

温州大学瓯江学院

爬虫期中项目实验报告

| | | | | | |
|-------|----------|------|-----------|------|-------------|
| 实验名称: | | | | | |
| 班 级: | 16 计算机三班 | 姓 名: | 黄银萍 | 学 号: | 16219111328 |
| 实验地点: | 7-403 | 日 期: | 2019.4.22 | | |

一、首页:

截图如下:



二、爬取豆瓣数据:

主要代码如下:

```
import urllib.request
from bs4 import BeautifulSoup
import pymysql
import threading
import os
import random
import bs4

ua_list=[
    "Mozilla/5.0 (Macintosh; Intel Mac OS X 10.6; rv2.0.1) Gecko/20100101 Firefox/4.0.1",
    "Mozilla/5.0 (Windows NT 6.1; rv2.0.1) Gecko/20100101 Firefox/4.0.1",
    "Opera/9.80 (Macintosh; Intel Mac OS x 10.6.8; U; en) Presto/2.8.131 Version/11.11",
    "Opera/9.80 (Windows NT 6.1; U; en) Presto/2.8.131 Version/11.11",
```

"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_7_0) AppleWebKit/535.11 (KHTML,like Gecko) Chrome/17.0.963.56 Safari/535.11"]

```
class MySpider:
```

```
    def openDB(self):
```

```
        self.con=pymysql.connect(host='localhost',user='root',passwd='1597',db='root',charset='utf8')
```

```
        self.cursor=self.con.cursor()
```

```
    def initDB(self):
```

```
        self.count=0
```

```
        self.TS=[]
```

```
    def closeDB(self):
```

```
        self.con.commit()
```

```
        self.con.close()
```

```
    def splitItems(self,p):
```

```
        res = []
```

```
        flag = True
```

```
        for c in p.children:
```

```
            if isinstance(c,bs4.element.NavigableString):
```

```
                t = c.string.replace("\n","").strip()
```

```
                if t!="":
```

```
                    if flag:
```

```
                        pos = t.find("主演")
```

```
                        director = t[:pos].replace("导演:", "")
```

```
                        actor = t[pos + 3:]
```

```
                        res.append(director.strip())
```

```
                        res.append(actor.strip())
```

```
                    else:
```

```
                        st = t.split("/")
```

```
                        for e in st:
```

```
                            res.append(e.strip())
```

```
                        break
```

```
            elif isinstance(c,bs4.element.Tag) and c.name == "br":
```

```
                flag = False
```

```
        return res
```

```
    def spider(self,url):
```

```
        try:
```

```
            print(url)
```

```

req = urllib.request.Request(url=url,headers={"User-Agent":random.choice(ua_list)})
resp =urllib.request.urlopen(req)
html =resp.read().decode()
soup = BeautifulSoup(html,"lxml")

lis
=soup.find("div",attrs={"id":"content"}).find("ol",attrs={"class":"grid_view"}).find_all("li")
for li in lis:
    div=li.find("div",attrs={"class":"info"})
    hd=div.find("div",attrs={"class":"hd"})
    spans=hd.find_all("span",attrs={"class":"title"})
    mTitle=spans[0].text.replace("\n","").strip() if len(spans)>0 else ""
    print(mTitle)
    mNative=spans[1].text.replace("\n","").strip() if len(spans)>1 else ""
    print(mNative)
    mNickname=hd.find("span",attrs={"class":"other"}).text.replace("\n","").strip()
    print(mNickname)
    sdiv=li.find("div",attrs={"class":"star"})
    mPoint=sdiv.find("span",attrs={"class":"rating_num"}).text.replace("\n","").strip()
    print(mPoint)
    mComment=sdiv.find_all("span")[-1].text.replace("\n","").strip()
    print(mComment)
    bd=div.find("div",attrs={"class":"bd"})
    p=bd.find("p")
    res=self.splitltems(p)
    mDirectors=res[0] if len(res)>0 else ""
    print(mDirectors)
    mActors=res[1] if len(res)>1 else ""
    print(mActors)
    mTime=res[2] if len(res)>2 else ""
    print(mTime)
    mCountry=res[3] if len(res)>3 else ""
    print(mCountry)
    mType=res[4] if len(res)>4 else ""
    print(mType)
    img=li.find("div",attrs={"class":"pic"}).find("img")
    src=urllib.request.urljoin(url,img["src"])
    self.count += 1
    T = threading.Thread()
    T.setDaemon(False)
    T.start()

