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)
```

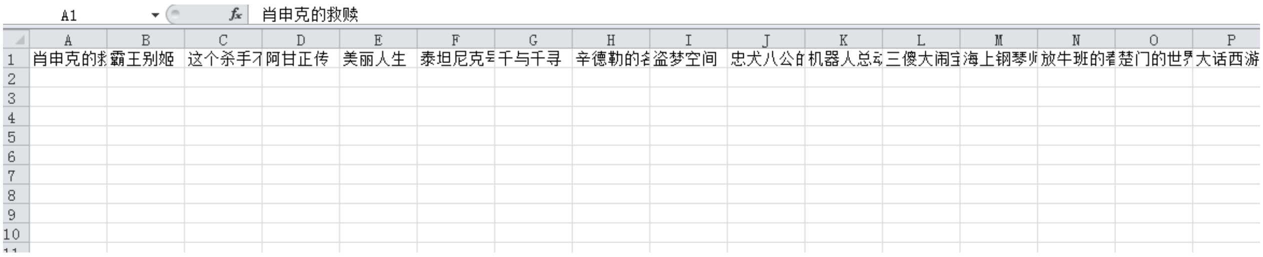
文件 One.py

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)
```



A1 肖申克的救赎

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	肖申克的救	霸王别姬	这个杀手不	阿甘正传	美丽人生	泰坦尼克号	千与千寻	辛德勒的名	盗梦空间	忠犬八公的	机器人总动	三傻大闹宝	海上钢琴师	放牛班的春	楚门的世界	大话西游
2																
3																
4																
5																
6																
7																
8																
9																
10																

文件 Two.py

将 csv 转存进数据库

代码：

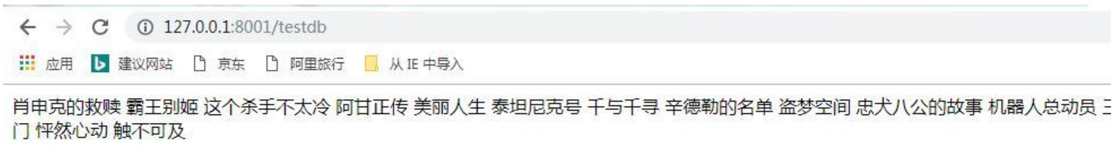
```
import csv
import pymysql

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



肖申克的救赎 霸王别姬 这个杀手不太冷 阿甘正传 美丽人生 泰坦尼克号 千与千寻 辛德勒的名单 盗梦空间 忠犬八公的故事 机器人总动员 三
门 怦然心动 触不可及

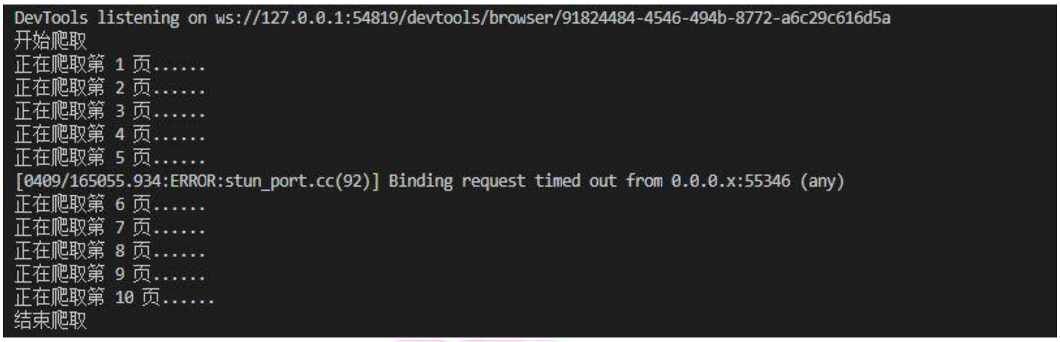
京东.py

爬取京东手机商品页面内容保存记事本（Jd.txt）

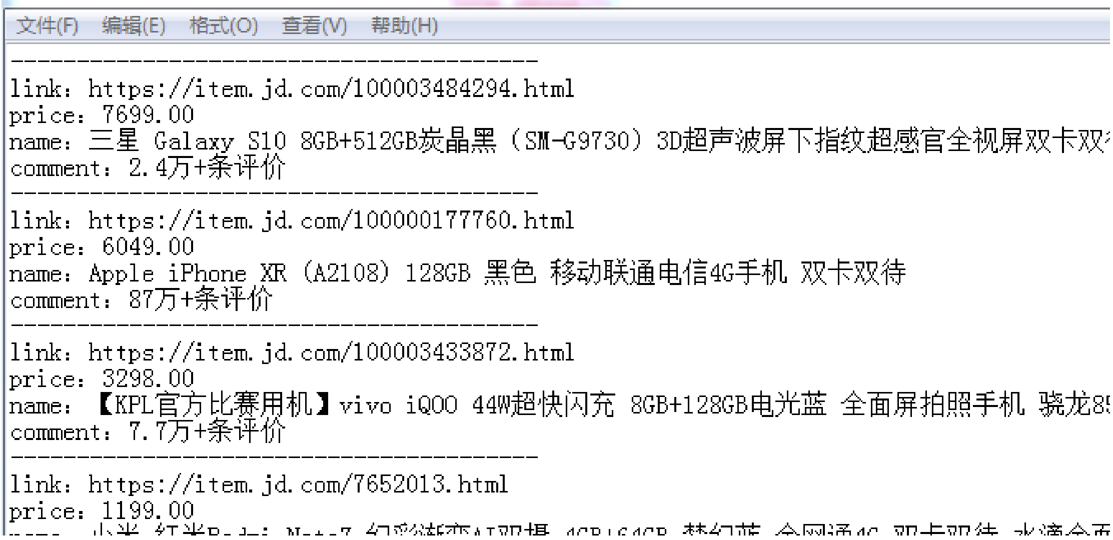
代码：

