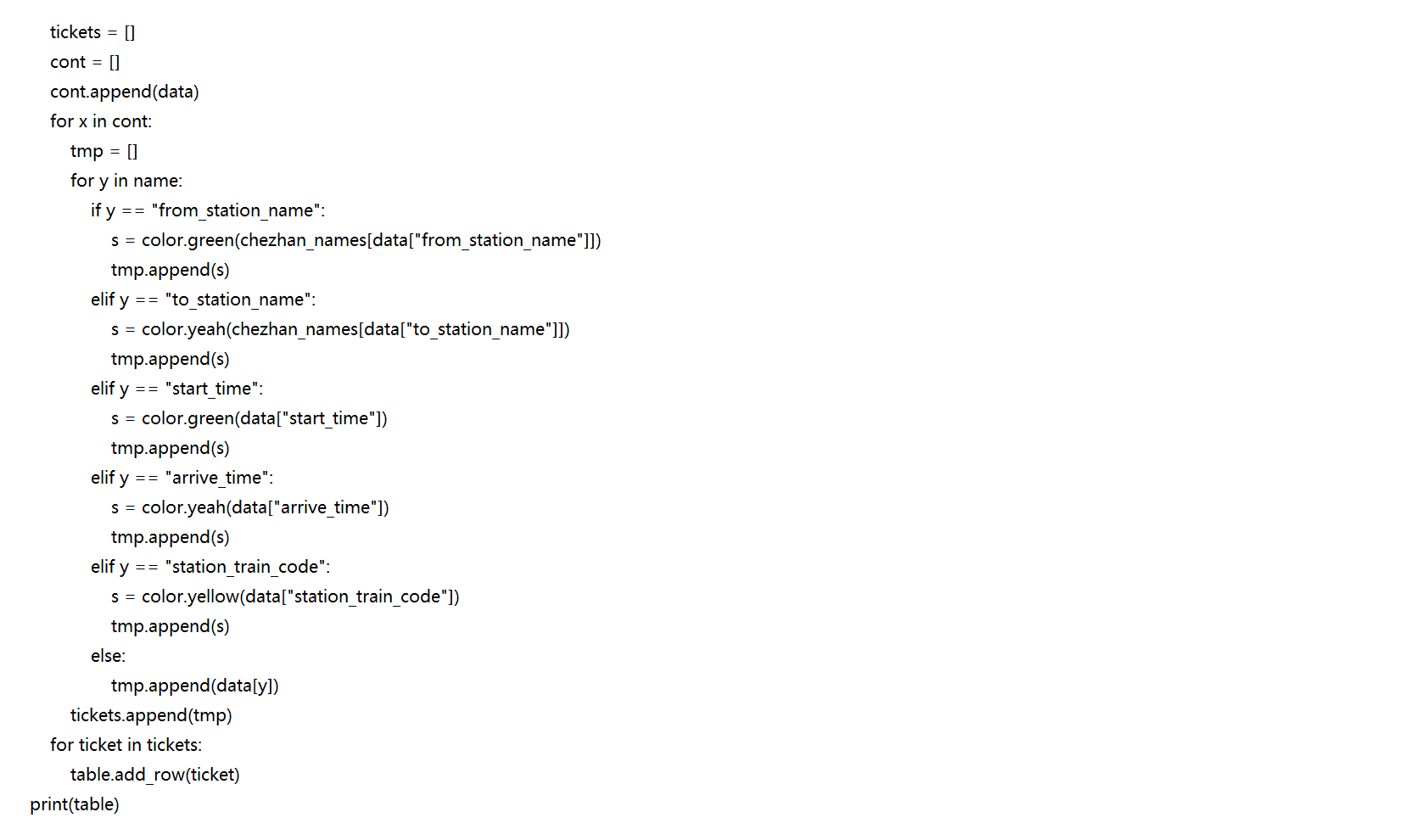
**4.2.2数据爬取实现**

* 向12306网站请求数据需要发送类似许可证的请求，包括user-Agent（请求头）和cookie。其中cookie中包含列车信息和车站代码。而车站名及其编码则应提前爬取。
* 请求发出后会收到返回内容，其形式为Ajax文件，其中包含列车有关信息（该信息会实时更新）。有关信息在文件中以“|”形式隔开。



* 应先对其中数据进行清洗提取操作，其中date为内容，而result为相关内容的结果。将清洗后的数据按照既定规则填入准备好的表中，并添page列作为列车的编号。





