温州大学瓯江学院

爬虫期中 实验报告

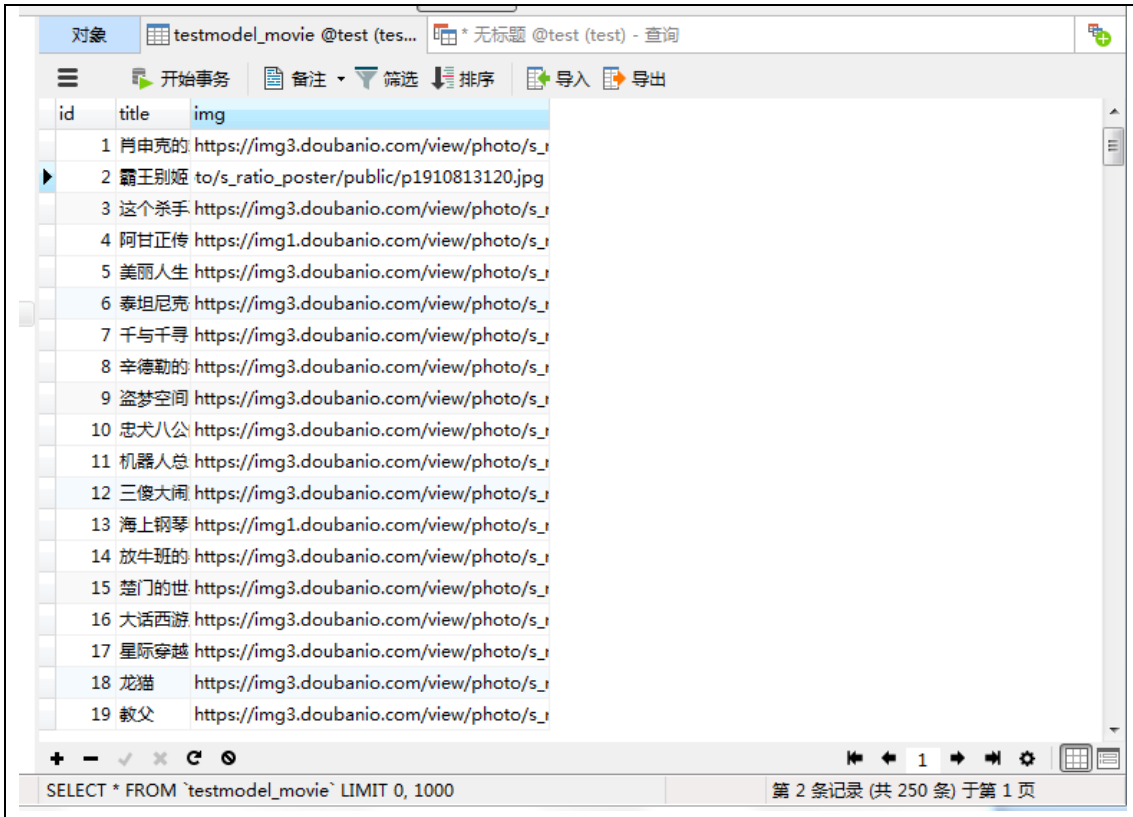
实验名称：	爬虫				
班　　级：	16 计算机三班	姓　　名：	童骋	学　　号：	14219110112
实验地点：	7-403	日　　期：	2019.4.23		

一、爬取豆瓣电影：

代码：

```
import requests
from lxml import etree
import urllib.request
import pymysql
from bs4 import BeautifulSoup


conn=pymysql.connect(host='localhost',user='root',passwd='1234',db='test',charset="utf8")
cursor=conn.cursor()
headers={'user-agent':'Mozilla/5.0(Windows NT 6.1;Win64;x64)AppleWebKit/537.36(KHTML,like
Gecko) Chrome/52.0.2743.82 Safari/537.36','Host':'movie.douban.com'}
for i in range(0,10):
        url = 'https://movie.douban.com/top250?start='+str(25*i)
        r = requests.get(url,headers=headers)
        html = etree.HTML(r.text)
        datas=html.xpath('//ol[@class="grid_view"]/li')
        a=0
        for data in datas:
                title=data.xpath('div/div[2]/div[@class="hd"]/a/span[1]/text()')
                img=data.xpath('div/div[1]/a/img/@src')
                urllib.request.urlretrieve(img[0],filename="G:top250/"+str(i*25+a+1)+".jpg")
                a+=1
                cursor.execute("insert into testmodel_movie(title,img) values(%s,%s)",(title,img))
cursor.close()
conn.commit()
conn.close()
```

