Python爬虫期中作业+期末作业

1.

静态爬虫：1.引用requests，BeautifulSoup库。 2.创建列表来对爬取的数据进行存储。3.发送URL 请求 获取豆瓣网页URL采用get方法获取 4.用find\_all方法来得到所要爬取的信息。4.将数据存储到列表中 5.最后输出爬取的数据

Django项目：1.创建项目 Movies ，以及APP项目 Movie 2.应用数据库文件sqlite 3.创建超级管理员superuser 4.. 定义模型 models

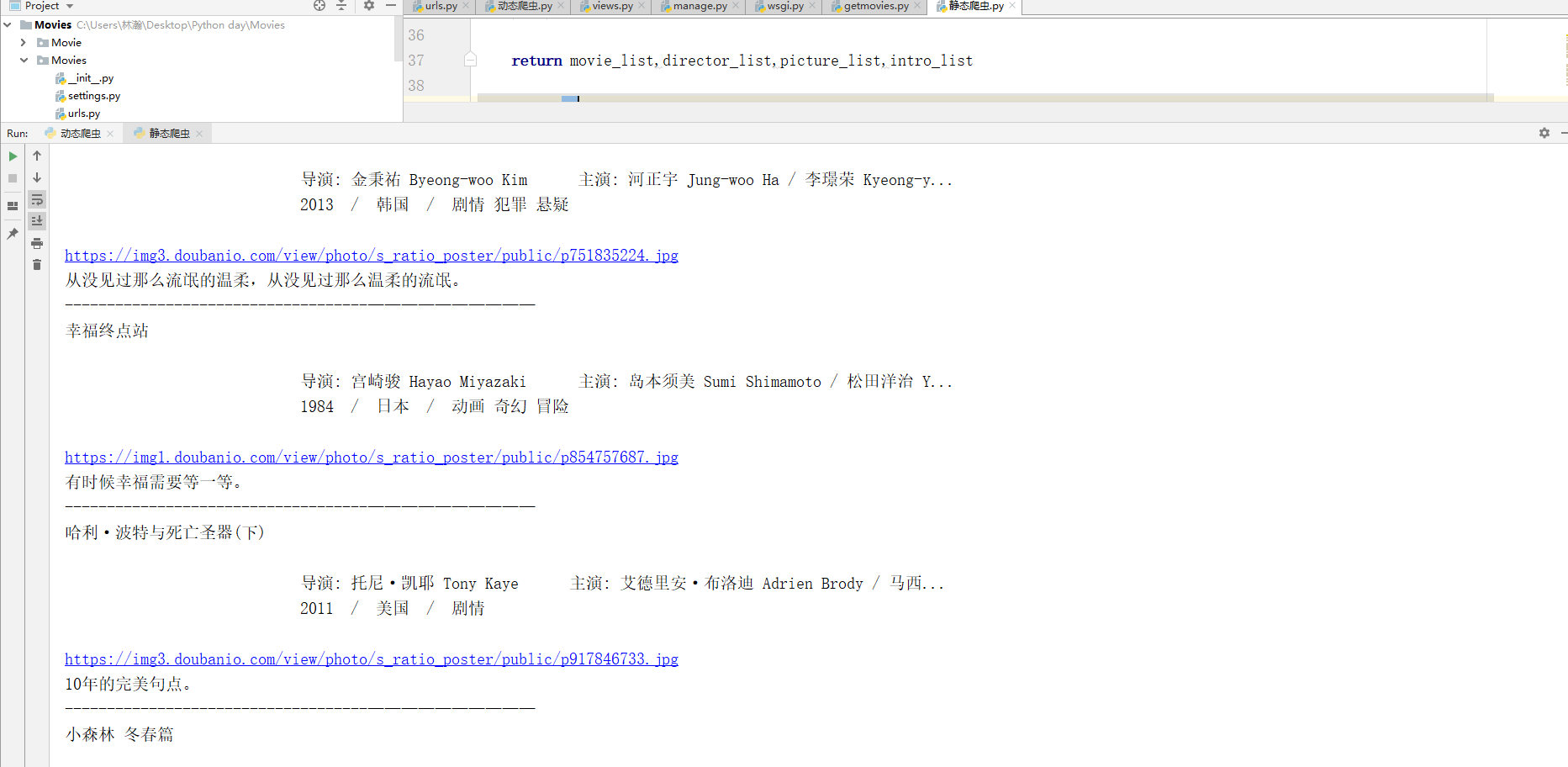
movieType，包含title（电影名），director（详细信息）,img（图片）,和dec（电影简介）。5.将模型models同步到数据库 ，并对admin进行编辑，将应用展现在后台管理界面。 编写views方法 object.get获取记录，将get\_movie 中爬取的电影信息存储到后台数据库中。6.编写templates 中的html模板文件，编写总路由编写一个网址，将后台数据库信息展示在该网址上。

