



温州大学瓯江学院

WENZHOU UNIVERSITY OUJIANG COLLEGE

Python

题 目： Python+django

分 院： 数信分院

班 级： 16 计算机科学技术与应用 3

姓 名： 李钻

学 号： 16219111329

完成日期： 2019 年 06 月 14 日

温州大学瓯江学院教务部

二〇一二年十一月制

先用 MyMovieDB 和 MyPhoneDB 爬去静态的豆瓣 250 电影名称和京东手机的信息及图片放进 Mysql 数据库中，在通过 django 和 bootstrap 框架展示在网页中

Github:<https://github.com/user-lz/Demo>

12306 的代码:

```
import requests
from selenium import webdriver
import time
from selenium.webdriver.common.keys import Keys
from PIL import Image
import os
from io import BytesIO
import urllib.request
import re
from selenium.webdriver.common.action_chains import ActionChains

driver=webdriver.Firefox()
driver.maximize_window()
```

```
def getImgCode():
headers = {"User-Agent": "Mozilla/5.0 (Windows; U; Windows NT 6.0 x64; en-US; rv:1.9pre)Gecko/2008072421 Minefield/3.0.2pre"}
imgName="imgCode.jpg"
src1 = driver.find_element_by_class_name("imgCode").get_attribute("src")
req = urllib.request.Request(src1, headers=headers)
resp = urllib.request.urlopen(req, timeout=100)
data = resp.read()
if os.path.isfile(imgName):
os.remove(imgName)
fobj = open(imgName, "wb")
fobj.write(data)
fobj.close()
```

```
def parse_img():
url="http://littlebigluo.qicp.net:47720/"
response=requests.request("POST",url,data={"type":"1"},files={'pic_xxfile':open('imgCode.jpg','rb')})
result=[]
for i in re.findall("<B>(.*?)</B>",response.text)[0].split(" "):
result.append(int(i)-1)
return result
```

```
def move():
```

```

result = parse_img()
try:
    Action=ActionChains(driver)
    imgCode=driver.find_element_by_class_name("imgCode")
    coordinate=[[-105,-20],[-35,-20],[40,-20],[110,-20],[-105,50],[-35,50],[40,50],[110,50]]
    for i in result:
        Action.move_to_element(imgCode).move_by_offset(coordinate[i][0],coordinate[i][1]).click()
    Action.perform()
except Exception as e:
    print(e)

```

```

def login(userName,password):
    try:
        driver.get('https://kyfw.12306.cn/otn/resources/login.html')
        time.sleep(3)
        account=driver.find_element_by_class_name("login-hd-account")
        account.click()
        username=driver.find_element_by_id('J-userName')
        username.send_keys(userName)
        driver.find_element_by_id("J-password").send_keys(password,Keys.ENTER)
        time.sleep(3)
        getImgCode()
        move()
        driver.find_element_by_class_name('login-btn').click()
        time.sleep(3)
    except:
        print("error")

```

```

if __name__ == "__main__":
    login("13958713211","lizuan7456")

```

扫码登录

账号登录

用户名 / 邮箱 / 手机号

密码

验证成功, 跳转中...

刷新



恭喜! 完成验证。

立即登录

[注册12306账号](#) | [忘记密码?](#)

1、12306.cn网站每日06:00~23:00提供服务。

2、在12306.cn网站购票、改签和退票须不晚于开车前30分钟；办理“变更到站”业务时，请不晚于开车前48小时。

Python--My_Web

MyMovieDB.py

```
import requests
from lxml import etree
import MySQLdb

conn=MySQLdb.connect(host='localhost',user='root',passwd='lizuan..7456',db='test',charset="utf8")
cur=conn.cursor()

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)
    title=tree.xpath('//span[@class="title"][1]/text()')

    return title

result=[]

def get_all_page(start,end):
    result=[]
    for i in range(start,end-start):
        title_list=get_page(i*25)
        result+=title_list

    return result
```

```

if __name__=="__main__":
result=get_all_page(0,10)
for i in range(0,250):
cur.execute("insert into testmodel_test(movie_id,movie_name) values(%,%)", (i+1,result[i]))
cur.close()
conn.commit()
conn.close()

```

MyPhoneDB.py

```

from selenium.webdriver.firefox.options import Options
from selenium import webdriver
import os
from selenium.webdriver.common.keys import Keys
import time
import urllib.request
import threading
import datetime

from django.http import HttpResponse
from TestModel.models import Phones
from django.shortcuts import render
from django.core.paginator import Paginator

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 = "static\\MyPhonePicture"
firefox_option=Options()
firefox_option.add_argument('-headless')
driver=webdriver.Firefox(options=firefox_option)