数据库截图：



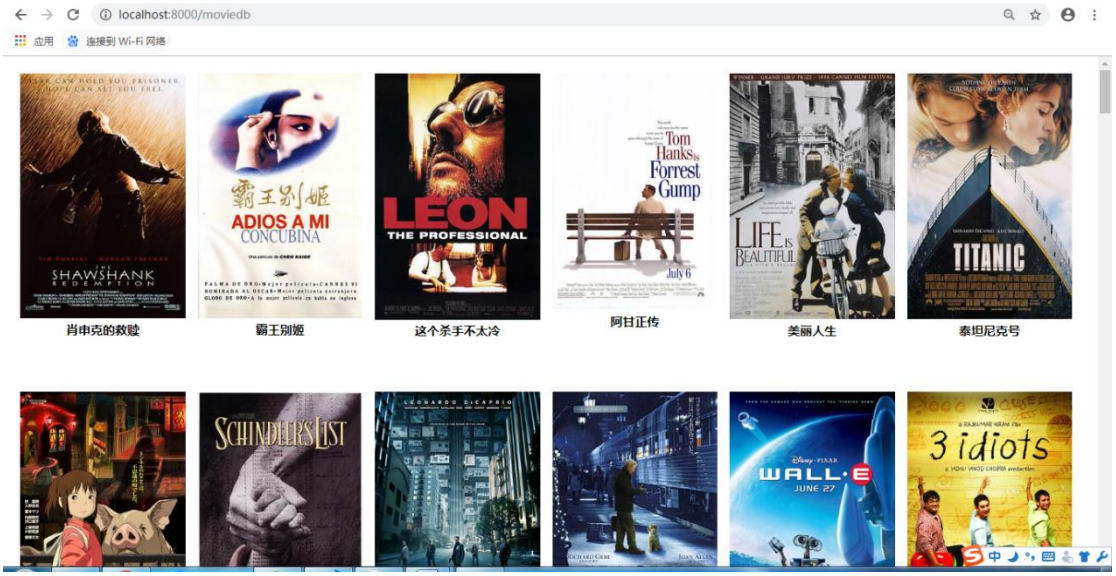
对象 testmodel_movie @test (tes... * 无标题 @test (test) - 查询

开始事务 备注 筛选 排序 导入 导出

id	title	img
1	肖申克的	https://img3.doubanio.com/view/photo/s_
2	霸王别姬	to/s_ratio_poster/public/p1910813120.jpg
3	这个杀手	https://img3.doubanio.com/view/photo/s_
4	阿甘正传	https://img1.doubanio.com/view/photo/s_
5	美丽人生	https://img3.doubanio.com/view/photo/s_
6	泰坦尼克	https://img3.doubanio.com/view/photo/s_
7	千与千寻	https://img3.doubanio.com/view/photo/s_
8	辛德勒的	https://img3.doubanio.com/view/photo/s_
9	盗梦空间	https://img3.doubanio.com/view/photo/s_
10	忠犬八公	https://img3.doubanio.com/view/photo/s_
11	机器人总	https://img3.doubanio.com/view/photo/s_
12	三傻大闹	https://img3.doubanio.com/view/photo/s_
13	海上钢琴	https://img1.doubanio.com/view/photo/s_
14	放牛班的	https://img3.doubanio.com/view/photo/s_
15	楚门的世	https://img3.doubanio.com/view/photo/s_
16	大话西游	https://img3.doubanio.com/view/photo/s_
17	星际穿越	https://img3.doubanio.com/view/photo/s_
18	龙猫	https://img3.doubanio.com/view/photo/s_
19	教父	https://img3.doubanio.com/view/photo/s_

