

配置:

Settings.py

```
DATABASES = {

    'default': {

        'ENGINE': 'django.db.backends.sqlite3',

        'NAME': os.path.join(BASE_DIR, 'db.sqlite3'),

    }

}

TEMPLATES = [

    {

        'BACKEND': 'django.template.backends.django.DjangoTemplates',

        'DIRS': [os.path.join(BASE_DIR, 'myapp/templates')],

        'APP_DIRS': True,

        'OPTIONS': {

            'context_processors': [

                'django.template.context_processors.debug',

                'django.template.context_processors.request',

                'django.contrib.auth.context_processors.auth',

                'django.contrib.messages.context_processors.messages',

            ],

        },

    ],

]

INSTALLED_APPS = [

    'django.contrib.admin',

    'django.contrib.auth',

    'django.contrib.contenttypes',

    'django.contrib.sessions',

    'django.contrib.messages',

    'django.contrib.staticfiles',

    'myapp',

]
```

Urls.py

```
from django.contrib import admin

from django.urls import include, path

from myapp import views

urlpatterns = [

    path('index/', include('myapp.urls')),

]
```

```
path('admin/', admin.site.urls),  
]
```

myapp/models.py(定义模型, 并迁移)

```
from django.db import models  
class movie(models.Model):  
    mname=models.CharField(max_length=20)  
    actor=models.CharField(max_length=20)  
  
    def __str__(self):  
        return self.mname  
  
    def short_actor(self):  
        return self.actor[:50]+'....'
```

迁移之后的 0001_initial.py 文件

```
from django.db import migrations, models  
class Migration(migrations.Migration):  
  
    initial = True  
  
    dependencies = [  
    ]  
  
    operations = [  
        migrations.CreateModel(  
            name='movie',  
            fields=[  
                ('id', models.AutoField(auto_created=True, primary_key=True,  
serialize=False, verbose_name='ID')),  
                ('mname', models.CharField(max_length=20)),  
                ('actor', models.CharField(max_length=20)),  
                ('ms', models.CharField(max_length=50)),  
            ],  
        ),  
    ]
```

myapp/get.py(爬取豆瓣上的数据, 保存至数据库中)

```
import requests  
from bs4 import BeautifulSoup
```

```

movies=[]

def get_message():
    url='https://movie.douban.com/chart'
    result=requests.get(url,timeout=10)
    soup=BeautifulSoup(result.text,'html.parser')
    getdata = soup.find_all('div', class_='pl2')
    for i in getdata:
        title = i.a.text.split('/')[0]
        actor = i.p.text
        movies.append({'title': title, 'actor': actor})

    return movies

```

myapp/views.py

```

from django.shortcuts import render
from .models import movie
from .get import get_message

# Create your views here.
from django.http import HttpResponse
def show(request):
    movies=get_message()
    if movie.objects.count()==0:
        for each in movies:
            addition = movie(mname=each['title'], actor=each['actor'])
            addition.save()

    data=movie.objects
    return render(request,'movie.html',{'data':data})

```

myapp/urls.py

```

from django.urls import path
from . import views

urlpatterns=[
    path('',views.show)
]

```

在 myapp 下创建 templates 文件在其中放入 html 文件

Movie.py

```
<!DOCTYPE html>
<head>
  <meta charset="UTF-8">
  <title>豆瓣新片榜</title>
  <style type="text/css">
    h1 { margin:0; padding:0; color:red; font:normal 42px/1.2em Arial, Helvetica, sans-
    serif; letter-spacing:-3px;background-color: azure}
    span { color:#3e2d27;font-size: 30px}
    small { display:block; padding-left:80px; font:normal 15px/1.2em Arial, Helvetica,
    sans-serif; color:#959595; letter-spacing:normal;}
    td {align-items: center;
        color:deepskyblue;}
  </style>
</head>
<body>
<h1 align="center">欢迎<span>来到我的网站</span><small>welcome to my website</small></h1>
<h2 align="center">豆瓣新片前十</h2>
<table border="0" align="center">
  <tr>
    <td style="color: blue">
      电影名称
    </td>
    <td style="color: blue" align="center">
      演员
    </td>
  </tr>
  {% for i in data.all %}
    <tr>
      <td>
        《{{ i.mname }}》
      </td>
      <td>
        {{ i.short_actor }}
      </td>
    </tr>
  {% endfor %}
</table>
</body>
</html>
```

截图如下

Django 管理

用户名:

ipj

密码:

.....

登录

Django 管理

站点管理

MYAPP		
Movies	+ 增加	✎ 修改
认证和授权		
用户	+ 增加	✎ 修改
组	+ 增加	✎ 修改

最近动作

我的动作

无可用的

选择 movie 来修改

动作

-----▼

执行

 10 个中 0 个被选

<input type="checkbox"/>	MOVIE
<input type="checkbox"/>	何以为家
<input type="checkbox"/>	神探蒲松龄
<input type="checkbox"/>	谁先爱上他的
<input type="checkbox"/>	神奇动物：格林德沃之罪
<input type="checkbox"/>	无敌破坏王2：大闹互联网
<input type="checkbox"/>	忌日快乐2
<input type="checkbox"/>	“大”人物
<input type="checkbox"/>	极限职业
<input type="checkbox"/>	海王
<input type="checkbox"/>	疯狂的外星人

10 movies

欢迎来到我的网站

welcome to my website

豆瓣新片前十

电影名称	演员
《 疯狂的外星人 》	2019-02-05(中国大陆) / 黄渤 / 沈腾 / 汤姆·派福瑞 / 马修·莫里森 / 徐峥
《 海王 》	2018-12-07(中国大陆) / 2018-12-21(美国) / 杰森·莫玛 / 艾梅柏·希尔....
《 极限职业 》	2019-01-23(韩国) / 柳承龙 / 李东辉 / 李荷妮 / 陈善圭 / 申河均 / 孔明
《 “大” 人物 》	2019-01-10(中国大陆) / 王千源 / 包贝尔 / 王迅 / 王砚辉 / 屈菁菁 / 周游....
《 忌日快乐2 》	2019-02-13(美国) / 杰西卡·罗德 / 伊瑟尔·布罗萨德 / 露比·莫迪恩 / 苏拉·沙....
《 无敌破坏王2：大闹互联网 》	2018-11-21(美国) / 2018-11-23(中国大陆) / 约翰·C·赖利 / 萨拉·西....
《 神奇动物：格林德沃之罪 》	2018-11-16(中国大陆) / 2018-11-16(美国) / 埃迪·雷德梅恩 / 凯瑟琳....
《 谁先爱上他的 》	2018-04-22(乌迪内远东电影节) / 2018-11-02(台湾) / 邱泽 / 谢盈萱 /....
《 神探蒲松龄 》	2019-02-05(中国大陆) / 成龙 / 阮经天 / 钟楚曦 / 林柏宏 / 林鹏 / 乔杉
《 何以为家 》	2018-05-17(戛纳电影节) / 2018-09-20(黎巴嫩) / 2019-04-29(中....

<https://github.com/blackioker/ct/tree/master/ct>