```

```

def startUp(url,key):
try:
if not os.path.exists(imagePath):
os.mkdir(imagePath)
images = os.listdir(imagePath)
for img in images:
s = os.path.join(imagePath, img)
os.remove(s)
except Exception as err:
print(err)
driver.get(url)
keyInput=driver.find_element_by_id("key")
keyInput.send_keys(key)
keyInput.send_keys(Keys.ENTER)
def closeUp():

```

```
try:
driver.close()
except Exception as err:
print(err)
```

```
def download(src1,src2,mFile):
data=None
if src1:
try:
req = urllib.request.Request(src1, headers=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=headers)
resp = urllib.request.urlopen(req, timeout=400)
data = resp.read()
except:
pass
if data:
fobj = open(imagePath + "\\\" + mFile, "wb")
fobj.write(data)
fobj.close()
```

```
threads=[]
No=0
pNo=[]
pBrand=[]
pPrice=[]
pContent=[]
pFile=[]
```

```
def processSpider():
try:
time.sleep(3)
print(driver.current_url)
driver.execute_script("window.scrollTo(0,7000)","1000")
time.sleep(5)
lis = 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"
pPrice.append(price)

```

```

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

```

```

global No
No+=1
no = str(No)
while len(no) < 6:
no = "0" + no
pNo.append(no)

```

```

if src1:
src1=urllib.request.urljoin(driver.current_url,src1)
p = src1.rfind(".")
mFile = no + src1[p:]
elif src2:
src2=urllib.request.urljoin(driver.current_url,src2)
p = src2.rfind(".")
mFile = no + src2[p:]
if src1 or src2:
T=threading.Thread(target=download,args=(src1,src2,mFile))
T.setDaemon(False)

```

```

T.start()
threads.append(T)
else:
mFile = ""
pFile.append(mFile)
print(no,brand,price,mFile)
try:
driver.find_element_by_xpath("//span[@class='p-num']/a[@class='pn-prev dosabled']")
except:
nextPage=driver.find_element_by_xpath("//span[@class='p-num']/a[@class='pn-next']")
nextPage.click()
processSpider()
except Exception as err:
print(err)
return pNo,pBrand,pContent,pPrice,pFile

```

```

def insertDB(request):
starttime = datetime.datetime.now()
url="http://www.jd.com"
print("Spider starting.....")
startUp(url,"手机")
pNo,pBrand,pContent,pPrice,pFile = processSpider()
closeUp()
print("Spider completed.....")
endtime = datetime.datetime.now()
elapsed = (endtime - starttime).seconds
print("Total ", elapsed, " seconds elapsed")
for i in range(0,len(pNo)):
phoneDB=Phones(pNo=pNo[i],pBrand=pBrand[i],pPrice=pPrice[i],pContent=pContent[i],pFile=pFile[i]
)
phoneDB.save()
for t in threads:
t.join()
print("Spider completed.....")

```

```

return HttpResponse("<p>数据添加成功! </p>")

```

```

def get_pages(totalpage=1,current_page=1):
WEB_DISPLAY_PAGE = 5
front_offset = int(WEB_DISPLAY_PAGE / 2)
if WEB_DISPLAY_PAGE % 2 == 1:
behind_offset=front_offset
else:
behind_offset=front_offset -1

```



```

if totalpage < WEB_DISPLAY_PAGE:
return list(range(1,totalpage+1))
elif current_page<=front_offset:
return list(range(1,WEB_DISPLAY_PAGE+1))
elif current_page>=totalpage-behind_offset:
start_page=totalpage-WEB_DISPLAY_PAGE+1
return list(range(start_page,totalpage+1))
else:
start_page=current_page-front_offset
end_page=current_page+behind_offset

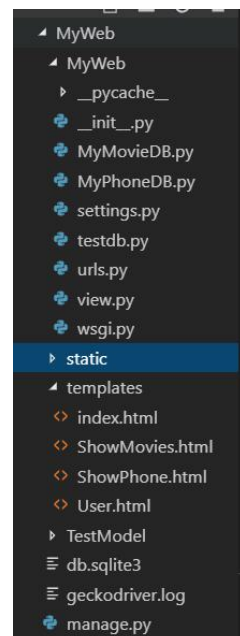
```

```

return list(range(start_page,end_page+1))