SELECT * FROM `testmodel_movie` LIMIT 0, 1000　　第 2 条记录 (共 250 条) 于第 1 页

界面截图：



二、爬取天气：

代码：

```
from bs4 import BeautifulSoup
from bs4 import UnicodeDammit
import urllib.request
import pymysql

conn=pymysql.connect(host='localhost',user='root',passwd='1234',db='test',charset="utf8")
cursor=conn.cursor()

headers={'user-agent':'Mozilla/5.0(Windows;U;Windows NT 6.0
x64;en-us;rv:1.9pre)Gecko/2008072421 MineField/3.0.2pre'}
```

```
citycode={"北京":"101010100","上海":"101020100","广州":"101280101","深圳":"101280601"}
for city in citycode:
    url="http://www.weather.com.cn/weather/"+citycode[city]+".shtml"
    try:
        req=urllib.request.Request(url,headers=headers)
        data=urllib.request.urlopen(req)
        data=data.read()
        dammint=UnicodeDammit(data,["utf-8","gbk"])
        data=dammint.unicode_markup
        soup=BeautifulSoup(data,"lxml")
        lis=soup.select("ul[class='t clearfix'] li")
        n=0
        for li in lis:
            try:
                date=li.select('h1')[0].text
                print(date)
                weather=li.select("p[class='wea']")[0].text
                if n>0:
                    temp=li.select("p[class='tem']          span")[0].text+"/"+li.select("p[class='tem']
i")[0].text
                else:
                    temp=li.select("p[class='tem'] i")[0].text
                cursor.execute("insert          into          testmodel_weather(city,date,weather,temp)
values(%s,%s,%s,%s)",(city,date,weather,temp))
                n=n+1
            except Exception as err:
                print(err)
    except Exception as err:
        print(err)
cursor.close()
conn.commit()
conn.close()
```

数据库截图：

对象 testmodel_weather @test (t...

开始事务 备注 筛选 排序 导入 导出

id	city	date	weather	temp
1	北京	25日（今	多云	8℃
2	北京	26日（明	晴转多云	21℃/10℃
3	北京	27日（后	多云	18℃/7℃
4	北京	28日（周	多云	21℃/9℃
5	北京	29日（周	多云转小雨	24℃/13℃
6	北京	30日（周	多云	26℃/14℃
7	北京	1日（周三	多云	24℃/14℃
8	上海	25日（今	阴	14℃
9	上海	26日（明	多云转晴	18℃/12℃
10	上海	27日（后	多云	18℃/14℃
11	上海	28日（周	小雨转多云	23℃/17℃
12	上海	29日（周	中雨转阴	24℃/17℃
13	上海	30日（周	阴转小雨	22℃/17℃
14	上海	1日（周三	小雨转多云	23℃/17℃
15	深圳	25日（今	小雨	25℃
16	深圳	26日（明	中雨转大雨	29℃/24℃
17	深圳	27日（后	大雨转雷阵雨	27℃/23℃
18	深圳	28日（周	雷阵雨	28℃/24℃
19	深圳	29日（周	雷阵雨	30℃/25℃

1

SELECT * FROM `testmodel_weather` LIMIT 0, 1000　　第 13 条记录 (共 28 条) 于第 1 页

界面截图：

localhost:8000/weatherdb

应用　连接到 Wi-Fi 网络

城市	日期	天气	温度
北京	25日（今天）	多云	8℃
北京	26日（明天）	晴转多云	21℃/10℃
北京	27日（后天）	多云	18℃/7℃
北京	28日（周日）	多云	21℃/9℃
北京	29日（周一）	多云转小雨	24℃/13℃
北京	30日（周二）	多云	26℃/14℃
北京	1日（周三）	多云	24℃/14℃
上海	25日（今天）	阴	14℃
上海	26日（明天）	多云转晴	18℃/12℃
上海	27日（后天）	多云	18℃/14℃
上海	28日（周日）	小雨转多云	23℃/17℃
上海	29日（周一）	中雨转阴	24℃/17℃
上海	30日（周二）	阴转小雨	22℃/17℃
上海	1日（周三）	小雨转多云	23℃/17℃
深圳	25日（今天）	小雨	25℃
深圳	26日（明天）	中雨转大雨	29℃/24℃
深圳	27日（后天）	大雨转雷阵雨	27℃/23℃
深圳	28日（周日）	雷阵雨	28℃/24℃
深圳	29日（周一）	雷阵雨	30℃/25℃
深圳	30日（周二）	雷阵雨转暴雨	30℃/23℃
深圳	1日（周三）	暴雨转阵雨	27℃/23℃
广州	25日（今天）	雷阵雨	24℃
广州	26日（明天）	中雨转中到大雨	28℃/24℃
广州	27日（后天）	中到大雨转雷阵雨	28℃/24℃
广州	28日（周日）	雷阵雨	29℃/25℃
广州	29日（周一）	雷阵雨转中雨	30℃/25℃
广州	30日（周二）	中雨转大到暴雨	30℃/22℃
广州	1日（周三）	大到暴雨转多云	26℃/20℃