```

```

        self.TS.append(T)
        self.cursor.execute("insert
testmodel_movie(mTitle,mNative,mNickname,mDirecors,mActors,
mTime,mCountry,mType,mPoint,mComment,mFile) values (%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s)",
(mTitle,mNative,mNickname,mDirectors,mActors,mTime,mCountry,mType,mPoint,mComment,src))
        div=soup.find("div",attrs={"class":"paginator"})
        link=div.find ("span",attrs={"class":"next"}).find("a")
        if link:
            href=link["href"]
            url = urllib.request.urljoin(url,href)
            self.spider(url)
        except Exception as err:
            print ("spider:"+str(err))

    def process(self):
        self.openDB()
        self.initDB()
        self.spider("https://movie.douban.com/top250")
        self.closeDB()
        for T in self.TS:
            T.join()

spider=MySpider()
while True:
    print("1.爬取")
    print("2.退出")
    s=input("选择 (1, 2) :")
    if s=="1":
        spider.process()
    elif s=="2":
        break

```


数据库截图:

| 对象 testmodel_movie @root (ro... | | | | | | | | | | | |
|---|---|------------------------|------------|-----------|---------|---------|------------|------------|------------|----------|-------|
| 开始事务 备注 筛选 排序 导入 导出 | | | | | | | | | | | |
| id | mTitle | mNative | mNickname | mDirecors | mActors | mTime | mCountry | mType | mPoint | mComment | mFile |
| 351 | 肖申克的救 / The Shawsh / 月黑高飞(港) / | 弗兰克·德拉邦特 | 蒂姆·罗宾斯 Ti | 1994 | 美国 | 犯罪 剧情 | 9.6 | 1401550人评价 | https://im | | |
| 352 | 霸王别姬 | / 再见，我的妻 / 陈凯歌 Kaige C | 张国荣 Leslie | 1993 | 中国大陆 香港 | 剧情 爱情 同 | 9.6 | 1038128人评价 | https://im | | |
| 353 | 这个杀手不: / Léon | / 杀手莱昂 / 终枪吕克·贝松 Luc B | 让·雷诺 Jean | 1994 | 法国 | 剧情 动作 犯 | 9.4 | 1279733人评价 | https://im | | |
| 354 | 阿甘正传 / Forrest Gun / 福雷斯特·冈普 | 罗伯特·泽米吉斯 汤姆·汉克斯 T | 1994 | 美国 | 剧情 爱情 | 9.4 | 1103796人评价 | https://im | | | |
| 355 | 美丽人生 / La vita è be / 一个快乐的传说(罗伯托·贝尼尼 R | 罗伯托·贝尼尼 | 1997 | 意大利 | 剧情 喜剧 爱 | 9.5 | 646317人评价 | https://im | | | |
| 356 | 泰坦尼克号 / Titanic / 铁达尼号(港 / 台 詹姆斯·卡梅隆 J | 詹姆斯·卡梅隆 | 1997 | 美国 | 剧情 爱情 灾 | 9.3 | 1044273人评价 | https://im | | | |
| 357 | 千与千寻 / 千と千尋の神隠し(台) / 宫崎骏 Hayao | 宫崎骏 Rumi | 2001 | 日本 | 剧情 动画 奇 | 9.3 | 1029533人评价 | https://im | | | |
| 358 | 辛德勒的名! / Schindler's / 舒特拉的名单(港 史蒂文·斯皮尔伯 | 史蒂文·斯皮尔伯 | 1993 | 美国 | 剧情 历史 战 | 9.5 | 576354人评价 | https://im | | | |
| 359 | 盗梦空间 / Inception / 潜行凶间(港) / 克里斯托弗·诺兰 | 克里斯托弗·诺兰 | 2010 | 美国 英国 | 剧情 科幻 悬 | 9.3 | 1109996人评价 | https://im | | | |
| 360 | 忠犬八公的 / Hachi: A Do / 忠犬小八(台) / 莱塞·霍尔斯道姆 | 理查·基尔 Rich | 2009 | 美国 英国 | 剧情 | 9.3 | 731581人评价 | https://im | | | |
| 361 | 机器人总动! / WALL-E / 瓦力(台) / 太空安德鲁·斯坦顿 A | 本·贝尔特 Ben | 2008 | 美国 | 爱情 科幻 动 | 9.3 | 735345人评价 | https://im | | | |
| 362 | 三傻大闹宝 / 3 Idiots / 三个傻瓜(台) / 拉库马·希拉尼 R | 阿米尔·汗 Aan | 2009 | 印度 | 剧情 喜剧 爱 | 9.2 | 996838人评价 | https://im | | | |
| 363 | 海上钢琴师 / La legend: / 声光伴我飞(港) 朱塞佩·托纳多雷 | 蒂姆·罗斯 Tim | 1998 | 意大利 | 剧情 音乐 | 9.2 | 818150人评价 | https://im | | | |
| 364 | 放牛班的春: / Les choriste / 歌声伴我心(港) 克里斯托夫·巴拉 | 热拉尔·朱尼奥 | 2004 | 法国 瑞士 德国 | 剧情 音乐 | 9.3 | 690160人评价 | https://im | | | |
| 365 | 楚门的世界 / The Trumar / 真人Show(港) / 彼得·威尔 Peter | 金·凯瑞 Jim C | 1998 | 美国 | 剧情 科幻 | 9.2 | 762482人评价 | https://im | | | |
| 366 | 大话西游之: / 西遊記大結局 / 西遊記完結篇(台) 刘镇伟 Jeffrey L | 周星驰 Steph | 1995 | 香港 中国大陆 | 喜剧 爱情 奇 | 9.2 | 771266人评价 | https://im | | | |
| 367 | 星际穿越 / Interstellar / 星际启示录(港) 克里斯托弗·诺兰 | 马修·麦康纳 M | 2014 | 美国 英国 加拿 | 剧情 科幻 冒 | 9.2 | 791518人评价 | https://im | | | |
| 368 | 龙猫 / とにのとり / 邻居托托罗 / 宫崎骏 Hayao | 宫崎骏 Nor | 1988 | 日本 | 动画 奇幻 冒 | 9.2 | 681728人评价 | https://im | | | |
| 369 | 教父 / The Godfat / Mario Puzo's T | 弗朗西斯·福特·科马 | 1972 | 美国 | 剧情 犯罪 | 9.3 | 500179人评价 | https://im | | | |
| 370 | 熔炉 / 도가니 / 无声呐喊(港) / 黄东赫 Dong-hy | 孔侑 Yoo Gor | 2011 | 韩国 | 剧情 | 9.3 | 446123人评价 | https://im | | | |
| 371 | 无间道 / 無間道 / Infernal Affairs 刘伟强 / 麦兆辉 刘德华 / 梁朝 | 梁朝 | 2002 | 香港 | 剧情 犯罪 悬 | 9.1 | 633194人评价 | https://im | | | |
| + - 开始事务 备注 筛选 排序 导入 导出 | | | | | | | | | | | |
| SELECT * FROM `testmodel_movie` LIMIT 0, 1000 | | | | | | | | | | | |
| 第 1 条记录 (共 250 条) 于第 1 页 | | | | | | | | | | | |

