链滴

Django_ 学员管理后台系统开发 (二)

作者：yuanhenglizhen
原文链接：https://ld246.com/article/1563377706732
来源网站：链滴

继上次的内容，我们继续bullettrain_side

1.前端页面

view.py

```
from django.shortcuts import render

# Create your views here.

def index(request):

words = 'Hi,guys'

return render(request, 'index.html', context={'words':words})
```

定义了函数index，接受request参数（这是对用户发过来的http请求的封装）

render函数顾名思义，即将内容渲染到模版

index.html就是templates模版，Django会在每个在settings里面注册的应用中寻找当前应用的模版,序是自上而下,所以这边我们自己手动建立一个templates文件夹,并在里面创建index.html，内容如下：

index.html

```
<html>

<head>

<title>学员管理平台-by mufengs</title>

</head>

<body>

2019 {{ words }}

</body>

</html>
```

这边的{{ words }}就是取view.py那边words的值

urls.py

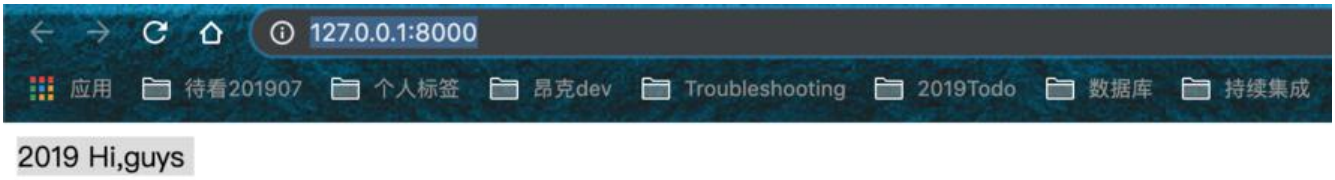
配置访问路由规则

```
from django.conf.urls import url
```

```
from django.contrib import admin
from django.urls import path
from student_sys.views import index
urlpatterns = [
url(r'^$', index, name='index'),
path('admin/', admin.site.urls),
]
```

访问：http://127.0.0.1/

就会出现下面的输出内容了



2.操作数据库

通过操作数据库，将内容渲染到页面上

修改view.py

```
from django.shortcuts import render
from .models import Student
# Create your views here.
def index(request):
students = Student.objects.all()
return render(request, 'index.html', context={'students':students})
```

从数据库中取出学员信息，接下来修改我们的模版

```
<html>
<head>
<title>学员管理平台-by mufengs</title>
</head>
<body>
<ul>
{% for student in students %}
<li>{{ student.name }} - {{ student.get_status_display }}</li>
{% endfor %}
</ul>
</body>
</html>
```

这边发现上次文章中的一个问题，注册app的时候写错了，所以这边一直报错过不去

```
RuntimeError: Model class student_sys.models.Student doesn't declare an explicit app_label a
d isn't in an application in INSTALLED_APPS
```

因为表是空的，所以页面打开是空白的

接下来我们来实现提交功能，创建一个form.py和view.py同级

```
from django import forms
from .models import Student
# 方法一
# class StudentForm(forms.Form):
# name = forms.CharField(label='name', max_length=128)
# sex = forms.ChoiceField(label='sex', choices=Student.SEX_ITEMS)
# profession = forms.CharField(label='profession', max_length=128)
# email = forms.EmailField(label='email', max_length=128)
# qq = forms.CharField(label='qq', max_length=128)
```

```
# phone = forms.CharField(label='phone', max_length=128)
#
# 方法二
class StudentForm(forms.ModelForm):
class Meta:
model = Student
fields = (
'name', 'sex', 'profession',
'email', 'phone', 'qq'
)
```

这边用了两种方法来实现，很明显第二种方法更方便

加入表单验证

最后的视图部分代码如下

views.py

```
from django.http import HttpResponseRedirect
from django.shortcuts import render
from django.urls import reverse
from student_sys.form import StudentForm
from .models import Student
# Create your views here.
def index(request):
students = Student.get_all()
if request.method == 'POST':
form = StudentForm(request.POST)
if form.is_valid():
# cleaned_data = form.cleaned_data
```

```
# student = Student()
# student.name = cleaned_data['name']
# student.sex = cleaned_data['sex']
# student.email = cleaned_data['email']
# student.profession = cleaned_data['profession']
# student.qq = cleaned_data['qq']
# student.phone = cleaned_data['phone']
# student.save()
form.save()
return HttpResponseRedirect(reverse('index'))
else:
form = StudentForm()
context = {
'students': students,
'form': form
}
return render(request, 'index.html', context=context)
```