三、爬取淘宝书包：

代码：

```
import re
import requests
import pymysql


conn=pymysql.connect(host='localhost',user='root',passwd='1234',db='test',charset='utf8')
cursor=conn.cursor()
def getHTMLText(url):
        try:
                r=requests.get(url,timeout=30)
```

```
                r.raise_for_status()
                r.encoding=r.apparent_encoding
                return r.text
        except:
                return ""

def parsePage(ilt,html):
        try:
                plt=re.findall(r'\"view_price\"\:\"[\d\.]*\"',html)
                tlt=re.findall(r'\"raw_title\"\:\".*?\"',html)
                for i in range(len(plt)):
                        price=eval(plt[i].split(':')[1])
                        title=eval(tlt[i].split(':')[1])
                        ilt.append([price,title])
                        cursor.execute("insert into testmodel_shubao(title,price) values(%s,%s)",(title,price))
        except:
                print("")
cursor.close()
conn.commit()
conn.close()

def printGoodsList(ilt):
        tplt="{:4}\t{:8}\t{:16}"
        print(tplt.format("序号","价格","商品名称"))
        count=0
        for g in ilt:
                count=count+1
                print(tplt.format(count,g[0],g[1]))

def main():
        goods="书包"
        depth=2
        start_url='https://s.taobao.com/search?q='+goods
        infoList=[]
        for i in range(depth):
                try:
                        url=start_url+'&s='+str(44*i)
                        html=getHTMLText(url)
                        parsePage(infoList,html)
                except:
                        continue
        printGoodsList(infoList)
main()
```

数据库截图：

对象 testmodel_shubao @test (te...

开始事务 备注 筛选 排序 导入 导出

id	price	title
1	45.80	小学生书包男生1-3-4-6年级6-12周岁儿童
2	39.90	迪卡侬双肩包运动背包男女健身包书包儿童学生户外旅行包KIPSTA
3	119.00	kk树书包小学生女孩6-12周岁儿童1-3-6年级女童双肩背包护脊减负
4	499.00	Fjallraven/北极狐双肩包kanken classic书包女户外旅行背包23510
5	129.00	小米双肩包简约休闲多功能书包男女笔记本电脑包时尚潮流旅行背包
6	258.00	电视剧款JanSport旗舰店官网杰斯伯双肩包时尚女书包背包男大容量
7	348.00	爆款anello官方旗舰店日本ins潮风双肩女背包男离家出走包书包
8	199.00	小米 米兔儿童书包 6-12岁男女小学生潮双肩背包幼儿园大容量背包
9	79.00	双肩包男士背包大容量旅行包电脑休闲女时尚潮流高中初中学生书包
10	109.00	七匹狼商务双肩包男书包中学生女电脑包旅行包休闲男士背包大容量
11	148.00	佑一良品男士背包双肩包男韩版大学生书包男时尚潮流大容量旅行包
12	69.00	巴布豆旗舰店书包1-3年级护脊减负儿童书包男4-6小学生书包轻便
13	299.00	BOPAI博牌电脑背包男户外旅行休闲双肩包商务书包出差多功能男包
14	49.00	小学生书包6-12周岁 女儿童双肩包 3-5年级女童背包 1-3年级女孩
15	45.80	儿童书包小学生男童1-3年级6-12周岁4-6年级男孩双肩背包轻便减负
16	59.80	商务背包男士双肩包韩版潮流旅行包休闲女学生书包简约时尚电脑包
17	168.00	双肩包男书包男士时尚潮流青年休闲简约潮牌旅行背包大学生电脑包
18	69.00	迪士尼书包小学生男女1-3-4-6年级米奇减负背包儿童书包8-10-12岁
19	119.00	巴朗商务双肩包休闲时尚潮流大学生书包15.6寸电脑包男士背包男潮

