温州大學瓯江學院

WENZHOU UNIVERSITY OUJIANG COLLEGE

《爬虫期末作业》

题　　目：__各次实验内容与 12306_____

分　　院：_____数信分院_____

班　　级：_____16 计算机科学与技术本三_____

姓　　名：_____赵佳豪_____

学　　号：_____16219111326_____

完成日期：_____2019 年 6 月 15 日_____

温州大学瓯江学院教务部

二〇一九年五月制

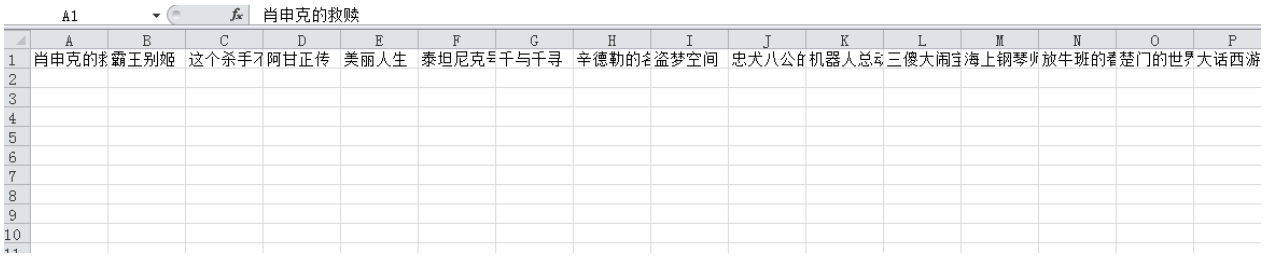
实验 2.py
普通爬取电影 top250
代码

```
import requests
from lxml import etree


def get_page(start_num):
    url='https://movie.douban.com/top250?start=%s&filter=' %start_num
    res=requests.get(url)

    tree=etree.HTML(res.text)
    top250=tree.xpath('//span[@class="title"][1]/text()')
    print(top250)
    return top250
get_page(0)
```

文件 One.py
爬取豆瓣电影 top 榜单，并转换为 csv
代码：

```
import requests
import csv
import lxml
from lxml import etree
class Spider:
    def __init__(self,version):
        self.version=version
        self.result=[]

    def get_page(self,start_num):
        url='https://movie.douban.com/top250?start=%s&filter='%start_num
        res=requests.get(url)

        tree=etree.HTML(res.text)
        top250=tree.xpath('//span[@class="title"][1]/text()')
        print(top250)
        return top250

    def go(self):
        print('Start...')
        for i in range(0,1):
```

```
            top250=self.get_page(i*25)
            self.result += top250

        return self.result
if __name__=="__main__":
    my_spider=Spider('1.0')
    res=my_spider.go()
    with open('D:/cs.csv','a+',encoding='UTF-8',newline='')as csvfile:
        w=csv.writer(csvfile)
        w.writerow(res)
```



文件 Two.py

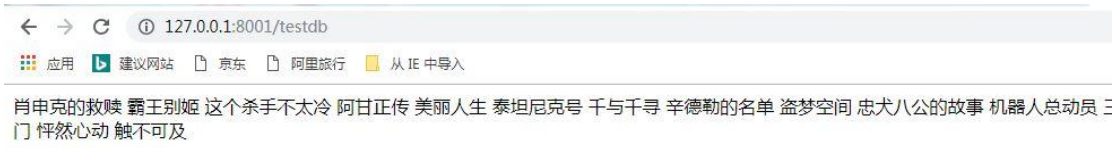
将 csv 转存进数据库

代码：

```
import csv
import pymysql

def readData():
    result=[]
    with open('D:/cs.csv','r',encoding='UTF-8') as csvfile:
        csv_reader=csv.reader(csvfile)
        for row in csv_reader:
            for a in row:
                result.append(a)
    return result


db=pymysql.connect("localhost","root","123456","mysql")
cursor=db.cursor()
res=readData()
print(res)
sql="INSERT INTO testmodel_test(Name) VALUES(%s)"
for a in res:
```

```
        cursor.execute(sql,(a))
        db.commit()
db.close()
```

文件夹 Test（django）
将爬取内容通过 Django 显示到网页
Test/Test/settings.py 配置数据库
Test/Test/testdb.py 连接数据库



