京东爬取笔记本搜索界面，django 显示

Jingdong 文件夹 scrapy 爬取文件

1. url 分析

根据分析这个界面主要是 page 的变换，每翻一页 page+2

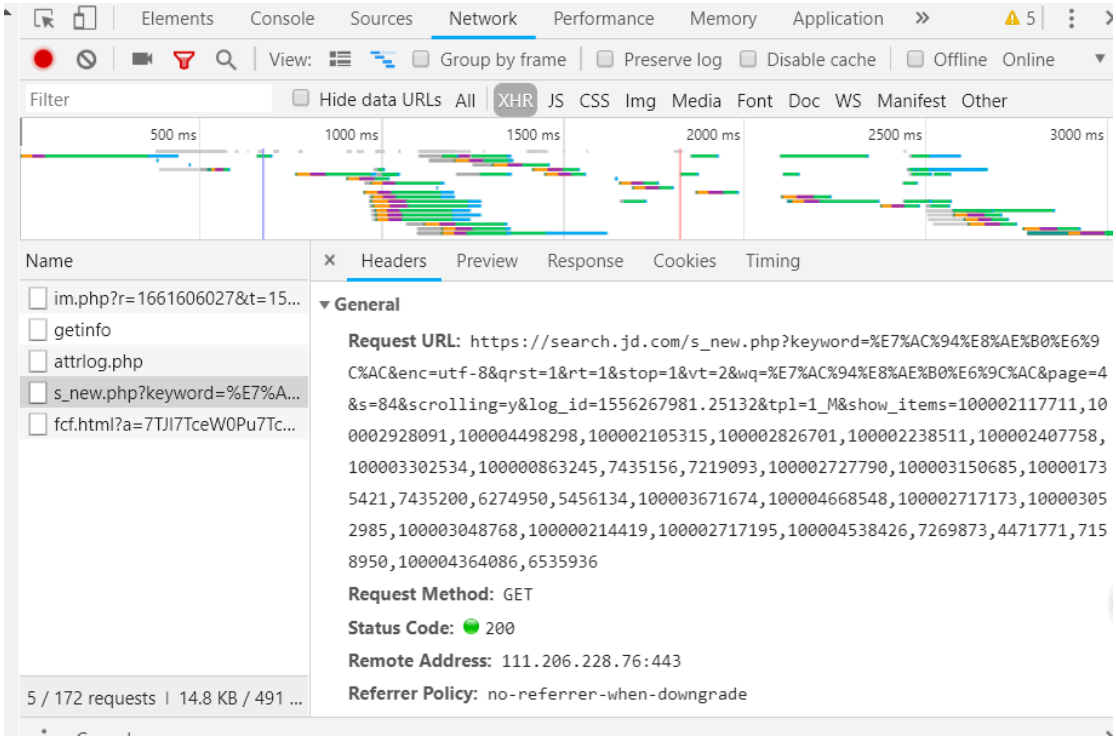
基本格式是：

```
https://search.jd.com/Search?keyword=笔记本
&enc=utf-8&qrst=1&rt=1&stop=1&vt=2&wq=笔记本&page=%d&click=0
```

2.

数据分析：

数据都在 li 标签里面，同样可以使用 xpath 解析，不过一个界面的 li 标签只有 30 个，由于他是动态网页，另外 30 个商品在隐藏的网址。



简化后的网址：

```
"https://search.jd.com/Search?keyword=笔记本
&enc=utf-8&qrst=1&rt=1&stop=1&vt=2&wq=笔记本
&page=%d&scrolling=y&show_items=%s"
```