```
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
```



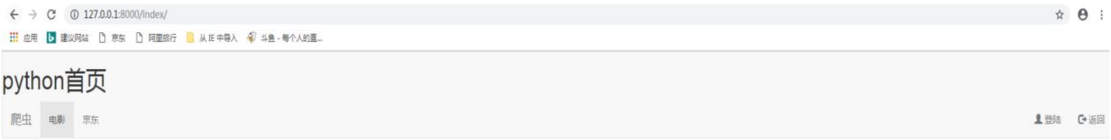
```
----------------------------------------
link: https://item.jd.com/100003484294.html
price: 7699.00
name: 三星 Galaxy S10 8GB+512GB炭晶黑 (SM-G9730) 3D超声波屏下指纹超感官全视屏双卡双
comment: 2.4万+条评价
----------------------------------------
link: https://item.jd.com/100000177760.html
price: 6049.00
name: Apple iPhone 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
name: 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 水滴全面
```

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>
```



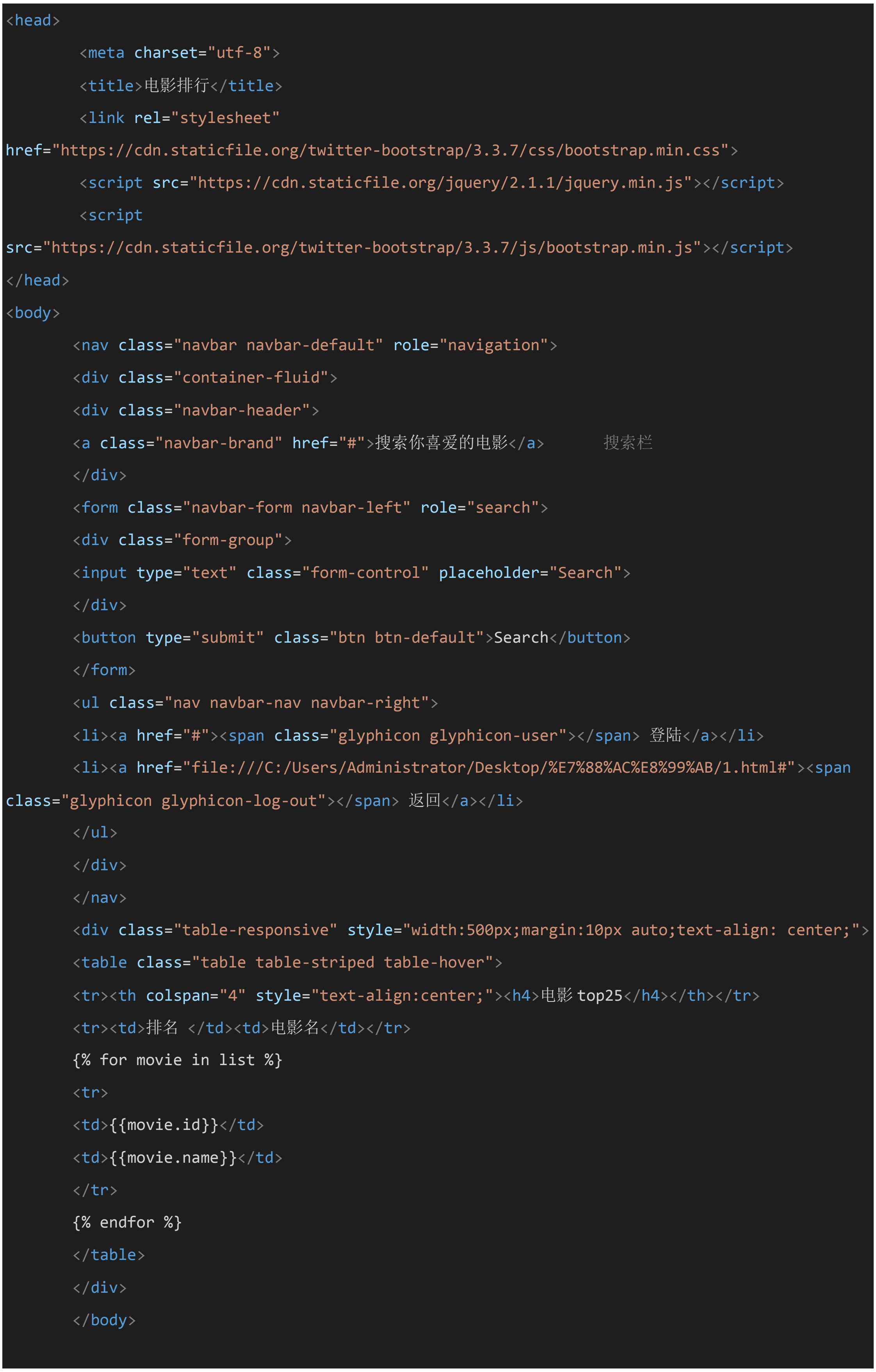