京东.py
爬取京东手机商品页面内容保存记事本（Jd.txt）
代码：

```
from selenium import webdriver
from selenium.webdriver.support.wait import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
from selenium.webdriver.common.by import By
import selenium.common.exceptions
import json
import csv
import time

class JdSpider():
    def open_file(self):
        self.fm = input('请输入文件保存格式（txt、json、csv）：')
        while self.fm!='txt' and self.fm!='json' and self.fm!='csv':
            self.fm = input('输入错误，请重新输入文件保存格式（txt、json、csv）：')
        if self.fm=='txt' :
```

```
                self.fd = open('D:\Jd.txt','w',encoding='utf-8')
            elif self.fm=='json' :
                self.fd = open('Jd.json','w',encoding='utf-8')
            elif self.fm=='csv' :
                self.fd = open('Jd.csv','w',encoding='utf-8',newline='')

        def open_browser(self):
            self.browser = webdriver.Chrome()
            self.browser.implicitly_wait(10)
            self.wait = WebDriverWait(self.browser,10)

        def init_variable(self):
            self.data = zip()
            self.isLast = False

        def parse_page(self):
            try:
                skus = self.wait.until(EC.presence_of_all_elements_located((By.XPATH,'//li[@class="gl-item"]')))
                skus = [item.get_attribute('data-sku') for item in skus]
                links = ['https://item.jd.com/{sku}.html'.format(sku=item) for item in skus]
                prices = self.wait.until(EC.presence_of_all_elements_located((By.XPATH,'//div[@class="gl-i-wrap"]/div[3]/strong/i')))
                prices = [item.text for item in prices]
                names = self.wait.until(EC.presence_of_all_elements_located((By.XPATH,'//div[@class="gl-i-wrap"]/div[4]/a/em')))
                names = [item.text for item in names]
                comments = self.wait.until(EC.presence_of_all_elements_located((By.XPATH,'//div[@class="gl-i-wrap"]/div[5]/strong')))
                comments = [item.text for item in comments]
                self.data = zip(links,prices,names,comments)
            except selenium.common.exceptions.TimeoutException:
                print('parse_page: TimeoutException')
                self.parse_page()
            except selenium.common.exceptions.StaleElementReferenceException:
                print('parse_page: StaleElementReferenceException')
                self.browser.refresh()

        def turn_page(self):
            try:
```

```
self.wait.until(EC.element_to_be_clickable((By.XPATH,'//a[@class="pn-next"]'))).click()
                time.sleep(1)
                self.browser.execute_script("window.scrollTo(0,document.body.scrollHeight)")
                time.sleep(2)
            except selenium.common.exceptions.NoSuchElementException:
                self.isLast = True
            except selenium.common.exceptions.TimeoutException:
                print('turn_page: TimeoutException')
                self.turn_page()
            except selenium.common.exceptions.StaleElementReferenceException:
                print('turn_page: StaleElementReferenceException')
                self.browser.refresh()

        def write_to_file(self):
            if self.fm == 'txt':
                for item in self.data:
                    self.fd.write('---------------------------------------\n')
                    self.fd.write('link：' + str(item[0]) + '\n')
                    self.fd.write('price：' + str(item[1]) + '\n')
                    self.fd.write('name：' + str(item[2]) + '\n')
                    self.fd.write('comment：' + str(item[3]) + '\n')
            if self.fm == 'json':
                temp = ('link','price','name','comment')
                for item in self.data:
                    json.dump(dict(zip(temp,item)),self.fd,ensure_ascii=False)
            if self.fm == 'csv':
                writer = csv.writer(self.fd)
                for item in self.data:
                    writer.writerow(item)

        def close_file(self):
            self.fd.close()

        def close_browser(self):
            self.browser.quit()

        def crawl(self):
            self.open_file()
            self.open_browser()
            self.init_variable()
            print('开始爬取')

self.browser.get('https://search.jd.com/Search?keyword=%E6%89%8B%E6%9C%BA&enc=utf-8')
            time.sleep(1)
```

```
            self.browser.execute_script("window.scrollTo(0,document.body.scrollHeight)")
            time.sleep(2)
            count = 0
            while count!=10:
                time.sleep(2)
                count += 1
                print('正在爬取第 ' + str(count) + ' 页.....')
                self.parse_page()
                self.write_to_file()
                self.turn_page()
            self.close_file()
            self.close_browser()
            print('结束爬取')

if __name__ == '__main__':
    spider = JdSpider()
    spider.crawl()
```

```
DevTools listening on ws://127.0.0.1:54819/devtools/browser/91824484-4546-494b-8772-a6c29c616d5a
开始爬取
正在爬取第 1 页.....
正在爬取第 2 页.....
正在爬取第 3 页.....
正在爬取第 4 页.....
正在爬取第 5 页.....
[0409/165055.934:ERROR:stun_port.cc(92)] Binding request timed out from 0.0.0.x:55346 (any)
正在爬取第 6 页.....
正在爬取第 7 页.....
正在爬取第 8 页.....
正在爬取第 9 页.....
正在爬取第 10 页.....
结束爬取
```