动态爬虫：1.引入selenium库 ，对浏览器搜索关键字，得到查询后的商品列表

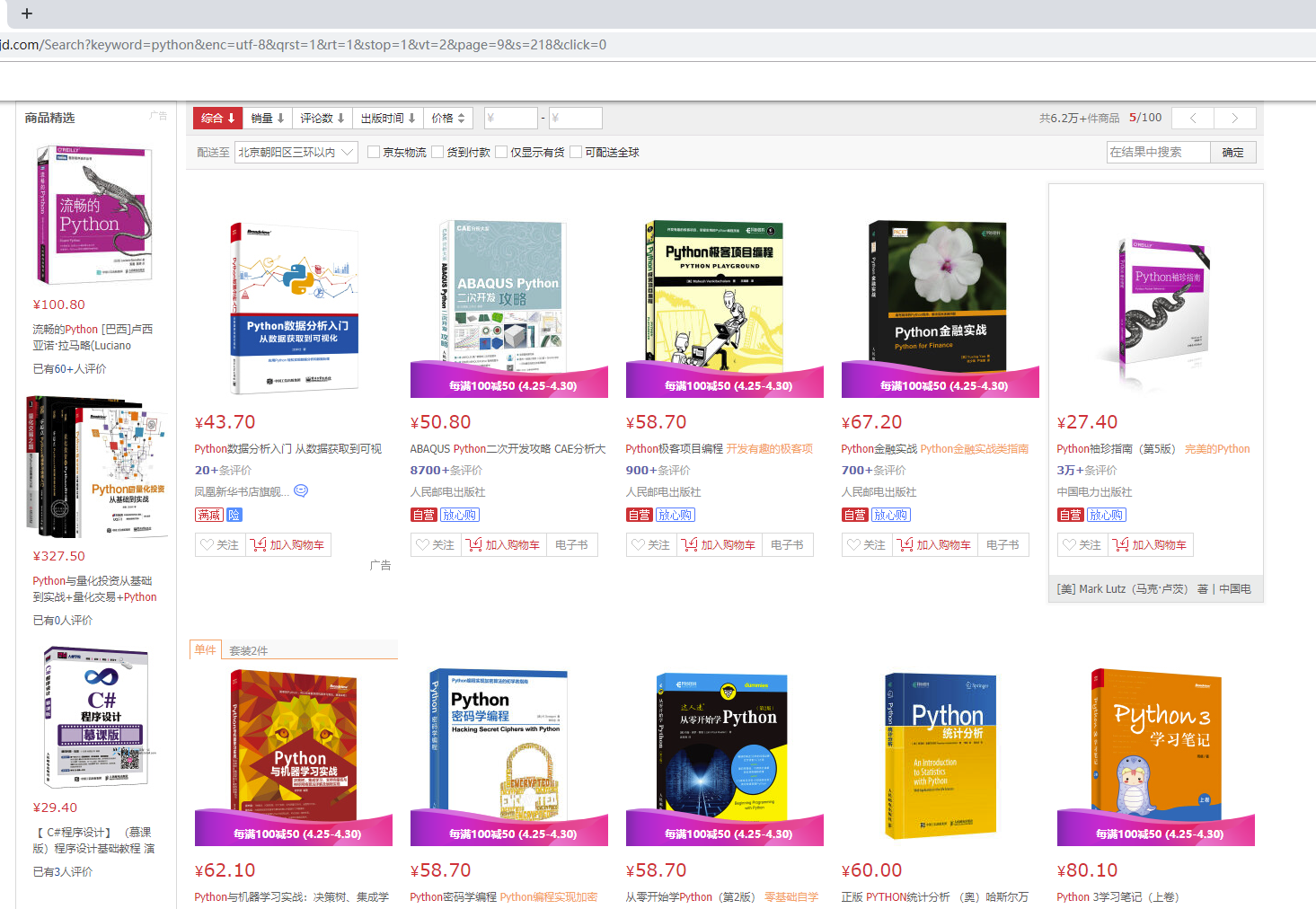
2.分析页码，得到商品的页码数，模拟翻页，下拉，得到翻页之后的商品列表。

3.利用lxml中的etree方法分析源码，用Xpath获取商品的各项信息数据，并输出。

2.效果截图：  
静态爬虫：



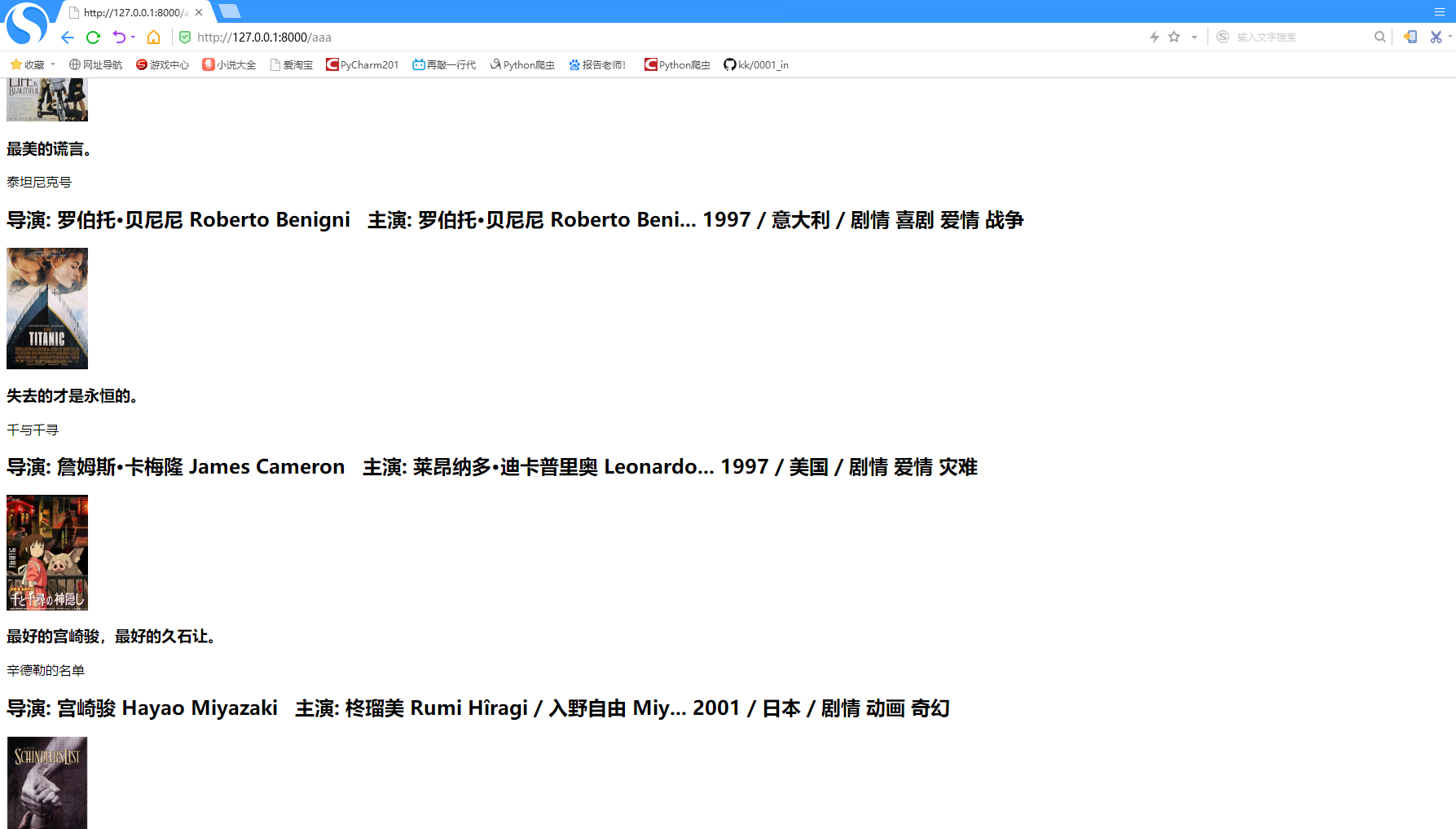
动态爬虫：





静态爬虫显示到Django网站上：

网页效果图：



Django 默认的数据库：



12306自动登录：

主要工具：

Selenium、python、request、超级鹰,

基本思路

1. 使用selenium工具访问12306
2. 自动输入用户名和密码
3. 获取验证码图片
4. 使用超级鹰提供的验证码识别

核心代码：

图片的获取：

**def** get\_pic(self):  
 tag = self.browser.find\_element\_by\_class\_name(**'imgCode'**)  
 temp = tag.get\_attribute(**'src'**)  
 b64\_pic = re.sub(**r'data:image/jpg;base64,'**, **''**, temp)  
 pic = base64.b64decode(b64\_pic)  
 **return** pic

验证码图片的发布：

**def** PostPic(self, im, codetype):  
 params = {  
 **'codetype'**: codetype,  
 }  