SELECT * FROM `testmodel_shubao` LIMIT 0, 1000　　第 5 条记录 (共 92 条) 于第 1 页

界面截图：

Blog Template for Bootstrap

localhost:8000/shubaodb

应用　连接到 Wi-Fi 网络

标题：小学生书包男生1-3-4-6年级6-12周岁儿童

价格：45.80

标题：迪卡侬双肩包运动背包男女健身包书包儿童学生户外旅行包KIPSTA

价格：39.90

标题：kk树书包小学生女孩6-12周岁儿童1-3-6年级女童双肩背包护脊减负

价格：119.00

标题：Fjallraven/北极狐双肩包kanken classic书包女户外旅行背包23510

价格：499.00

标题：小米双肩包简约休闲多功能书包男女笔记本电脑包时尚潮流旅行背包

价格：129.00

标题：电视剧款JanSport旗舰店官网杰斯伯双肩包时尚女书包背包男大容量

价格：258.00

标题：爆款anello官方旗舰店日本ins潮风双肩女背包男离家出走包书包

四、爬取京东手机：

代码：

```
from selenium import webdriver
from selenium.webdriver.chrome.options import Options
import urllib.request
import threading
import pymysql
import os
import datetime
```

```
from selenium.webdriver.common.keys import Keys
import time

class MySpider:
    headers= {
        "User-Agent": "Mozilla/5.0 (Windows; U; Windows NT 6.0 x64; en-US; rv:1.9pre)Gecko/2008072421 Minefield/3.0.2pre"
    }

    imagePath = "download"

    def startUp(self,url,key):
        chrome_options=Options()
        chrome_options.add_argument('--headless')
        chrome_options.add_argument('--disable-gpu')
        self.driver = webdriver.Chrome(chrome_options=chrome_options)
        self.threads=[]
        self.No=0
        self.imgNo=0
        try:

self.con=pymysql.connect(host='localhost',user='root',passwd=1234,db=test,charset="utf8")
            self.cursor=self.con.cursor()
            try:
                self.cursor.execute("drop table testmodel_phone ")
            except:
                pass
            try:
                sql="create table testmodel_phone(mNo varchar(32) primary key,mMark varchar(256),mPrice varchar(32),mNote varchar(1024),mFile varchar(256))"
                self.cursor.execute(sql)
            except:
                pass
        except Exception as err:
            print(err)

        try:
            if not os.path.exists(MySpider.imagePath):
                os.mkdir(MySpider.imagePath)
            images=os.listdir(MySpider.imagePath)
            for img in images:
                s=os.path.join(MySpider.imagePath,img)
                os.remove(s)
        except Exception as err:
            print(err)

        self.driver.get(url)
        keyInput=self.driver.find_element_by_id("key")
```