Page 第一页是 2，隔页增 2，items 是前三十个商品的 id 的集合

代码与前三十个雷同

2. 爬取数据

用的 scrapy 库爬取

安装使用：https://www.runoob.com/w3cnote/scrapy-detail.html

代码解析

Items.py： 需要爬取的数据结构

```
class JingdongItem(scrapy.Item):
    # define the fields for your item here like:
    # name = scrapy.Field()
    goods_id = scrapy.Field()  # 商品 ID
    link = scrapy.Field()  # 商品链接
    goods_name = scrapy.Field()  # 商品名字
    shop_name = scrapy.Field()  # 店家名字
    price = scrapy.Field()  # 电脑价格
    comments = scrapy.Field()  # 评论数量
```

settings.py:用于 pipelines.py 里数据库连接

```
MYSQL_HOST = 'localhost'
MYSQL_DBNAME = 'jingdong'
MYSQL_USER = 'root'
MYSQL_PASSWD = '123456'
MYSQL_PORT = 3306
```

Pipelines.py:连数据库导入数据，防错处理

```
def from_settings(cls, settings):
        '''1、@classmethod 声明一个类方法，而对于平常我们见到的则叫做实例方法。
           2、类方法的第一个参数 cls（class 的缩写，指这个类本身），而实例方法的第一个参数是 self，表示该类的一个实例
           3、可以通过类来调用，就像 C.f()，相当于 java 中的静态方法'''
        dbparams = dict(
            host=settings['MYSQL_HOST'],
            db=settings['MYSQL_DBNAME'],
            user=settings['MYSQL_USER'],
            passwd=settings['MYSQL_PASSWD'],
            charset='utf8',  # 编码格式
            cursorclass=MySQLdb.cursors.DictCursor,
            use_unicode=False,
        )
        dbpool = adbapi.ConnectionPool('MySQLdb', **dbparams)  # **表示将字典扩展为关键字参数
        return cls(dbpool)  # 相当于 dbpool 付给了这个类

    def __init__(self, dbpool):
        self.dbpool = dbpool
def process_item(self, item, spider):
```

```
        query = self.dbpool.runInteraction(self._conditional_insert, item)
# 调用插入的方法
        query.addErrback(self._handle_error, item, goods_ider)  # 调用异常
处理方法
        return item

         # 写入数据库中
    def _conditional_insert(self, tx, item):
        sql = "insert into
jingdong(goods_id,goods_name,link,shop_name,price,comments)
values(%s,%s,%s,%s,%s,%s)"

        params = (
        item["goods_id"], item["goods_name"], item["link"],
item["shop_name"],item["price"], item["comments"])
        tx.execute(sql, params)

        # 错误处理方法
    def _handle_error(self, failue, item, goods_ider):
        print('--------------database operation
exception!!-----------------')
        print(failue)
```

jd.py:爬取数据，未能实现隐藏三十条爬取

起始 url

```
page=1
    url = "https://search.jd.com/Search?keyword=笔记本
&enc=utf-8&qrst=1&rt=1&stop=1&vt=2&wq=笔记本&page=%d&click=0"
```

start_requests **方法将 url 调用给** parsefirstPage

```
def start_requests(self):
        yield scrapy.Request(self.url % (self.page),
callback=self.parsefirstPage)
```

分析找到的 url 中的所有商品 link,给 parsegoods **使用，page+2 提供下一个 url**

```
def parsefirstPage(self, response):
        infos =
response.xpath('//li[@class="gl-item"]/div/div[@class="p-img"]/a')
        for info in infos:
            item = JingdongItem()
            url = info.xpath('@href').extract()
```

```
            goods_link = response.urljoin(url[0])
            item['link'] = goods_link  # 商品链接
            for link in url:
                url = response.urljoin(link)
                yield Request(url, meta={'meta': item},
callback=self.parsegoods)
            if self.page<200:
                self.page +=2 #翻页
                yield scrapy.Request(self.url % (self.page),
callback=self.parsefirstPage)
```