```

MyWeb，项目结构：



模型设计 models.py

```

# models.py

from django.db import models

class Test(models.Model):
    movie_id= models.CharField(max_length=10)
    movie_name= models.CharField(max_length=20)

class Phones(models.Model):
    pNo = models.CharField(max_length=7)
    pBrand = models.CharField(max_length=100)
    pPrice = models.CharField(max_length=20)
    pContent = models.CharField(max_length=100)
    pFile = models.CharField(max_length=20)

```

在 settings 中设置连接数据库

```
DATABASES = {
    'default': {

        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'test',
        'USER': 'root',
        'PASSWORD': 'lizuan..7456',
        'HOST': 'localhost',
        'PORT': '3306',}
}
```

Testadb.py:

```
from TestModel.models import Test,Phones
from django.http import HttpResponse
from django.shortcuts import render
import chardet

def showPhone(request):
    phone_list=Phones.objects.all()
    return render(request, 'ShowPhone.html', {'phone_list': phone_list})
```

```
def showDB(request):
    movie_list = Test.objects.all()
```

```
return render(request, 'ShowMovies.html', {'movie_list': movie_list})
```

View.py:

```
from django.shortcuts import render
from TestModel.models import Test

def User_Information(request):
    context = {}
    context['sname'] = '李钻'
    context['sno'] = '16219111329'
    context['sclass'] = '16 计算机科学与技术三'
    return render(request, 'User.html', context)

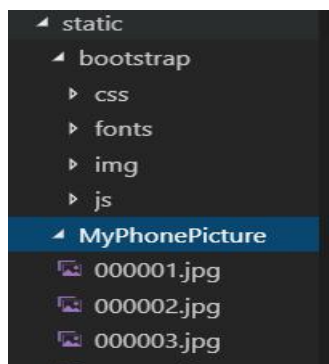
def index(request):
    return render(request, 'index.html', locals())
```

Urls.py:

```
from django.contrib import admin
from django.conf.urls import *
from . import testdb,view,MyPhoneDB
```

```
urlpatterns = [
url(r'^ShowMovies/',testdb.showDB),
url(r'^InsertPhone/',MyPhoneDB.insertDB),
url(r'^ShowPhone/',testdb.showPhone),
url(r'^index/',view.index),
url(r'^User/',view.User_Information),
]
```

Bootstrap 放在了 static 目录下

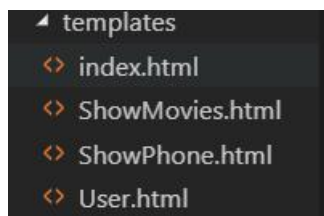


在 settings 中设置路径

```
STATIC_URL = '/static/'
STATICFILES_DIRS = (
os.path.join(BASE_DIR, 'static'),os.path.join(BASE_DIR, 'static//MyPhonePicture')
)
```

前端页面设计

网站拥有页面如下



Index.html---->首页

ShowMovies.html--->展示电影（是展示爬到的豆瓣电影的数据）网页

ShowPhone.html--->展示手机（展示的是动态爬取的京东手机商品的数据）网页

User.html--->用户信息网页

Index.html:



代码:

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>Bootstrap 实例 - 响应式的导航栏</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>
  <style>
    .fakeimg {
      height: 200px;
      background: #aaa;
    }
  </style>
</head>
<body>

<nav class="navbar navbar-inverse" role="navigation">
  <div class="container-fluid">
    <div class="navbar-header">
      <button type="button" class="navbar-toggle" data-toggle="collapse"
        data-target="#example-navbar-collapse">
        <span class="sr-only">切换导航</span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </button>
    </div>
  </div>
</nav>
```