Django 网页显示:

豆瓣电影 × +

localhost:8000/moviedb




肖申克的救赎 / The Shawshank Redemption / 月黑高飞(港) / 刺激1995(台)

导演: 弗兰克·德拉邦特 Frank Darabont 主演: 蒂姆·罗宾斯 Tim Robbins /...

1994/美国/犯罪 剧情

9.6 1401550人评价

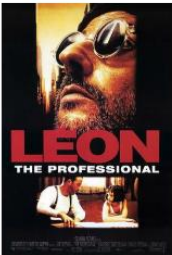


霸王别姬 / 再见，我的妻 / Farewell My Concubine

导演: 陈凯歌 Kaige Chen 主演: 张国荣 Leslie Cheung / 张丰毅 Fengyi Zha...

1993/中国大陆 香港/剧情 爱情 同性

9.6 1038128人评价




这个杀手不太冷 / Léon / 杀手莱昂 / 终极追杀令(台)

导演: 吕克·贝松 Luc Besson 主演: 让·雷诺 Jean Reno / 娜塔莉·波特曼 ...

1994/法国/剧情 动作 犯罪

9.4 1279733人评价



阿甘正传 / Forrest Gump / 福雷斯特·冈普

三、爬取京东商城数据：

主要代码如下：

```
from selenium import webdriver
from selenium.webdriver.firefox.options import Options
import urllib.request
import threading
import MySQLdb
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 = MySQLdb.connect(host='localhost',user='root',passwd='1597',db='root',charset="utf8")
            self.cursor = self.con.cursor()
            try:
                self.cursor.execute("drop table testmodel_phones")
            except:
                pass
            try:
                sql = "create table testmodel_phones (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 insertDB(self, mNo, mMark, mPrice, mNote, mFile):
    try:
        sql = "insert into testmodel_phones(mNo,mMark,mPrice,mNote,mFile) values (%s,%s,%s,%s,%s)"
        self.cursor.execute(sql, (mNo, mMark, mPrice, mNote, mFile))
    except Exception as err:
        print(err)
def showDB(self):
    try:
con=MySQLdb.connect(host='localhost',user='root',passwd='1597',db='root',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 testmodel_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(5)
        print(self.driver.current_url)
        self.driver.execute_script('window.scrollTo(0,7000)','1000')
        time.sleep(5)
        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=""

```



```

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=urlib.request.urljoin(self.driver.current_url,src1)
    p = src1.rfind(".")
    mFile = no + src1[p:]
elif src2:
    src2=urlib.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.insertDB(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("Spider starting.....")
    self.startUp(url, key)
    self.processSpider()
    self.closeUp()
    for t in self.threads:
        t.join()
    print("Spider completed.....")
    endtime = datetime.datetime.now()
    elapsed = (endtime - starttime).seconds
    print("Total ", elapsed, " seconds elapsed")

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:

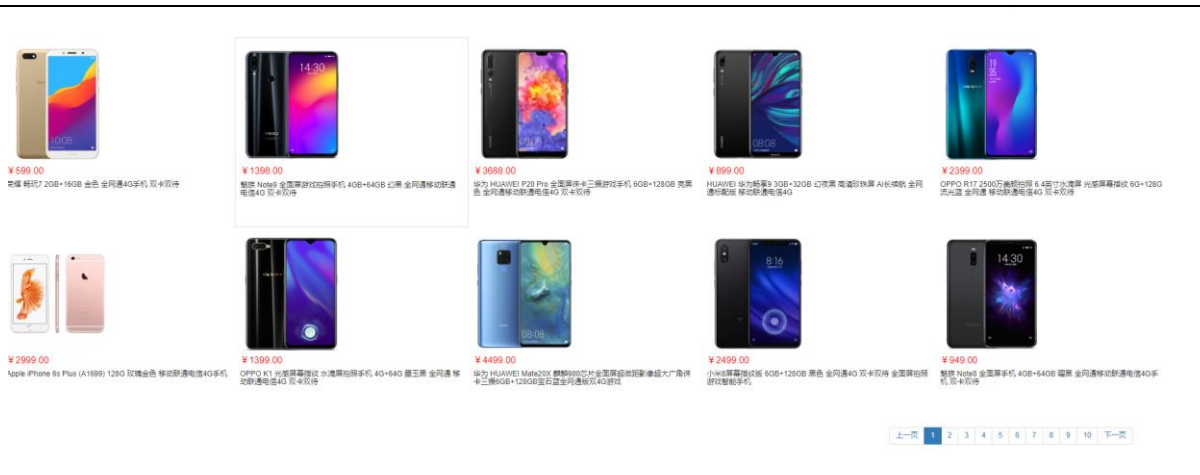