parsegoods 进入商品详情页分析其他字段

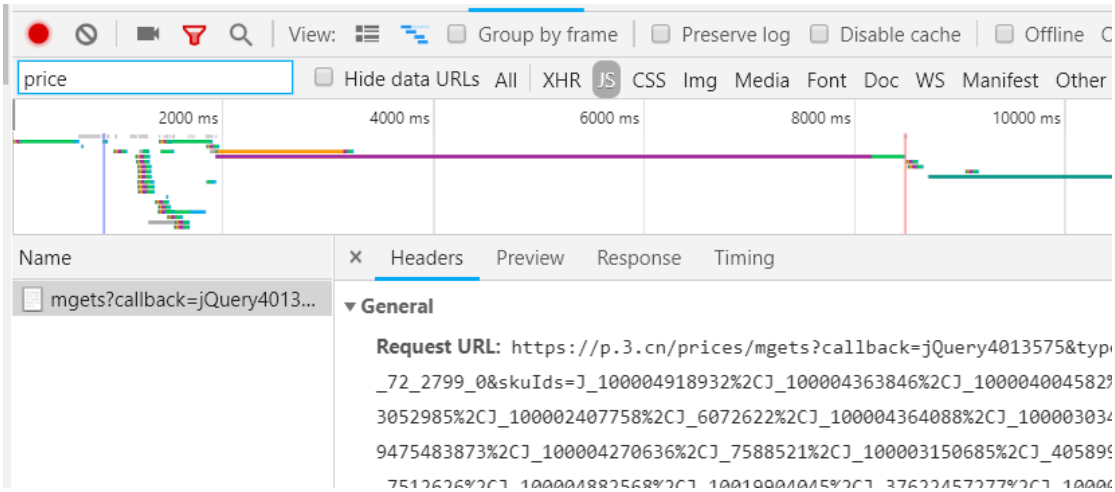
```
def parsegoods(self, response):
        item = response.meta['meta']
        id= response.xpath('//a[@class="compare J-compare
J_contrast"]/@data-sku').extract()[0] # 商品 id
        #ids = []
        #ids.append(''.join(id))
        item['goods_id'] = id
        item['goods_name'] =
response.xpath('//div[@class="sku-name"]/text()').extract()[0].strip() #
名称
        shop_name=
response.xpath('//div[@class="name"]/a/text()').extract()[0] # 商店名称

print("----------------------",shop_name,"----------------------")
        item['shop_name']=shop_name
```

分析价格 每个商品的价格的具体信息都在

```
https://p.3.cn/prices/mgets?callback=jQuery7726740&skuIds=
```

ids 后面商品 id

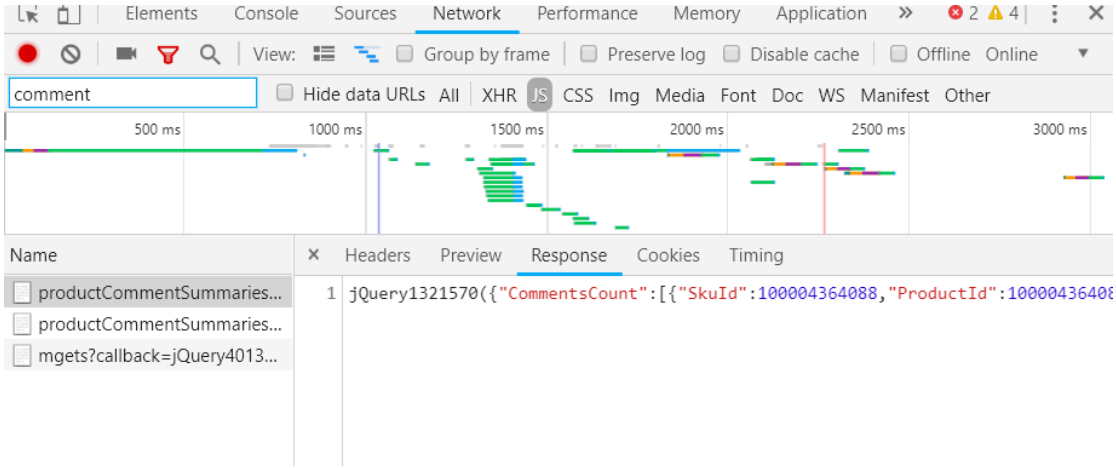


```
price_url = "https://p.3.cn/prices/mgets?callback=jQuery7726740&skuIds="
+ str(id)
        price = requests.get(price_url).text
        money = re.findall(r'\"p\"\:\"(.*?)\"}]\)', price)
        item['price'] = money[0]
```