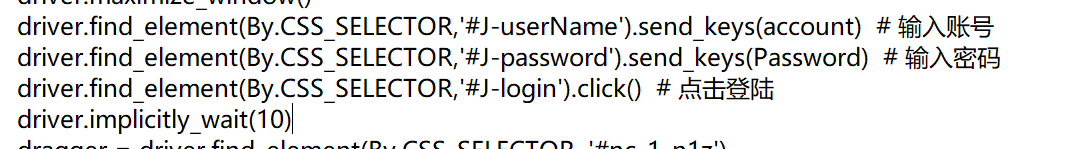
* 在UI界面展示所有列车信息，
* 使用BUY.get.train(int(word),Fs,Ts,date)命令即可选择购票车次，跳转购票页面，之后不需要进行任何操作。

**4.2.3登录功能实现**

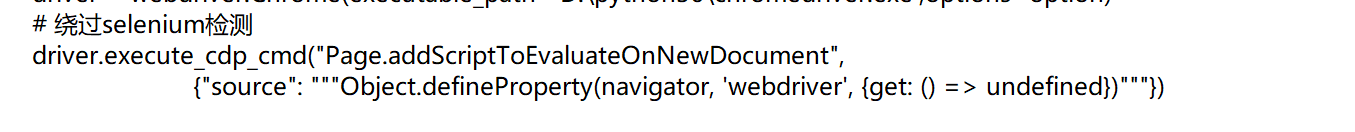
* 在前端界面首先输入用户名与密码，并告知程序购票者的身份为学生还是成人，点击“提交”按钮后系统先将这些信息进行保存，然后点击“跳转”按钮后将页面跳转至输入需要车票信息的网页。



* 采用网页驱动器webdriver中的selenium模块来进行车票预定功能的实现。通过executable.path来声明webdriver地址和需要其操作的网页的地址。
* 通过css.selector找到所需填入或点击的位置，账号密码应提前存入以便此时能够自动填入，填入方式通过send.keys来实现，并通过click来进行自动点击。添加drive.implicitly.wait(10)操作来等待网页跳转。

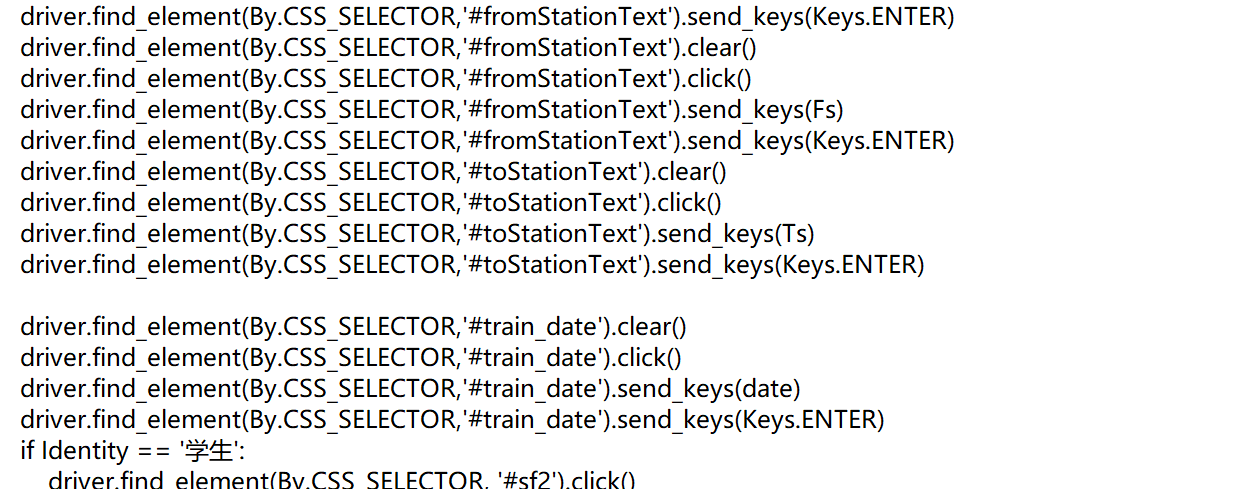


* 因为在输入账号密码点击登录后会有一个滑块验证模块，用于检测滑动操作是否类似人机，即该操作是否被机器控制，若是则会检验失败，无法继续登录。所以我们引入了options库，通过操作可以跳过滑块滑动操作后的检测步骤，即阻止系统对滑块是否为人机的检验工作。