python首页

爬虫 电影 京东 登陆 返回

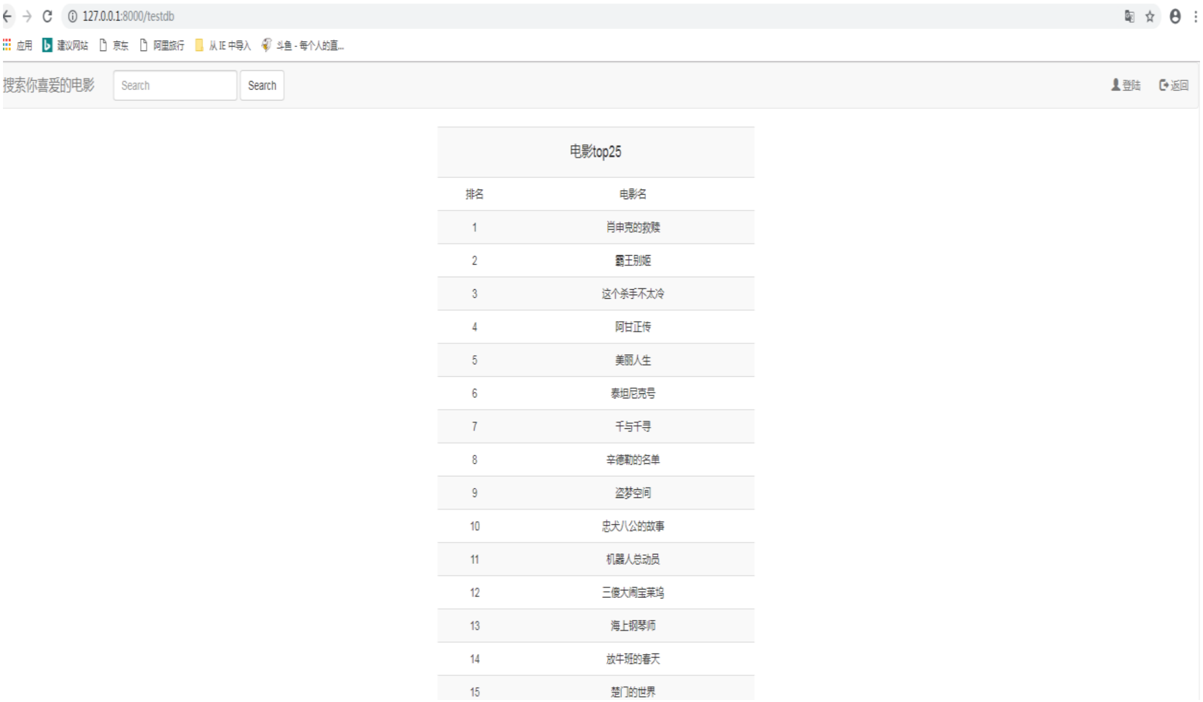
电影 top.html

代码：

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



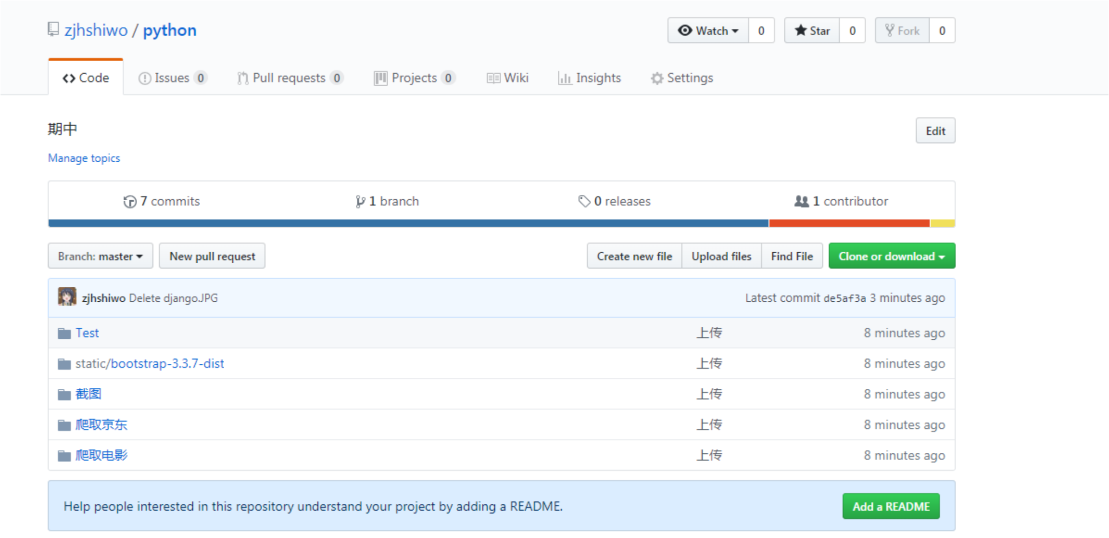
```
<head>
        <meta charset="utf-8">
        <title>电影排行</title>
        <link rel="stylesheet"
href="https://cdn.staticfile.org/twitter-bootstrap/3.3.7/css/bootstrap.min.css">
        <script src="https://cdn.staticfile.org/jquery/2.1.1/jquery.min.js"></script>
        <script
src="https://cdn.staticfile.org/twitter-bootstrap/3.3.7/js/bootstrap.min.js"></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>
```


127.0.0.1:8000/testdb
应用 建议网站 京东 阿里旅行 从 IE 中导入 斗鱼 - 每个人的直...
搜索你喜爱的电影
Search
Search
登陆
返回

电影top25

排名	电影名
1	肖申克的救赎
2	霸王别姬
3	这个杀手不太冷
4	阿甘正传
5	美丽人生
6	泰坦尼克号
7	千与千寻
8	辛德勒的名单
9	盗梦空间
10	忠犬八公的故事
11	机器人总动员
12	三傻大闹宝莱坞
13	海上钢琴师
14	放牛班的春天
15	楚门的世界

将文件代码上传到 github



期中

7 commits 1 branch 0 releases 1 contributor

zjhshiwo Delete django.JPG Latest commit de5af3a 3 minutes ago

Test	上传	8 minutes ago
static/bootstrap-3.3.7-dist	上传	8 minutes ago
截图	上传	8 minutes ago
爬取京东	上传	8 minutes ago
爬取电影	上传	8 minutes ago

Help people interested in this repository understand your project by adding a README.

https://github.com/zjhshiwo/python