实验报告

何锦烨 PB18000345
2021.4.11

一：任务说明

此次代码通过爬取豆瓣读书（https://book.douban.com/）中两个模块——新书速递，最受关注图书榜——的相关信息，分别进入它们的界面，找到虚构及非虚构类中排名前十的图书，进入这些图书的主页，获取书名，作者，评分，简介等相关信息，在 GUI 中展示出来。

二：实验细节

1．**目标网址**：豆瓣读书网址，https://book.douban.com/

2．类

book
属性：

名称	数据类型	说明
name	str	书名
author	str	作者
grade	str	评分（因可能由于人数不足暂无评分，所以设为字符型）
Introduction	str	简介

方法：book.info（url）
参数：url——该图书的主页网址
调用 get_info(url)函数，获得图书的相关信息，赋值给书的对应属性。

3．函数

1）、ask_url(url)
参数：url——需要爬取的网页的网址；
调用 requests 包中的 requests.get 获取网页信息，赋值给 response 变量；
设置访问头 head，做简单反反爬处理；
返回 response 变量。

2）、find_newbook(num)

参数：num——需要获取的图书主页的序号
其中序号 1-20 是虚构类图书，20-40 是非虚构类图书
调用 ask_url(url)访问需要访问的网页，通过 bs4 包中的 beautifulsoup 函数对爬取的网页进行解析，利用 CSS 选择器选得需要的信息
该函数功能是从豆瓣读书主页中找到新书速递网页，再从新书速递网页中找到，指定序号的图书主页的网址，返回该网址。

3）、find_popularbook(num)
参数：num——需要获取的图书主页的序号
调用 ask_url(url)访问需要访问的网页，通过 bs4 包中的 beautifulsoup 函数对爬取的网页进行解析，利用 CSS 选择器选得需要的信息
该函数功能是从豆瓣读书主页中找到最受关注图书榜网页，其中虚构类与非虚构类网址不同，需要分别找到他们的网址。
再从相应最受关注图书榜网页中找到指定序号的图书主页的网址，返回该网址。

4）、get_info(url)
参数：url——需要获得的图书信息的图书主页
调用 ask_url(url)访问需要访问的网页，通过 bs4 包中的 beautifulsoup 函数对爬取的网页进行解析，利用 CSS 选择器选得需要的信息，即图书的书名，作者，评分，简介。
把这些信息封装进一个变量 info，返回 info。

以上函数出现在文件 utils.py 中

5）、helptxt():
定义了单击菜单栏中 help 项后的反应——弹出窗口显示文字说明简要介绍了 GUI 的数据来源和功能。
设置了新窗口 helpwin，在上面添加 label 部件，利用 label 显示文字说明。

6）、getindex(listbox)
参数：listbox——指定需要寻找选中项标号的 listbox。
遍历指定 listbox 中所有标号，判断它有没有被选中，若有返回标号值；若最终没有一项被选中则返回-1。

7）、showintro()
定义了单击 button 部件后的反应。
根据 Radiobutton 得到的值，寻找对应 listbox 中被选中的标签的标号。
调用了 getindex(listbox)实现该功能，若返回值是-1，说明没有被选中的项，则类型选择不正确，弹出对话框，提示选择正确的图书类型；
否则，则找到了对应的图书，通过他的标号和类别，我们能找到它的相关信息，在 Text 中显示出来。

4．难点与解决方案

1)、如何选择需要的信息
通过 requests.get 得到的文件无法直接获取信息，利用 beautifulsoup 解析后，如何找到指定的信息。
通过网络上的相关材料，学习了 CSS 选择器的相关用法，通过标签和属性找到需要的信息。
有些信息在网页中是文本形式，而另一些是作为属性存在的，要通过不同的方式获取。
要仔细观察网站的结构，如在爬取最受关注图书榜时，运行中出现了错误，经查看后发现，原网页给最后一本书的信息的标签类的属性与其他不同，通过分支语句，进行不同的处理。

2)、如何展示获得的信息
由于豆瓣对书的简介较长，无法全部一起展示，我们选择了首先只展示书名，指定图书后再展示其作者，评分，简介等信息。
但这要求我们获得选中的图书哪一本的相关信息。我们通过 listbox 来展示书名，listbox 中各项有是否选中的状态区别，我们利用这一点，进行交互，获得选中图书的标号。但 python 似乎没有直接获得选中项标号的函数, 所以写了 get_index 函数来实现这一功能。
再利用 Radiobutton 来确定类别，由此我们可以得知需要的展示的究竟是哪一本书的信息。
由于不同书的简介长度不同，利用 label 部件很难较好的展示，所以我们选择了 Text 部件来展示简介。