| 对象 testmodel_phones @root (r... | | | | | |
|---|--------|-----------------|---------|-------------------------|------------|
| <div> <div>三</div> <div>开始事务</div> <div>备注</div> <div>筛选</div> <div>排序</div> <div>导入</div> <div>导出</div> </div> | | | | | |
| id | mNo | mMark | mPrice | mNote | mFile |
| 1 | 000001 | 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 | 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 | 000007.jpg |
| 8 | 000008 | vivo | 799.00 | vivo U1 水滴全面屏 AI智 | 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 | 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 | 000012.jpg |
| 13 | 000013 | 小米8SE | 1399.00 | 小米8SE 全面屏智能游戏 | 000013.jpg |
| 14 | 000014 | 小米9 | 3299.00 | 小米9 4800万超广角三 | 000014.jpg |
| 15 | 000015 | 小米8青春版 | 1499.00 | 小米8青春版 镜面渐变 | 000015.jpg |
| 16 | 000016 | vivo | 1598.00 | vivo Z3 6GB+64GB 极光 | 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 | 000018.jpg |
| 19 | 000019 | Apple | 6199.00 | Apple iPhone X (A1865) | 000019.jpg |
| 20 | 000020 | vivo | 3598.00 | vivo X27 8GB+256GB大 | 000020.jpg |
| 21 | 000021 | 小米 | 649.00 | 小米 红米6A AI美颜 | 000021.jpg |

SELECT * FROM `testmodel_phones` LIMIT 0, 1000

第 1 条记录 (共 1000 条) 于

django 页面显示截图：

| | | | | |
|--|--|--|---|---|
|  <p>¥ 3198.00</p> <p>【预售】魅族 16s 骁龙855全面屏拍照游戏手机, 5GB+128GB 铂犀黑 全网通移动联通电信4G 双卡双待</p> |  <p>¥ 5698.00</p> <p>Apple iPhone XR (A2108) 128GB 黑色 移动联通电信4G手机, 双卡双待</p> |  <p>¥ 3298.00</p> <p>【KPL官方比赛用机】vivo iQOO 4400超快充电 8GB+128GB电竞全面屏拍照手机, 骁龙855电竞游戏, 全网通4G</p> |  <p>¥ 3988.00</p> <p>华为 HUAWEI P30 超感光徕卡三摄麒麟980AI智能芯片全网通移动联通电信4G 双卡双待</p> |  <p>¥ 1299.00</p> <p>荣耀8X 千元屏霸 91%屏占比, 2000万AI双摄 4GB+64GB 幻影黑 移动联通电信4G全网通 双卡双待</p> |
|  <p>¥ 1199.00</p> <p>小米 红米Redmi Note7 幻彩渐变AI双摄 4GB+64GB 梦幻蓝 全网通4G 双卡双待 全网通全网通全网通</p> |  <p>¥ 1299.00</p> <p>荣耀10青春版 幻彩渐变 AI双摄 2400万AI自拍 全网通4GB+64GB 梦幻蓝 移动联通电信4G全网通 双卡双待</p> |  <p>¥ 799.00</p> <p>vivo U1 水滴全面屏 AI智慧拍照手机, 3GB+32GB 极光色 移动联通电信4G</p> |  <p>¥ 2999.00</p> <p>联想Z6 Pro 8GB+128GB 黑色 骁龙855 4800万AI双摄 4000mAh大电池 P-C曲面水滴屏 游戏, 全网通4G 双卡双待</p> |  <p>¥ 799.00</p> <p>小米 红米6 4GB+64GB 铂犀黑 全网通4G手机, 双卡双待</p> |
|  <p>¥ 2799.00</p> <p>荣耀V20 胡歌同款 麒麟980芯片 智能全视屏 4800万双摄 6GB+128GB 幻影黑 全网通 移动联通电信4G全网通</p> |  <p>¥ 899.00</p> <p>荣耀畅玩8C两天一充 莱茵护眼 全网通4GB+32GB 幻影黑 移动联通电信4G全网通 双卡双待</p> |  <p>¥ 1399.00</p> <p>小米8青春版 全面屏智能游戏拍照手机, 6GB+64GB 灰色 骁龙710处理器 全网通4G 双卡双待</p> |  <p>¥ 3299.00</p> <p>小米9 4800万超广角三摄 8GB+128GB全网通4G双卡双待 全网通4G 双卡双待 全网通全网通全网通</p> |  <p>¥ 1499.00</p> <p>小米8青春版 镜面渐变AI双摄 6GB+64GB 梦幻蓝 全网通4G 双卡双待 全网通全网通全网通</p> |



四、爬取天气预报数据：

主要代码如下：

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