评论也存在一个单独的链接

https://club.jd.com/comment/productCommentSummaries.action?referenceIds

ids 后面也是商品的 id

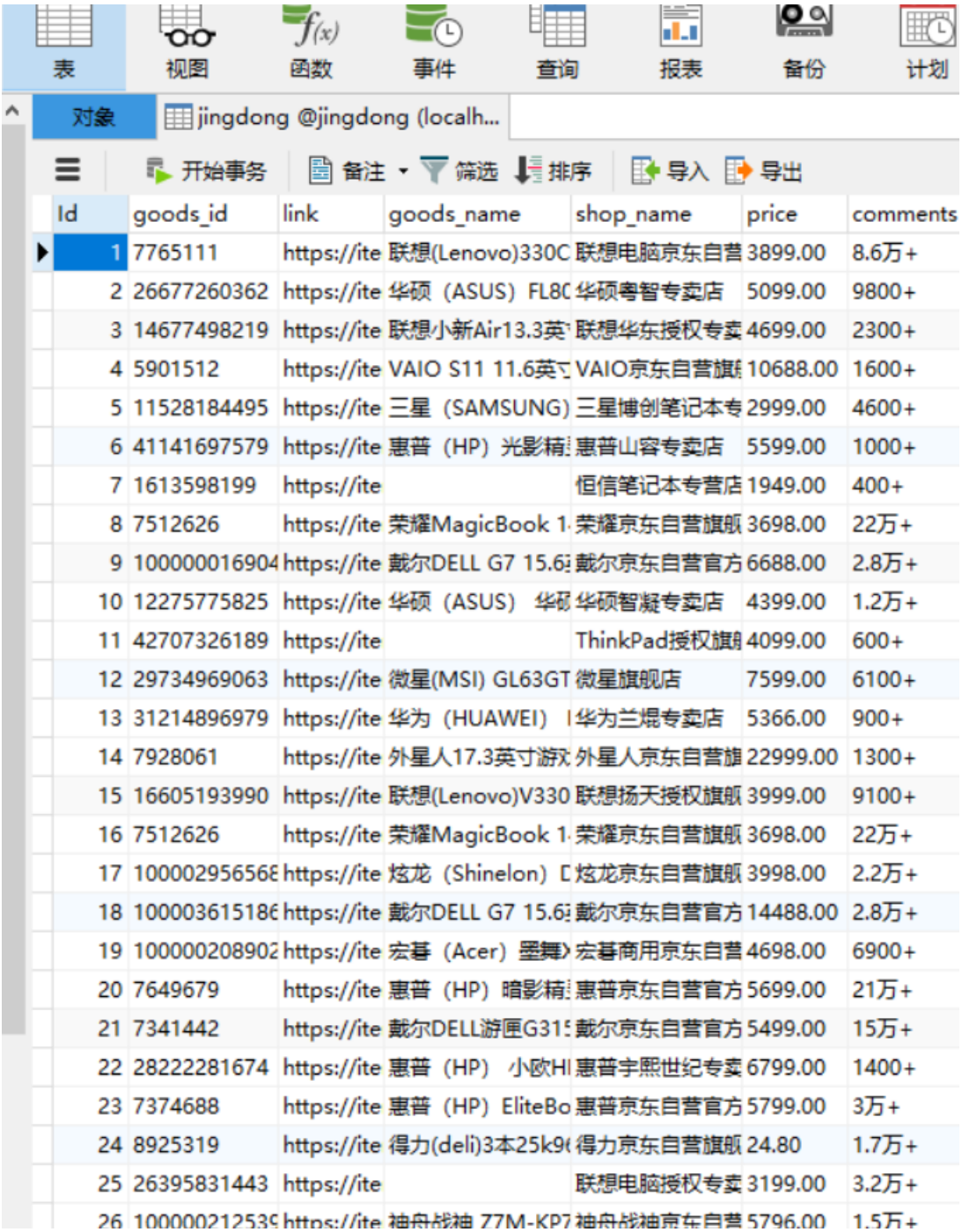



```
comments =
"https://club.jd.com/comment/productCommentSummaries.action?referenceId
s=" + str(id)
        yield scrapy.Request(comments, meta={'item': item},
callback=self.parse_getCommentnum)



    def parse_getCommentnum(self, response):
        item = response.meta['item']
        date = json.loads(response.text)  #解析 json
        item['comments']= date['CommentsCount'][0]['CommentCountStr']   #
评论数量
```