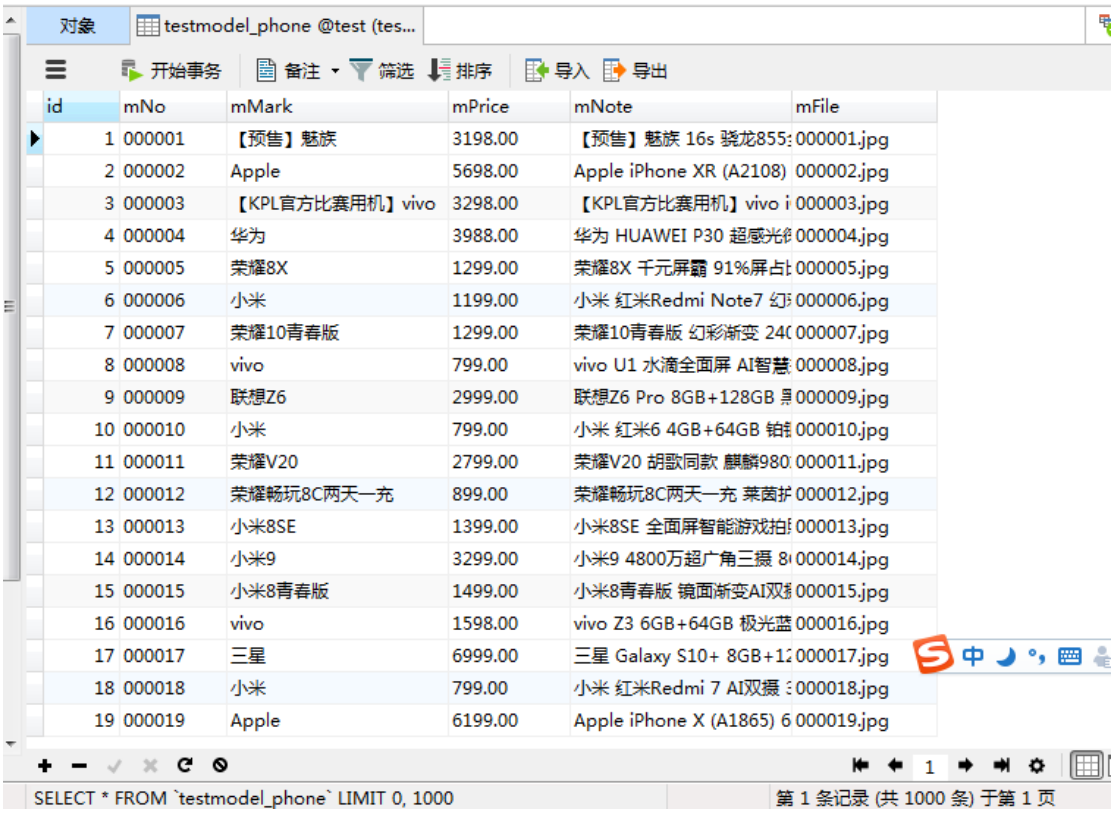
```
        keyInput.send_keys(key)
        keyInput.send_keys(Keys.ENTER)

    def closeUp(self):
        try:
            self.con.commit()
            self.con.close()
            self.driver.close()
        except Exception as err:
            print(err)

    def showDB(self):
        try:
            con=pymysql.connect(host='localhost',user='root',passwd='1234',db=test,charset="utf8")
            cursor=con.cursor()
            print("%-8s %-16s %-8s %-16s %s" % ("No","Mark","Price","Image","Note"))
            cursor.execute("select mNo,mMark,mPrice,mFile,mNote from phones order by mNo")
            rows=cursor.fetchall()
            for row in rows:
                print("%-8s %-16s %-8s %-16s %s" % (row[0],row[1],row[2],row[3],row[4]))
            con.close()
        except Exception as err:
            print(err)


    def download(self,src1,src2,mFile):
        data=None
        if src1:
            try:
                req=urllib.request.Request(src1,headers=MySpider.headers)
                resp=urllib.request.urlopen(req,timeout=400)
                data=resp.read()
            except:
                pass
        if not data and src2:
            try:
                req=urllib.request.Request(src2,headers=MySpider.headers)
                resp=urllib.request.urlopen(req,timeout=400)
                data=resp.read()
            except:
                pass
        if data:
            fobj=open(MySpider.imagePath+"\\"+mFile,"wb")
            fobj.write(data)
            fobj.close()
            print("download",mFile)

    def processSpider(self):
```

```
        try:
            time.sleep(2)
            print(self.driver.current_url)
            lis=self.driver.find_elements_by_xpath("//div[@id='J_goodsList']//li[@class='gl-item']")
            for li in    lis:
                try:

src1=li.find_element_by_xpath(".//div[@class='p-img']//a//img").get_attribute("src")
                except:
                    src1=""

                try:

src2=li.find_element_by_xpath(".//div[@class='p-img']//a//img").get_attribute("data-lazy-img")
                except:
                    src2=""

                try:
                    price=li.find_element_by_xpath(".//div[@class='p-price']//i").text
                except:
                    price="0"

                try:
                    note=li.find_element_by_xpath(".//div[@class='p-name
p-name-type-2']//em").text
                    mark=note.split(" ")[0]
                    mark=mark.replace("爱心东东\n","")
                    mark=mark.replace(",","")
                    note=note.replace("爱心东东\n","")
                    note=note.replace(",","")
                except:
                    note=""
                    mark=""

                self.No=self.No+1
                no=str(self.No)
                while len(no)<6:
                    no="0"+no
                print(no,mark,price)

                if src1:
                    src1=urllib.request.urljoin(self.driver.current_url,src1)
                    p=src1.rfind(".")
                    mFile=no+src1[p:]
                elif src2:
                    src2=urllib.request.urljoin(self.driver.current_url,src2)
                    p=src2.rfind(".")
```

```
                        mFile=no+src2[p:]
                    if src1 or src2:
                        T=threading.Thread(target=self.download,args=(src1,src2,mFile))
                        T.setDaemon(False)
                        T.start()
                        self.threads.append(T)
                    else:
                        mFile=""

                    self.cursor.execute("insert into jd_test(mNo,mMark,mPrice,mNote,mFile)
values(%s,%s,%s,%s,%s)",(no,mark,price,note,mFile))

                try:
                    self.driver.find_element_by_xpath("//span[@class='p-num']//a[@class='pn-next
disabled']")
                except:

nextPage=self.driver.find_element_by_xpath("//span[@class='p-num']//a[@class='pn-next']")
                    nextPage.click()
                    self.processSpider()
            except Exception as err:
                print(err)

    def executeSpider(self,url,key):
        starttime=datetime.datetime.now()
        print("开始...")
        self.startUp(url,key)
        self.processSpider()
        self.closeUp()
        for t in self.threads:
            t.join()
        print("爬取完成....")
        endtime=datetime.datetime.now()
        elapsed=(endtime-starttime).seconds
        print("共用",elapsed,"秒时间")

url="http://www.jd.com"
spider=MySpider()
while True:
    print("1.爬取")
    print("2.显示")
    print("3.退出")
    s=input("请选择(1,2,3):")
    if s=="1":
        spider.executeSpider(url,"手机")
    elif s=="2":
        spider.showDB()
    elif s=="3":
```