```
conn=pymysql.connect(host='localhost',user='root',passwd='1597',db='root',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()
```

```
dammit=UnicodeDammit(data,["utf-8","gbk"])
```

```
data=dammit.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
```

```
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_cityweather(city,date,weather,temp)
```

```

values(%s,%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:

| id | city | date | weather | temp |
|----|------|----------|----------|-----------|
| 1 | 深圳 | 24日 (今天) | 多云 | 25°C |
| 2 | 深圳 | 25日 (明天) | 多云转暴雨 | 31°C/25°C |
| 3 | 深圳 | 26日 (后天) | 暴雨转大雨 | 30°C/23°C |
| 4 | 深圳 | 27日 (周六) | 大雨转雷阵雨 | 27°C/23°C |
| 5 | 深圳 | 28日 (周日) | 雷阵雨转阵雨 | 28°C/24°C |
| 6 | 深圳 | 29日 (周一) | 阵雨 | 29°C/25°C |
| 7 | 深圳 | 30日 (周二) | 阵雨转大雨 | 30°C/24°C |
| 8 | 广州 | 24日 (今天) | 多云 | 24°C |
| 9 | 广州 | 25日 (明天) | 雷阵雨 | 29°C/22°C |
| 10 | 广州 | 26日 (后天) | 大雨转大到暴雨 | 26°C/23°C |
| 11 | 广州 | 27日 (周六) | 大到暴雨转雷阵雨 | 26°C/23°C |
| 12 | 广州 | 28日 (周日) | 雷阵雨 | 28°C/24°C |
| 13 | 广州 | 29日 (周一) | 雷阵雨 | 30°C/24°C |
| 14 | 广州 | 30日 (周二) | 雷阵雨转大雨 | 30°C/24°C |
| 15 | 上海 | 24日 (今天) | 阴 | 18°C |
| 16 | 上海 | 25日 (明天) | 小雨转多云 | 25°C/14°C |
| 17 | 上海 | 26日 (后天) | 多云 | 19°C/13°C |
| 18 | 上海 | 27日 (周六) | 晴转多云 | 21°C/16°C |
| 19 | 上海 | 28日 (周日) | 多云 | 24°C/18°C |
| 20 | 上海 | 29日 (周一) | 大雨转中雨 | 26°C/17°C |
| 21 | 上海 | 30日 (周二) | 小雨 | 21°C/18°C |