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

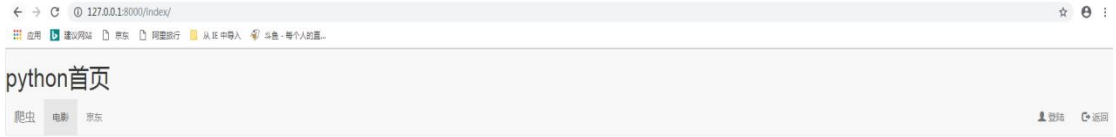
```
----------------------------------------
link：https://item.jd.com/100003484294.html
price：7699.00
name：三星 Galaxy S10 8GB+512GB炭晶黑（SM-G9730）3D超声波屏下指纹超感官全视屏双卡双
comment：2.4万+条评价
----------------------------------------
link：https://item.jd.com/100000177760.html
price：6049.00
name：Apple iPhone XR (A2108) 128GB 黑色 移动联通电信4G手机 双卡双待
comment：87万+条评价
----------------------------------------
link：https://item.jd.com/100003433872.html
price：3298.00
name：【KPL官方比赛用机】vivo iQOO 44W超快闪充 8GB+128GB电光蓝 全面屏拍照手机 骁龙8
comment：7.7万+条评价
----------------------------------------
link：https://item.jd.com/7652013.html
price：1199.00
```

网页展示

index.html（首页）

代码：

```
<head>
    <meta charset="utf-8">
    <title>赵佳豪的导航栏</title>
    <link rel="stylesheet" href="https://cdn.staticfile.org/twitter-bootstrap/3.3.7/css/bootstrap.min.css">
    <script src="https://cdn.staticfile.org/jquery/2.1.1/jquery.min.js"></script>
    <script src="https://cdn.staticfile.org/twitter-bootstrap/3.3.7/js/bootstrap.min.js"></script>
</head>
<body>
<nav class="navbar navbar-default" role="navigation">
    <h1>python 首页</h1>
    <div class="container-fluid">
    <div class="navbar-header">
        <a class="navbar-brand" href="#">爬虫</a>
    </div>
    <div>
        <ul class="nav navbar-nav">
            <li class="active"><a href="http://127.0.0.1:8000/testdb/">电影</a></li>
            <li><a href="#">京东</a></li>

            </li>
        </ul>
        <ul class="nav navbar-nav navbar-right">
                <li><a href="#"><span class="glyphicon glyphicon-user"></span> 登陆</a></li>
                <li><a href="http://127.0.0.1:8000/index/"><span class="glyphicon glyphicon-log-out"></span> 返
回</a></li>
        </ul>
    </div>
    </div>
</nav>
```