三：实验总结

1. 结果分析

代码最终能较好实现所需达到的功能。
可以改进的的地方:
1)、在爬取简介时，原网站的简介并非整块文字，通过各种符号进行了分段，比较难处理，最终现实的简介未能保持原来比较美观的状态。
2)、部分代码仍可以精简，如可以一次性爬取所有图书主页的网址，而非每次调用获取一个，当然这样的好处在于，可以改变需要获取的网址的数量，降低占用的内存。

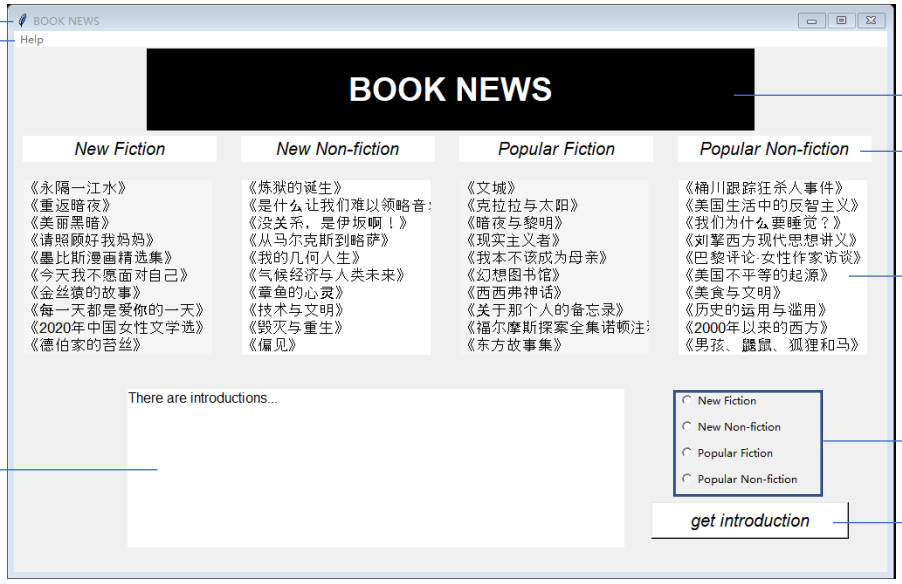
2. 心得体会

在开始写代码之前先设计好功能及实现用的函数，能很大程度上帮助写代码。特别在码 GUI 部分的代码的时候。
在 GUI 的布局上，利用 place（）坐标轴进行摆放能更好实现对齐，使界面更简洁，特别是在部件较多时。

四：界面布局

窗口名

菜单栏，单机弹出 help 页面



大标题

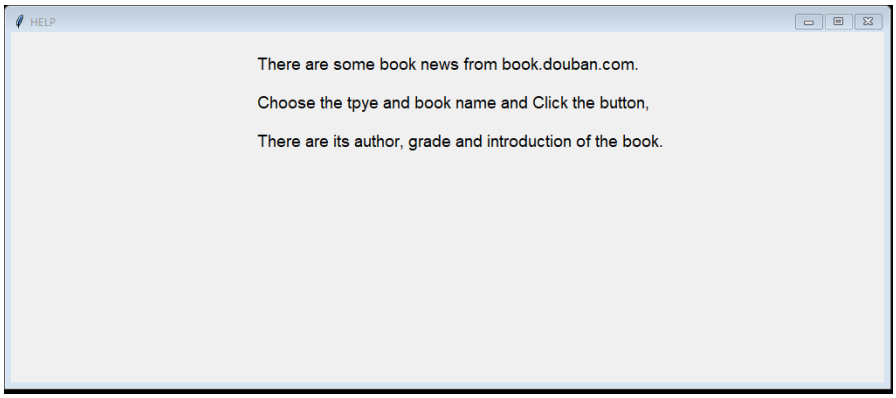
小标题

展示书名

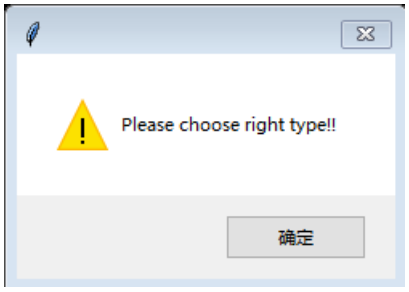
显示简介

选择类别

按钮，单击在文本框展示简介



Help 窗口



类别错误提示框