SELECT * FROM `testmodel_cityweather` LIMIT 0, 1000

第 1 条记录 (共 28 条) 于第 1 页

Django 界面截图:

| 城市 | 日期 | 天气 | 温度 |
|----|----------|----------|-----------|
| 深圳 | 24日 (今天) | 多云 | 25°C |
| 深圳 | 25日 (明天) | 多云转暴雨 | 31°C/25°C |
| 深圳 | 26日 (后天) | 暴雨转大雨 | 30°C/23°C |
| 深圳 | 27日 (周六) | 大雨转雷阵雨 | 27°C/23°C |
| 深圳 | 28日 (周日) | 雷阵雨转阵雨 | 28°C/24°C |
| 深圳 | 29日 (周一) | 阵雨 | 29°C/25°C |
| 深圳 | 30日 (周二) | 阵雨转大雨 | 30°C/24°C |
| 广州 | 24日 (今天) | 多云 | 24°C |
| 广州 | 25日 (明天) | 雷阵雨 | 29°C/22°C |
| 广州 | 26日 (后天) | 大雨转大到暴雨 | 26°C/23°C |
| 广州 | 27日 (周六) | 大到暴雨转雷阵雨 | 26°C/23°C |
| 广州 | 28日 (周日) | 雷阵雨 | 28°C/24°C |
| 广州 | 29日 (周一) | 雷阵雨 | 30°C/24°C |
| 广州 | 30日 (周二) | 雷阵雨转大雨 | 30°C/24°C |
| 上海 | 24日 (今天) | 阴 | 18°C |
| 上海 | 25日 (明天) | 小雨转多云 | 25°C/14°C |
| 上海 | 26日 (后天) | 多云 | 19°C/13°C |
| 上海 | 27日 (周六) | 晴转多云 | 21°C/16°C |
| 上海 | 28日 (周日) | 多云 | 24°C/18°C |
| 上海 | 29日 (周一) | 大雨转中雨 | 26°C/17°C |
| 上海 | 30日 (周二) | 小雨 | 21°C/18°C |
| 北京 | 24日 (今天) | 小雨 | 6°C |
| 北京 | 25日 (明天) | 多云 | 18°C/6°C |
| 北京 | 26日 (后天) | 晴转多云 | 21°C/11°C |
| 北京 | 27日 (周六) | 小雨 | 18°C/8°C |
| 北京 | 28日 (周日) | 多云 | 22°C/10°C |
| 北京 | 29日 (周一) | 多云转小雨 | 25°C/13°C |
| 北京 | 30日 (周二) | 多云 | 26°C/14°C |

五、淘宝商品信息定向爬虫：

主要代码如下：

```
import re
import requests
import pymysql

conn=pymysql.connect(host='localhost',user='root',passwd='1597',db='root',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_taobao(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_taobao @root (r... | | |
|---------------------------------|--------|---|
| 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背减负 |
| 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年级护背减负儿童书包男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 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 |
| 20 | 99.00 | 米熙休闲运动背包双肩包女书包中学生男韩版时尚大容量旅游旅行包 |
| 21 | 195.02 | 国家地理背包女运动户外时尚双肩包男牛津布旅行防水学生情侣书包 |

SELECT * FROM `testmodel_taobao` LIMIT 0, 1000

第 1 条记录 (共 92 条) 于第 1 页

django 页面显示截图：

淘宝书包

× +

← → 不安全 | 10.218.122.101:8000/taobaodb

淘宝书包信息

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：129.00