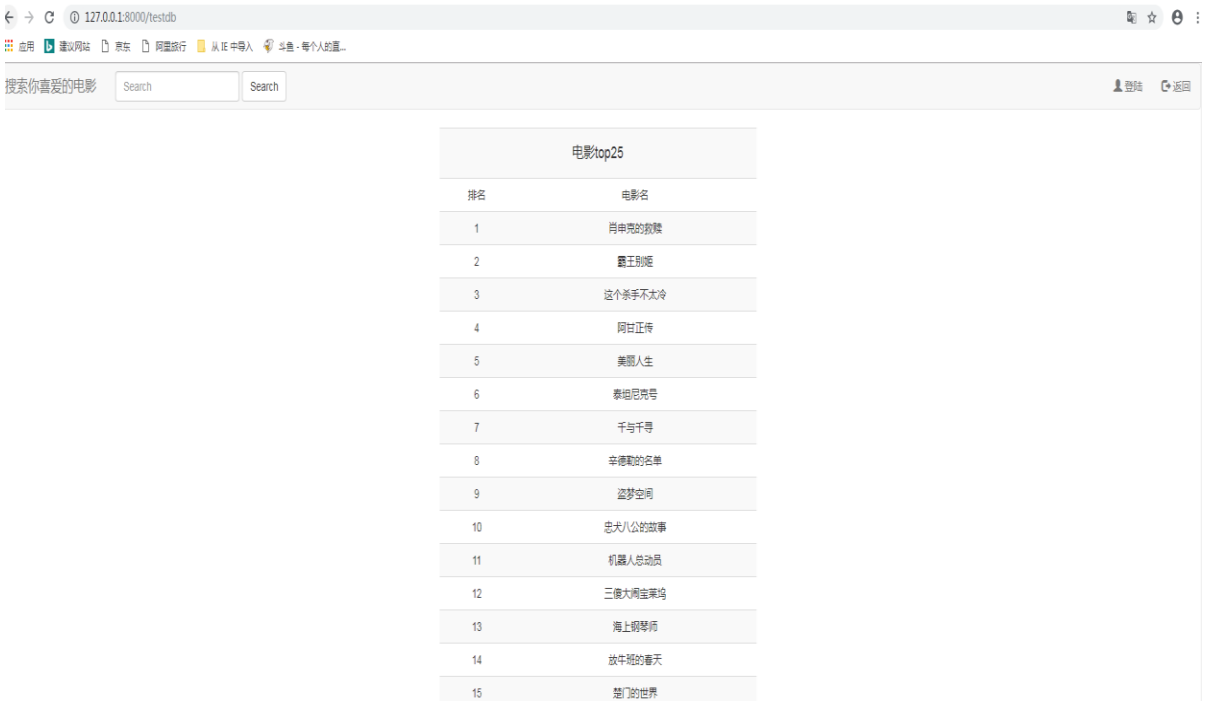


电影 top.html

代码：

```
<!DOCTYPE html>
<html>
```

```
<head>
        <meta charset="utf-8">
        <title>电影排行</title>
        <link rel="stylesheet"
href="https://cdn.staticfile.org/twitter-bootstrap/3.3.7/css/bootstrap.min.css">
        <script src="https://cdn.staticfile.org/jquery/2.1.1/jquery.min.js"></script>
        <script
src="https://cdn.staticfile.org/twitter-bootstrap/3.3.7/js/bootstrap.min.js"></script>
</head>
<body>
       <nav class="navbar navbar-default" role="navigation">
       <div class="container-fluid">
       <div class="navbar-header">
       <a class="navbar-brand" href="#">搜索你喜爱的电影</a>       搜索栏
       </div>
       <form class="navbar-form navbar-left" role="search">
       <div class="form-group">
       <input type="text" class="form-control" placeholder="Search">
       </div>
       <button type="submit" class="btn btn-default">Search</button>
       </form>
       <ul class="nav navbar-nav navbar-right">
       <li><a href="#"><span class="glyphicon glyphicon-user"></span> 登陆</a></li>
       <li><a href="file:///C:/Users/Administrator/Desktop/%E7%88%AC%E8%99%AB/1.html#"><span
class="glyphicon glyphicon-log-out"></span> 返回</a></li>
       </ul>
       </div>
       </nav>
       <div class="table-responsive" style="width:500px;margin:10px auto;text-align: center;">
       <table class="table table-striped table-hover">
       <tr><th colspan="4" style="text-align:center;"><h4>电影 top25</h4></th></tr>
       <tr><td>排名 </td><td>电影名</td></tr>
       {% for movie in list %}
       <tr>
       <td>{{movie.id}}</td>
       <td>{{movie.name}}</td>
       </tr>
       {% endfor %}
       </table>
       </div>
       </body>
```



Scrapy 爬取豆瓣 top250(名字，导演，评分，名言)