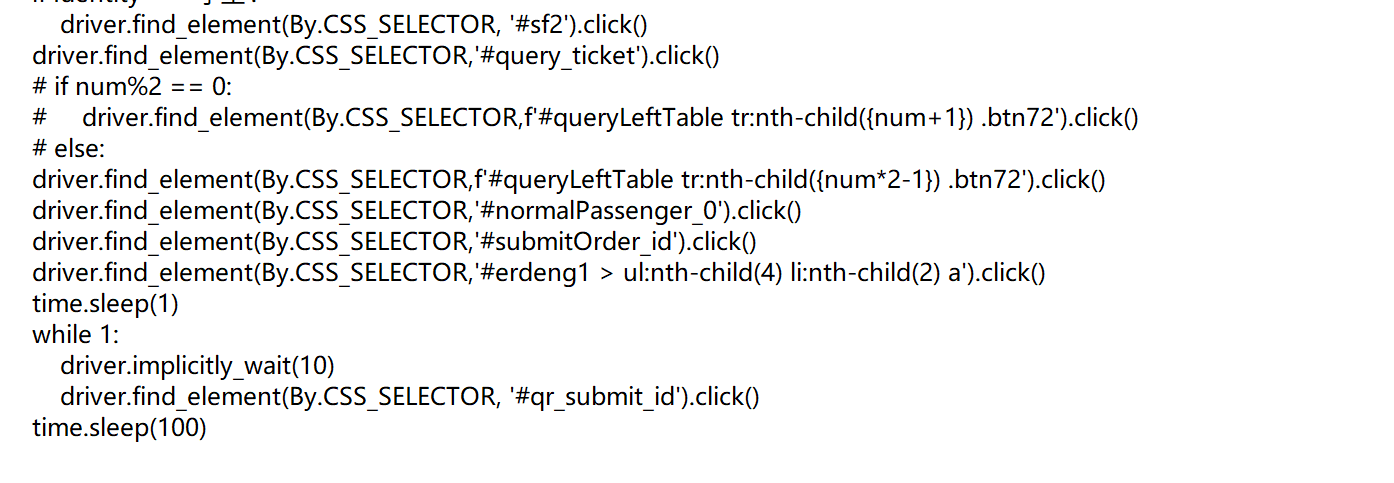


**4.2.4预定功能实现**

* 点击跳转后会出现一个“提示”界面，需要先点击确定按钮，做法与之前的click操作相同。之后的点击操作也较为相似，之后便不再赘述。
* 点击上方的车票选项，选择下方的单程票选项，点击出发地/目的地/日期，先进行清空处理，在键入内容，必须按Enter健进行确认，即进行Key.Enter操作。身份信息默认为普通，通过If identity等操作可以将类型转化为学生，点击查询按钮。



* 跟随之前选择的车次序号找到车次位置，选择乘车人后直接点击提交订单，然后点击确认。预定完成后页面将停留，等待用户进行支付。



**五、系统开发进度**

**5.1每日工作汇报**

**5.1.1 6月24日工作完成情况**

确认了小组组名和小组口号；确定了小组项目实践的选题工作；确认了小组内各成员的工作分配：夏海峰担任小组组长，熊诗驰和杨亮同学负责程序编写工作，虞含怿同学担任文档的编写工作，鄢阳天同学担任最终的汇报工作。

**5.1.2 6月25日工作完成情况**

程序部分明确了项目开发路径，学习了相关技术，确定软件形态为一个使用brython编译的Chrome插件，并且在GitHub上建立了项目库；文档方面完成了目录的编写工作，确定了项目文档的大致格式：答辩方面我们大致确定了展示的大纲。确定了明日的工作计划。

**5.1.3 6月26日工作完成情况**

程序部分经过多次尝试最终实现了对12306网站的request和反反爬虫操作，并实现了chrome拓展的简单页面；文档部分完善了文档的总体结构，完成了部分非有关代码章节的撰写；展示答辩部分确定了最终展示的PPT模板，并完成了非代码部分的PPT制作。