```

    <a class="navbar-brand" href="#">Python</a>
</div>
<div class="collapse navbar-collapse" id="example-navbar-collapse">
    <ul class="nav navbar-nav">
        <li class="active"><a href="http://127.0.0.1:8000/ShowMovies/">Movies</a></li>
        <li><a href="http://127.0.0.1:8000/ShowPhone/">JD_Phone</a></li>
        <li class="dropdown">
            <a href="#" class="dropdown-toggle" data-toggle="dropdown">
                Help <b class="caret"></b>
            </a>
            <ul class="dropdown-menu">
                <li><a href="#">User_Information</a></li>
                <li><a href="#">EJB</a></li>
                <li><a href="#">Jasper Report</a></li>
                <li class="divider"></li>
                <li><a href="#">分离的链接</a></li>
                <li class="divider"></li>
                <li><a href="#">另一个分离的链接</a></li>
            </ul>
        </li>
    </ul>
    <ul class="nav navbar-nav navbar-right">
        <li><a href="http://127.0.0.1:8000/User/"><span class="glyphicon glyphicon-user"></span>
        用户信息</a></li>
        <li><a href="#"><span class="glyphicon glyphicon-user"></span> 注册</a></li>
        <li><a href="#"><span class="glyphicon glyphicon-log-in"></span> 登录</a></li>
    </ul>
</div>
</div>
</nav>
<div class="container">
    <div class="row">
        <div class="col-sm-4">
            <h2>关于 Python</h2>
            <h5>我的照片:</h5>
            <div class="fakeimg">这边插入图像</div>
            <p>关于我的介绍..</p>
            <h3>链接</h3>
            <p>描述文本。</p>
            <ul class="nav nav-pills nav-stacked">
                <li class="active"><a href="#">链接 1</a></li>
                <li><a href="#">链接 2</a></li>
                <li><a href="#">链接 3</a></li>
            </ul>

```

```
<hr class="hidden-sm hidden-md hidden-lg">

</div>

<div class="col-sm-8">

  <h2>标题</h2>

  <h5>副标题</h5>

  <div class="fakeimg">图像</div>

  <p>一些文本..</p>

  <p>菜鸟教程，学的不仅是技术，更是梦想！！！ 菜鸟教程，学的不仅是技术，更是梦想！！！ 菜鸟教程，学的不仅是技术，更是梦想！！</p>

  <br>

  <h2>标题</h2>

  <h5>副标题</h5>

  <div class="fakeimg">图像</div>

  <p>一些文本..</p>

  <p>菜鸟教程，学的不仅是技术，更是梦想！！！ 菜鸟教程，学的不仅是技术，更是梦想！！！ 菜鸟教程，学的不仅是技术，更是梦想！！</p>

</div>

</div>

</div>
```

```
<div class="jumbotron text-center" style="margin-bottom:0">

  <p>底部内容</p>

</div>

</body>

</html>
```

ShowMovies.html:



代码:

```
<!DOCTYPE html>

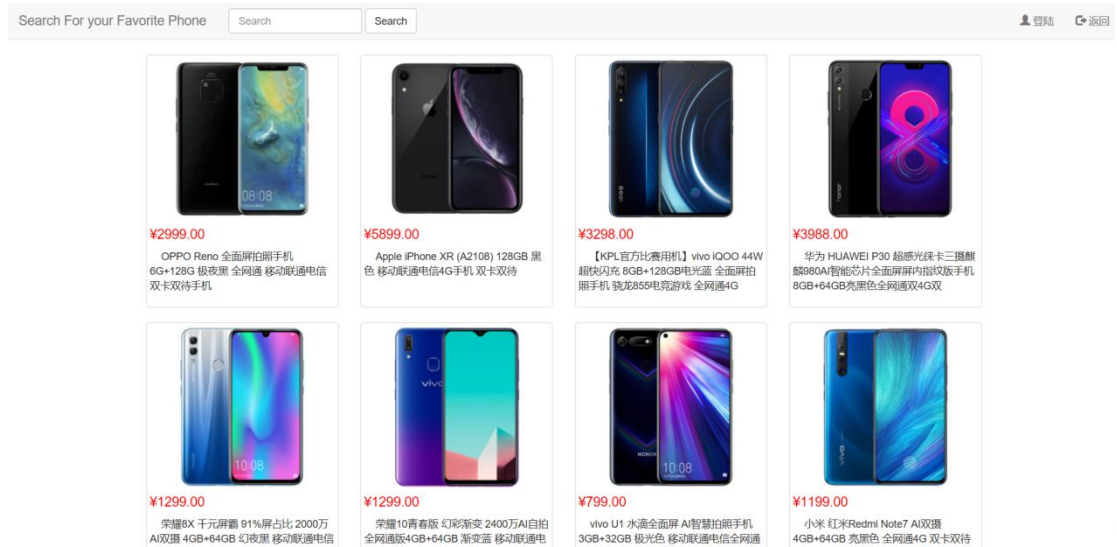
{% load staticfiles %}
```

```

<html>
<head>
<meta charset="UTF-8">
<link href="{% static 'bootstrap/css/bootstrap.min.css' %}" rel="stylesheet" type="text/css">
<title></title>
</head>
<body>
<nav class="navbar navbar-default" role="navigation">
<div class="container-fluid">
<div class="navbar-header">
<a class="navbar-brand" href="#">Search For your Favorite Movie</a>
</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="http://127.0.0.1:8000/index/"><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>TOP250-DB_Movies</h4></th></tr>
<tr><td>ranking list </td><td>movies_Name</td></tr>
{% for movie in movie_list %}
<tr>
<td>{{movie.movie_id}}</td>
<td>{{movie.movie_name}}</td>
</tr>
{% endfor %}
</table>
</div>
</body>

```

ShowPhone.html:



代码:

```
<!DOCTYPE html>
{% load staticfiles %}
<html>
<head>
<meta charset="UTF-8">
<link href="{% static 'bootstrap/css/bootstrap.min.css' %}" rel="stylesheet" type="text/css">
<title></title>
</head>
<body>
<nav class="navbar navbar-default" role="navigation">
<div class="container-fluid">
<div class="navbar-header">
<a class="navbar-brand" href="#">Search For your Favorite Phone</a>
</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="http://127.0.0.1:8000/index/"><span class="glyphicon glyphicon-log-out"></span> 返回</a></li>
</ul>
</div>
</nav>
<div class="container">
<div class="row paddtop">
```



```
{% for phone in phone_list %}
<div class="col-md-3">
<div class="thumbnail" style="height:345px">

<h4 style="color:red;">¥{{phone.pPrice}}</h4>
    {{phone.pContent}}
</div>
</div>
{% endfor%}
</div>
</div>
</body>
```

User.html

Python
Movies
JD_Phone
Help

Personal Information

Name:

李钻

Sno:

16219111329

Class:

16计算机科学与技术三

代码:

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>Python</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-inverse" role="navigation">
<div class="container-fluid">
<div class="navbar-header">
<button type="button" class="navbar-toggle" data-toggle="collapse"
data-target="#example-navbar-collapse">
<span class="sr-only">☰☰☰☰☰☰</span>
<span class="icon-bar"></span>
<span class="icon-bar"></span>
<span class="icon-bar"></span>
</button>
```

```

<a class="navbar-brand" href="#">Python</a>
</div>
<div class="collapse navbar-collapse" id="example-navbar-collapse">
<ul class="nav navbar-nav">
<li class="active"><a href="http://127.0.0.1:8000/ShowMovies/">Movies</a></li>
<li><a href="http://127.0.0.1:8000/ShowPhone/">JD_Phone</a></li>
<li class="dropdown">
<a href="#" class="dropdown-toggle" data-toggle="dropdown">
Help <b class="caret"></b>
</a>
<ul class="dropdown-menu">
<li><a href="#">User_Information</a></li>
<li><a href="#">EJB</a></li>
<li><a href="#">Jasper Report</a></li>
<li class="divider"></li>
<li><a href="#">Python</a></li>
<li class="divider"></li>
<li><a href="#">c#</a></li>
</ul>
</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-in"></span> 返回</a></li>
</ul>
</div>
</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;"><h5>Personal Imformation</h5></th></tr>
<tr><td>Name:</td><td>{{sname}}</td></tr>
<tr><td>Sno:</td><td>{{sno}}</td></tr>
<tr><td>Class:</td><td>{{sclass}}</td></tr>
</table>
</div>
</body>

```

Mysql 数据库:



testmodel_phones
testmodel_test

Phones:

对象 testmodel_phones @test (lz...						
新建 保存 另存为 添加栏位 插入栏位 删除栏位 主键 上移 下移						
栏位	索引	外键	触发器	选项	注释	SQL 预览
名	类型	长度	小数点	不是 null		
id	int	11	0	<input checked="" type="checkbox"/>	1	
pNo	varchar	7	0	<input checked="" type="checkbox"/>		
pBrand	varchar	100	0	<input checked="" type="checkbox"/>		
pPrice	varchar	20	0	<input checked="" type="checkbox"/>		
pContent	varchar	100	0	<input checked="" type="checkbox"/>		
pFile	varchar	20	0	<input checked="" type="checkbox"/>		

test:

名	类型	长度	小数点	不是 null	
id	int	11	0	<input checked="" type="checkbox"/>	1
movie_id	varchar	10	0	<input checked="" type="checkbox"/>	
movie_name	varchar	20	0	<input checked="" type="checkbox"/>	