break

数据库截图：



对象 | testmodel_phone @test (tes...

开始事务 | 备注 | 筛选 | 排序 | 导入 | 导出

id	mNo	mMark	mPrice	mNote	mFile
1	000001	【预售】魅族	3198.00	【预售】魅族 16s 骁龙855:	000001.jpg
2	000002	Apple	5698.00	Apple iPhone XR (A2108)	000002.jpg
3	000003	【KPL官方比赛用机】vivo	3298.00	【KPL官方比赛用机】vivo i	000003.jpg
4	000004	华为	3988.00	华为 HUAWEI P30 超感光	000004.jpg
5	000005	荣耀8X	1299.00	荣耀8X 千元屏霸 91%屏占	000005.jpg
6	000006	小米	1199.00	小米 红米Redmi Note7 幻	000006.jpg
7	000007	荣耀10青春版	1299.00	荣耀10青春版 幻彩渐变 24	000007.jpg
8	000008	vivo	799.00	vivo U1 水滴全面屏 AI智慧	000008.jpg
9	000009	联想Z6	2999.00	联想Z6 Pro 8GB+128GB	000009.jpg
10	000010	小米	799.00	小米 红米6 4GB+64GB 铂	000010.jpg
11	000011	荣耀V20	2799.00	荣耀V20 胡歌同款 麒麟980	000011.jpg
12	000012	荣耀畅玩8C两天一充	899.00	荣耀畅玩8C两天一充 莱茵护	000012.jpg
13	000013	小米8SE	1399.00	小米8SE 全面屏智能游戏拍	000013.jpg
14	000014	小米9	3299.00	小米9 4800万超广角三摄 8	000014.jpg
15	000015	小米8青春版	1499.00	小米8青春版 镜面渐变AI双	000015.jpg
16	000016	vivo	1598.00	vivo Z3 6GB+64GB 极光蓝	000016.jpg
17	000017	三星	6999.00	三星 Galaxy S10+ 8GB+1	000017.jpg
18	000018	小米	799.00	小米 红米Redmi 7 AI双摄	000018.jpg
19	000019	Apple	6199.00	Apple iPhone X (A1865) 6	000019.jpg

SELECT * FROM `testmodel_phone` LIMIT 0, 1000

第 1 条记录 (共 1000 条) 于第 1 页

界面截图：