**5.1.4 6月27日工作完成情况**

程序部分尝试使用request与12306进行交换完成购票操作，但因网页返回内容与期望不符并无良好解决办法最终选择放弃，转而通过webdrive使用selenium模块去模拟人在浏览器直接跳转到购票页面的操作。程序部分还新增了chrome拓展的查询结果和购票功能页面；文档部分完成了分非代码和部分代码部分的文档编写工作，因程序方面实现方式的转变，代码部分的文档需要重新作出调整；展示部分完成了部分有关代码部分的展示PPT制作。

**5.1.5 6月28日工作完成情况**

程序部分攻克了12306的滑块验证，实现了前后端数据交互，在前端显示车票的查询结果，使登录购票一键化；文档部分的编写工作基本完成，前端界面介绍，功能测试和总结报告尚未完成，需进一步改进；展示部分完成了大部分PPT的制作，为最后的展示进行了文案方面的准备。

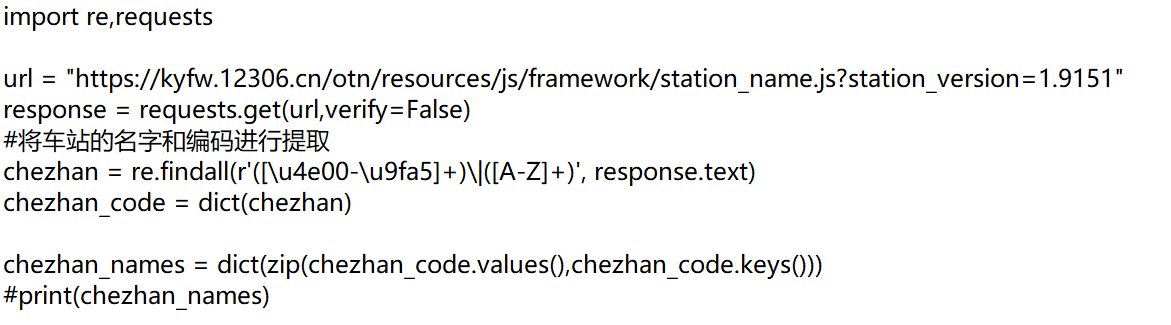
**5.1.6 6月29日工作完成情况**

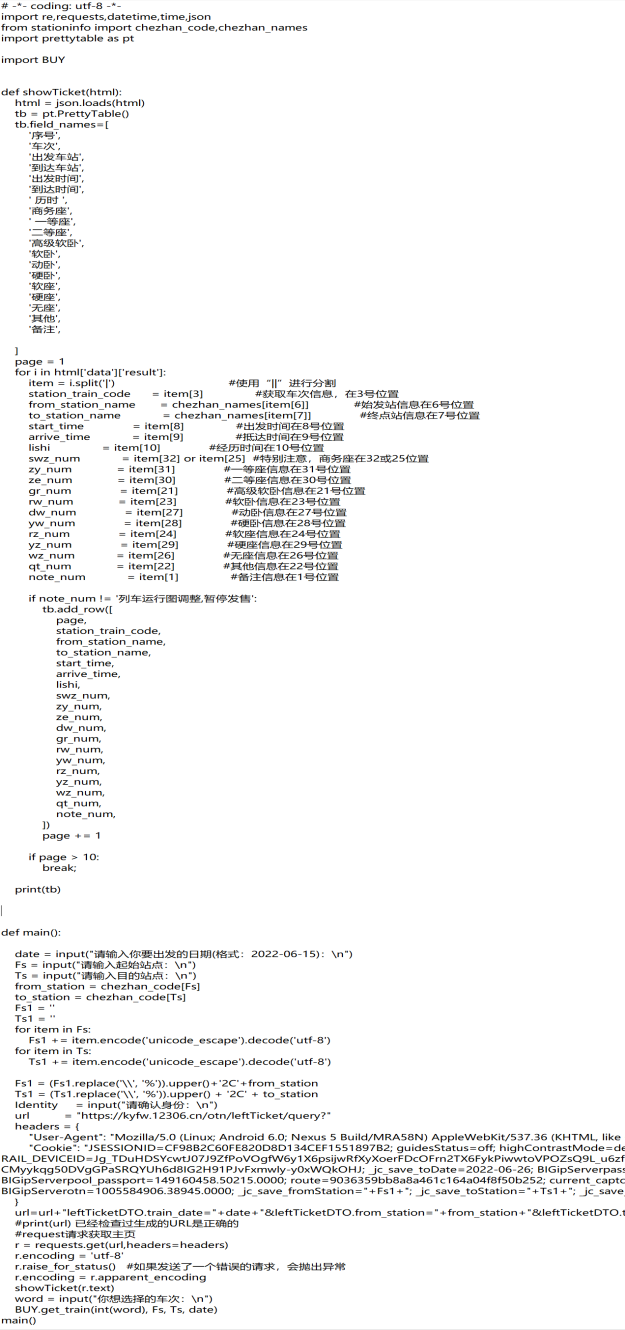
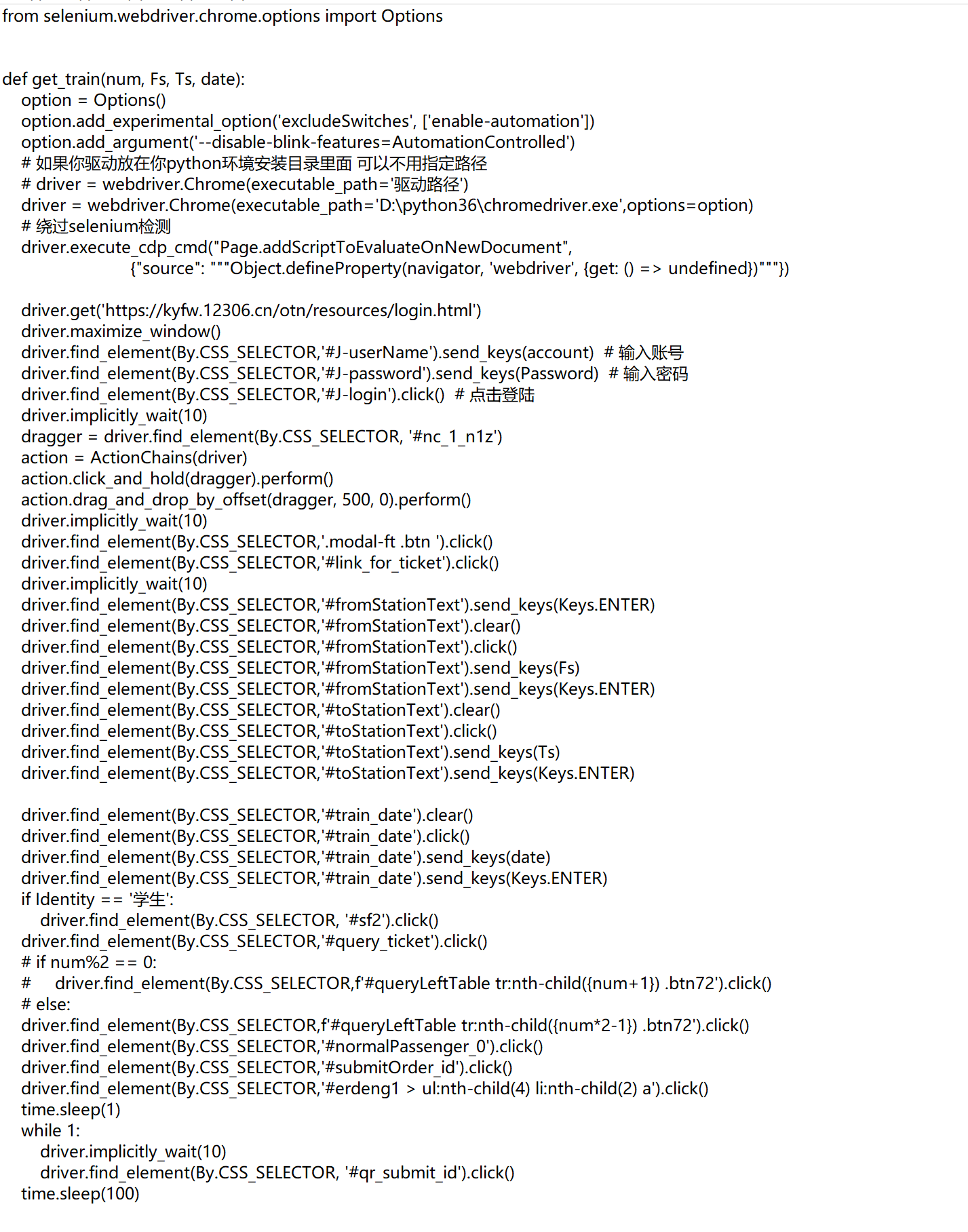
**5.2测试工作汇报**