Douban.py

```
# -*- coding: utf-8 -*-

import scrapy

from scrapy.http import Request

from scrapy.selector import Selector

from doubanmovie.items import DoubanmovieItem

from urllib.parse import urljoin



class Douban(scrapy.spiders.Spider):

    name = "douban"

    allowed_domains = ["douban.com"]

    # redis_key = 'douban:start_urls'

    start_urls = ['https://movie.douban.com/top250']



    def parse(self, response):

        item = DoubanmovieItem()
```

```
        selector = Selector(response)
        Movies = selector.xpath('//div[@class="info"]')
        for eachMovie in Movies:
            title = eachMovie.xpath('div[@class="hd"]/a/span/text()').extract() # 多个 span 标签
            fullTitle = "".join(title) # 将多个字符串无缝连接起来
            movieInfo = eachMovie.xpath('div[@class="bd"]/p/text()').extract()
            star = eachMovie.xpath('div[@class="bd"]/div[@class="star"]/span/text()').extract()[0]
            quote = eachMovie.xpath('div[@class="bd"]/p[@class="quote"]/span/text()').extract()
            # quote 可能为空，因此需要先进行判断
            if quote:
                quote = quote[0]
            else:
                quote = ''
            item['title'] = fullTitle
            item['movieInfo'] = ';'.join(movieInfo)
            item['star'] = star
            item['quote'] = quote
            yield item
        nextLink = selector.xpath('//span[@class="next"]/link/@href').extract()
        # 第 10 页是最后一页，没有下一页的链接
        if nextLink:
            nextLink = nextLink[0]
            yield Request(urljoin(response.url, nextLink), callback=self.parse)
```

items.py

```
# -*- coding: utf-8 -*-

# Define here the models for your scraped items
#
# See documentation in:
# https://doc.scrapy.org/en/latest/topics/items.html

import scrapy


class DoubanmovieItem(scrapy.Item):
    title = scrapy.Field()  # 电影名字
    movieInfo = scrapy.Field()  # 电影的描述信息，包括导演、主演、电影类型等等
    star = scrapy.Field()  # 电影评分
    quote = scrapy.Field()  # 电影中最经典或者说脍炙人口的一句话
    pass
```

ID	title	movieInfo	star	quote
235	黄金三镖	导演: Sergio Leone 主演: (	9.1	最棒的西部片。
236	秒速5厘米	导演: 新海诚 Makoto Shinka	8.3	青春就是放弃和怀念。
237	疯狂的麦	导演: 乔治·米勒 George Mill	8.6	“多么美好的一天！”轰轰轰砰咚，啪哒哒哒轰隆隆
238	非常嫌疑	导演: 布莱恩·辛格 Bryan Sing	8.6	我不信仰上帝，但我敬畏上帝。
239	我爱你 /	导演: 秋昌民 Chang-min Ch	9	你要相信，这世上真的有爱存在，不管在什么年纪
240	国王的演	导演: 汤姆·霍珀 Tom Hoope	8.4	皇上无话儿。
241	卡萨布兰	导演: 迈克尔·柯蒂兹 Michael	8.6	世界上有那么多女人那么多酒馆，但她偏偏走进我的
242	千钧一发	导演: 安德鲁·尼科尔 Andrew	8.7	一部能引人思考的科幻励志片。
243	遗愿清单	导演: 罗伯·莱纳 Rob Reiner	8.6	用剩余不多的时间，去燃烧整个生命。
244	美国丽人	导演: 萨姆·门德斯 Sam Men	8.5	每个人的内心都是深不可测的大海。
245	驴得水 /	导演: 周申 Shen Zhou / 刘露	8.3	过去的如果就让它过去了，未来只会越来越糟！
246	四个春天	导演: 陆庆屹 Lu Qing Yi 主	8.9	来也匆匆去也匆匆，就这样风雨兼程。
247	新世界 /	导演: 朴勋政 Hoon-jung Par	8.7	要做就做得狠一点，这样才能活下去。
248	荒岛余生	导演: 罗伯特·泽米吉斯 Robe	8.5	一个人的独角戏。
249	釜山行 /	导演: 延尚昊 Sang-ho Yeon	8.4	
250	枪火 / 鎗	导演: 杜琪峰 Johnnie To 主	8.7	一群演技精湛的戏骨，奉献出一个精致的黑帮小品，