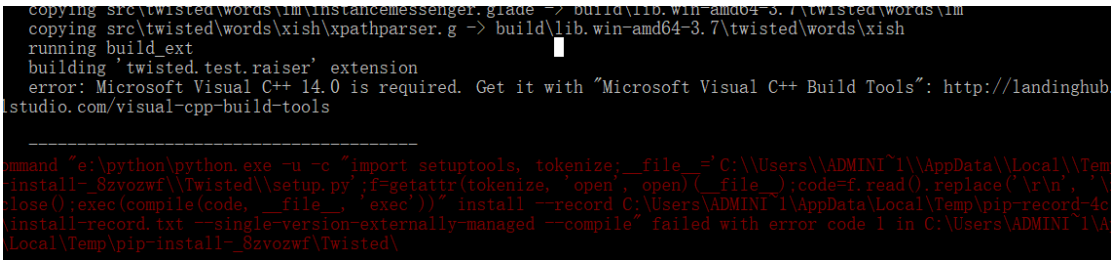
运行结果：



Id	goods_id	link	goods_name	shop_name	price	comments
1	7765111	https://ite	联想(Lenovo)330C	联想电脑京东自营	3899.00	8.6万+
2	26677260362	https://ite	华硕（ASUS）FL8(	华硕粤智专卖店	5099.00	9800+
3	14677498219	https://ite	联想小新Air13.3英	联想华东授权专卖	4699.00	2300+
4	5901512	https://ite	VAIO S11 11.6英寸	VAIO京东自营旗舰	10688.00	1600+
5	11528184495	https://ite	三星（SAMSUNG）	三星博创笔记本专	2999.00	4600+
6	41141697579	https://ite	惠普（HP）光影精	惠普山容专卖店	5599.00	1000+
7	1613598199	https://ite		恒信笔记本专营店	1949.00	400+
8	7512626	https://ite	荣耀MagicBook 1	荣耀京东自营旗舰	3698.00	22万+
9	100000016904	https://ite	戴尔DELL G7 15.6	戴尔京东自营官方	6688.00	2.8万+
10	12275775825	https://ite	华硕（ASUS） 华硕	华硕智凝专卖店	4399.00	1.2万+
11	42707326189	https://ite		ThinkPad授权旗舰	4099.00	600+
12	29734969063	https://ite	微星(MSI) GL63GT	微星旗舰店	7599.00	6100+
13	31214896979	https://ite	华为（HUAWEI）	华为兰焜专卖店	5366.00	900+
14	7928061	https://ite	外星人17.3英寸游戏	外星人京东自营旗	22999.00	1300+
15	16605193990	https://ite	联想(Lenovo)V330	联想扬天授权旗舰	3999.00	9100+
16	7512626	https://ite	荣耀MagicBook 1	荣耀京东自营旗舰	3698.00	22万+
17	100002956568	https://ite	炫龙（Shinelon）	炫龙京东自营旗舰	3998.00	2.2万+
18	100003615186	https://ite	戴尔DELL G7 15.6	戴尔京东自营官方	14488.00	2.8万+
19	100000208902	https://ite	宏碁（Acer）墨舞	宏碁商用京东自营	4698.00	6900+
20	7649679	https://ite	惠普（HP）暗影精	惠普京东自营官方	5699.00	21万+
21	7341442	https://ite	戴尔DELL游匣G315	戴尔京东自营官方	5499.00	15万+
22	28222281674	https://ite	惠普（HP） 小欧HI	惠普宇熙世纪专卖	6799.00	1400+
23	7374688	https://ite	惠普（HP）EliteBo	惠普京东自营官方	5799.00	3万+
24	8925319	https://ite	得力(deli)3本25k9(	得力京东自营旗舰	24.80	1.7万+
25	26395831443	https://ite		联想电脑授权专卖	3199.00	3.2万+
26	100000212539	https://ite	神舟战神 Z7M-KP7	神舟战神京东自营	5796.00	1.5万+