**5.2.1系统测试计划**

* 6月26日测试12306网站的request和反反爬虫操作；前端UI界面设计
* 6月27日，在前一日测试成功的前提下，尝试使用request与12306进行交互完成购票操作：增加Chrome拓展的查询结果和购票功能页面
* 计划更改：6月28日，因前一日尝试使用request与12306进行交互完成购票的测试失败，我组决定转变编写思路，采用通过webdrive使用selenium模块去模拟人在浏览器直接跳转到购票页面的方式来实现帮助抢票的功能。
* 新增计划：6月29日，在老师的建议下，我们新增了购票成功后将页面跳转至支付页面的操作。

**5.2.2系统测试分析**

* 6月26日测试成功，程序实现了对12306网页的request请求和反反爬虫操作，成功从12306网页爬取了所有所需列车的信息；前端UI界面模型已建立。
* 6月27日，在尝试使用request与12306进行交互完成购票操作的过程中，我们遇到了许多困难，并最终因没有找到好的解决办法决定更换购票实现方式。
* 采用网页驱动器webdriver中的selenium模块来进行车票预定功能的实现。通过executable.path来声明webdriver地址和需要其操作的网页的地址。通过css.selector找到所需填入或点击的位置，账号密码应提前存入以便此时能够自动填入，填入方式通过send.keys来实现，并通过click来进行自动点击。添加drive.implicitly.wait(10)操作来等待网页跳转。因为在输入账号密码点击登录后会有一个滑块验证模块，用于检测滑动操作是否类似人机，即该操作是否被机器控制，若是则会检验失败，无法继续登录。所以我们引入了options库，通过操作可以跳过滑块滑动操作后的检测步骤，即阻止系统对滑块是否为人机的检验工作。点击跳转后会出现一个“提示”界面，需要先点击确定按钮，做法与之前的click操作相同。之后的点击操作也较为相似，之后便不再赘述。点击上方的车票选项，选择下方的单程票选项，点击出发地/目的地/日期，先进行清空处理，在键入内容，必须按Enter健进行确认，即进行Key.Enter操作。身份信息默认为普通，通过If identity等操作可以将类型转化为学生，点击查询按钮。跟随之前选择的车次序号找到车次位置，选择乘车人后直接点击提交订单，然后点击确认。预定完成后页面将停留，等待用户进行支付。



* 在测试跳转支付界面的过程中我组遇到了一些问题，因为12306网站在完成购票操作后还要进行一段时间案的操作才能通知用户进行支付，而这个时间的长短没有规律可循，这就导致我们很难制作点击操作去将页面进行跳转。在6月29日中午验收时我们完成了部分功能，可以将页面跳转至选择支付方式的界面。剩余内容我们将在结课后继续完成制作。

**六、项目开发总结报告**

**6.1对项目开发效率的报告**

因我组在项目开始的第一天便制定了详细的工作计划，并为每位成员分配了详细的工作，组长也对每位组员在每天需完成的内容进行了监督与指导，我组在此次开发中的时间分配较为合理。

但因为我组少一人的原因，在编写代码的工作上任务较重，特别是因为开发前期我们错误估计了理想方式的时间难度，错误估计了12306网页的反爬能力，并且错误判断了插件的实现难度，导致开发前期走了很多弯路，最后两天的时间较为紧张。

**6.2对项目开发质量的评价**

我组在程序实现功能后进行了足够多次数的测试，保证了运行成功率维持在一个较高水平。但需要注意的一点是，若该系统短时间在同一台设备上运行了过多次数，滑块验证模块的人机验证会有失败的风险，可以通过在进行滑块验证时任意移动鼠标来提高通过验证的概率。

总体来说，本系统质量较高，用户可以放心使用。但因为系统开发仍处在初级阶段，系统的安全无法得到较好的保证。所以用户将数据存入系统中可能会存在一定的风险。

**6.3对项目开发方法的评价**

本系统在前端使用了HTML、CSS等相关方面的知识；后端方面应用爬虫相关知识，使用了网页驱动器webdriver中的selenium模块进行代码编写，引入options库进行滑块验证的跳过等。

**6.4经验教训**

在本次开发系统中，最大的收获莫过于团队协作，这也是小组第一次合作开发项目，经验有些不足，但总的来说协作还算默契。其次，我们的任务时间分配还有进一步改进的空间，代码部分任务较重，而文档编写与PPT制作方面的空闲时间较多。希望可以在今后的协作中进一步改进。