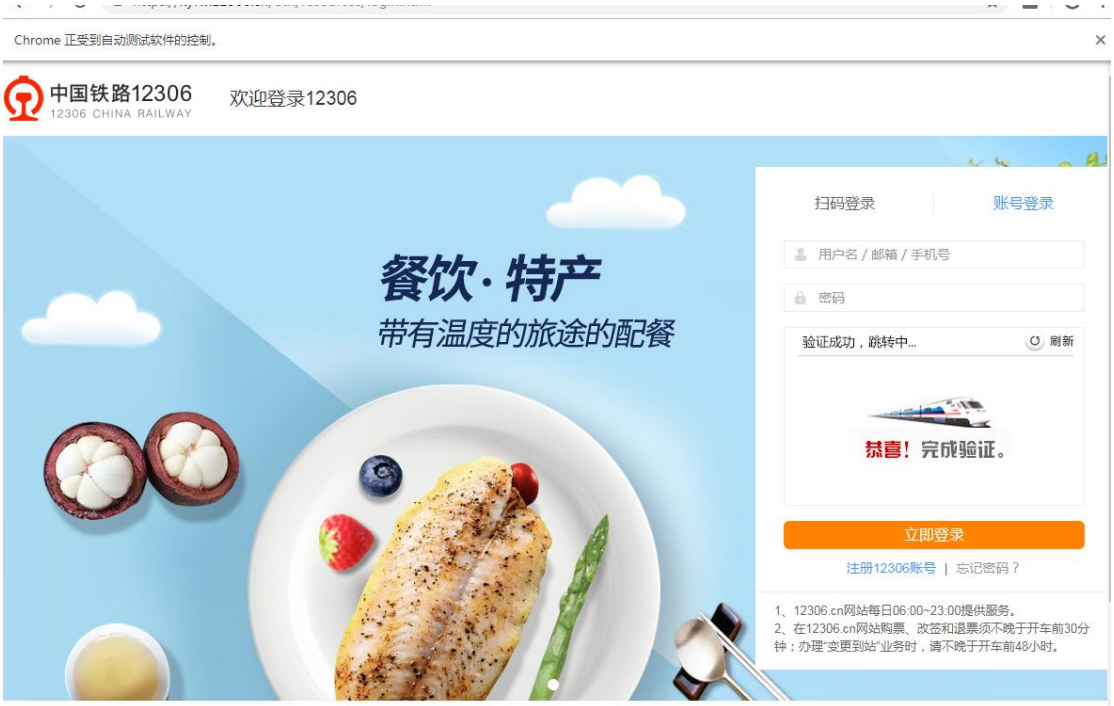
12306 自动登录：

```
from selenium import webdriver
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
from selenium.webdriver.common.action_chains import ActionChains
import requests
import base64
import re
import time


class Demo():
    def __init__(self):
        self.coordinate=[[-105,-20],[-35,-20],[40,-20],[110,-20],[-105,50],[-35,50],[40,50],[110,50]]
    def login(self):
        login_url="https://kyfw.12306.cn/otn/resources/login.html"
        driver = webdriver.Chrome()
        driver.set_window_size(1200, 900)
        driver.get(login_url)
        time.sleep(1)
        account=driver.find_element_by_class_name("login-hd-account")
        account.click()
```

```
        userName=driver.find_element_by_id("J-userName")
        userName.send_keys("15888601085")
        password=driver.find_element_by_id("J-password")
        password.send_keys("ljh13127984971")
        self.driver=driver
    def getVerifyImage(self):
        try:
            img_element =WebDriverWait(self.driver, 100).until(
                EC.presence_of_element_located((By.ID, "J-loginImg"))
                )
        except Exception :
            print(u"网络开小差,请稍后尝试")
        base64_str=img_element.get_attribute("src").split(",")[-1]
        imgdata=base64.b64decode(base64_str)
        with open('d:\\verify.jpg','wb') as file:
            file.write(imgdata)
        self.img_element=img_element
    def getVerifyResult(self):
        driver1 = webdriver.Chrome()
        driver1.get('http://littlebigluo.qicp.net:47720/')
        upload = driver1.find_elements_by_tag_name('input')[0]
        time.sleep(3)
        upload.send_keys('d:\\verify.jpg')  # send_keys
        submit = driver1.find_elements_by_tag_name('input')[1].click()
        response=driver1.find_element_by_xpath("/html/body/p[1]/font/font/b").text
        result=[]
        for i in response.split(" "):
            result.append(int(i)-1)
        self.result=result
        driver1.close
        print(result)
    def moveAndClick(self):
        try:
            Action=ActionChains(self.driver)
            for i in self.result:
                Action.move_to_element(self.img_element).move_by_offset(self.coordinate[i][0],self.coordina
te[i][1]).click()
            Action.perform()
        except Exception as e:
            print(e)
    def submit(self):
        self.driver.find_element_by_id("J-login").click()
    def __call__(self):
        self.login()
```

```
        time.sleep(3)
        self.getVerifyImage()
        time.sleep(1)
        self.getVerifyResult()
        time.sleep(1)
        self.moveAndClick()
        time.sleep(1)
        self.submit()
        time.sleep(10000)

Demo()
```

图片验证使用网站：http://littlebigluo.qicp.net:47720/



将文件代码上传到 github

https://github.com/zjhshiwo/python