模版文件如下

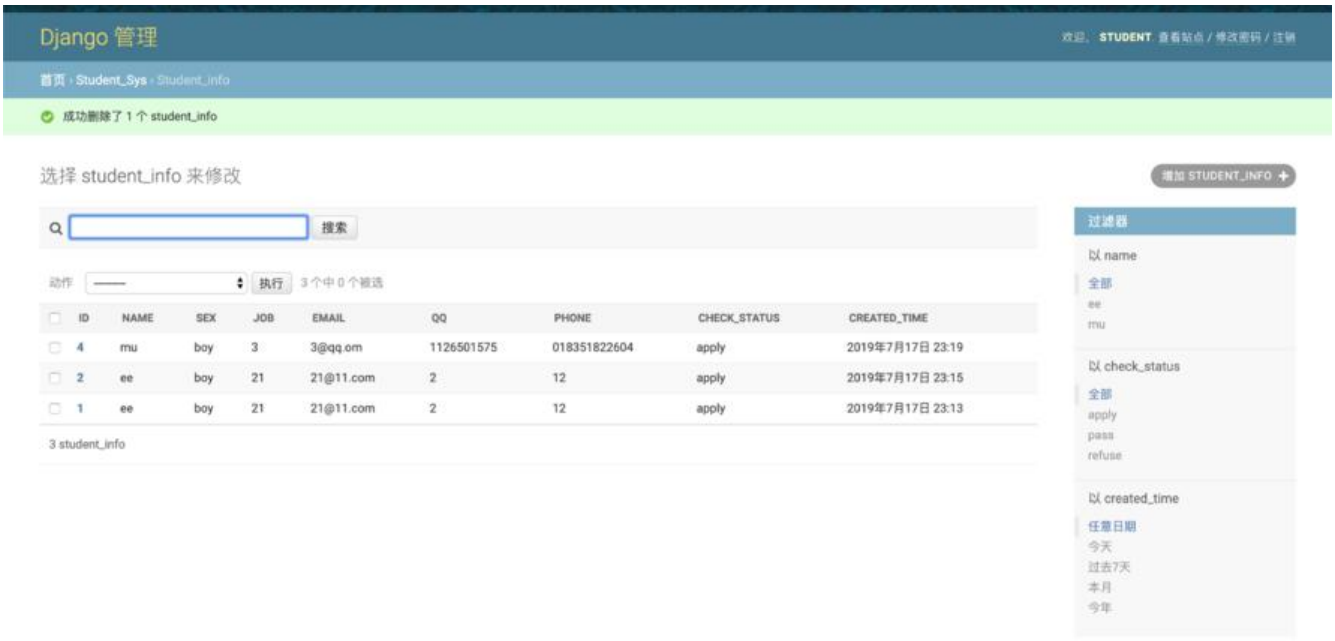
```
<html>
<head>
<title>学员管理平台-by mufengs</title>
</head>
<body>
<h3><a href="/admin/">Admin</a></h3>
<ul>
{% for student in students %}
```

```
<li>{{ student.name }} - {{ student.get_status_display }}</li>
{% endfor %}
</ul>
<form action="/" method="post">
{% csrf_token %}
{{ form }}
<input type="submit" value="Submit" />
</form>
</body>
</html>
```

又到了最后的效果时间了，曾经有人和我说过，做开发最开心的时候，做出一个功能，是有一种别人解不了的成就感

当然我不是一个开发，我也没有所谓的成就感，只是兴趣使然，做自己喜欢的事情

后台



前台

今天就搞这些吧，还需要多理解多消化，盲目的跟着书本敲代码，也没多大用处

@lizhongyue248 你那个问题就一个标签的事情，没有那么复杂啊，聊天发不了标签 @88250 ，所这边顺便给你看下我的

```
<amp-auto-ads type="adsense"
              data-ad-client="xxx">
</amp-auto-ads>
<script async src="//pagead2.googlesyndication.com/pagead/js/adsbygoogle.js"></script>
<ins class="adsbygoogle"
     style="display:block; text-align:center;"
     data-ad-layout="in-article"
     data-ad-format="fluid"
     data-ad-client="xxx"
     data-ad-slot="xxx"></ins>
<script>
     (adsbygoogle = window.adsbygoogle || []).push({});
</script>
```