 params.update(self.base\_params)  
 files = {**'userfile'**: (**'12306.jpg'**, im)}  
 r = requests.post(**'http://upload.chaojiying.net/Upload/Processing.php'**, data=params, files=files, headers=self.headers)  
 **return** r.json()

登录函数：

**def** login(self):  
 self.browser = webdriver.Chrome()  
 self.browser.get(self.url)  
 time.sleep(2)  
 login\_select = self.browser.find\_element\_by\_class\_name(**'login-hd-account'**)  
 login\_select.click()  
 user = self.browser.find\_element\_by\_id(**'J-userName'**)  
 word = self.browser.find\_element\_by\_id(**'J-password'**)  
 user.send\_keys(self.username)  
 word.send\_keys(self.password)

点击事件：

**def** click(self, j):  
 temp = j.get(**'pic\_str'**)  
 locations = [list(map(int, i.split(**','**))) **for** i **in** temp.split(**'|'**)]  
 **for** location **in** locations:  
 ActionChains(self.browser).move\_to\_element\_with\_offset(self.browser.find\_element\_by\_class\_name(**'imgCode'**),  
 location[0], location[1]).click().perform()  
 time.sleep(5)  
 self.browser.find\_element\_by\_id(**'J-login'**).click()  
  
**def** get\_cookies(self):  
 **return** self.browser.get\_cookies()

主函数：

**if** \_\_name\_\_ == **'\_\_main\_\_'**:  
 a = Login(**'https://kyfw.12306.cn/otn/resources/login.html'**, **'13958920775'**, **'s13958920775'**)  
 a.login()  
 time.sleep(2)  
 chaojiying = Chaojiying\_Client(**'s674560845'**, **'s13958920775'**, **'900033'**)  
 im = a.get\_pic()  
 print(im)  
 z = chaojiying.PostPic(im, 9004)  
 print(z)  
 a.click(z)  
 print(a.get\_cookies())

效果截图：