几个错误

安装时的错误



具体解决：https://blog.csdn.net/u013078422/article/details/79014745

工作目录运行 scrapy crawl jd 问题

```
    from twisted.internet.stdio import StandardIO, PipeAddress
  File "e:\python\lib\site-packages\twisted\internet\stdio.py", line
    from twisted.internet import _win32stdio
  File "e:\python\lib\site-packages\twisted\internet\_win32stdio.py",
    import win32api
ModuleNotFoundError: No module named 'win32api'
```

```
D:\code\jdlaptop\jdlaptop>pip install pywin32
Collecting pywin32
  Downloading https://files.pythonhosted.org/packages/a3/8a/eada1e7990202cd27e58eca2a278c344f
/pywin32-224-cp37-cp37m-win_amd64.whl (9.0MB)
    100% |                                        | 9.1MB 99kB/s
```

由于连接方在一段时间后没有正确答复或连接的主机没有反应，连接尝试失败。. 网络原因

Django 显示

与豆瓣爬取显示类似

序号	商品编号	链接	商品名	商店名	价格	评论数
1	7765111	https://item.jd.com/7765111.html	联想(Lenovo)330C 英特尔酷睿i5 15.6英寸商务影音笔记本电脑(i5-8250U 4G 1T+128G SSD 独显MX110)黑	联想电脑京东自营旗舰店	3899.00	8.6万+
2	26677260362	https://item.jd.com/26677260362.html	华硕（ASUS）FL8000顽石五代i7游戏商务办公学生手提15.6英寸笔记本电脑【升级版i7】荣耀金FL8000 套餐五 8G内存/1TB硬盘+240G固态	华硕粤智专卖店	5099.00	9800+
3	14677498219	https://item.jd.com/14677498219.html	联想小新Air13.3英寸超轻薄笔记本电脑酷睿8代四核 100%sRGB高色域 i5-8265U 8G 256G金色官方核显标配	联想华东授权专卖店	4699.00	2300+
4	5901512	https://item.jd.com/5901512.html	VAIO S11 11.6英寸 845克 轻薄商务笔记本电脑 (i5-8250U 8G 512G SSD FHD Win10 指纹识别 背光/静音键盘)深夜黑	VAIO京东自营旗舰店	10688.00	1600+
5	11528184495	https://item.jd.com/11528184495.html	三星（SAMSUNG） 350XAA 15.6英寸超轻薄便携学生游戏手提电脑商务办公笔记本电脑 轻薄本 【哑光黑定制】 8G/128G固态	三星博创笔记本专卖店	2999.00	4600+
6	41141697579	https://item.jd.com/41141697579.html	惠普（HP）光影精灵4代Pro暗影精灵游戏本笔记本畅游人电竞版15.6英寸手提吃鸡电脑【新品上市】冰封版 i5-	惠普山容专卖店	5599.00	1000+