



# django实战项目 内容管理系统cms (六)视图层views.py2/4

D.Roger · 2016年6月28日 · 发表评论

谷雨：穀雨之日桐始华，又五日田鼠化为鴽，又五日虹始见。



我们回到veivs.py这个文件，接着写其他的处理函数

## 1. 配置urls

修改cms/focus/urls.py这个文件：

```
1 from django.conf.urls import include, url
2 from . import views
3
4 urlpatterns = [
5     url(r'^$', views.index, name='index'),
6     url(r'^register/$', views.register, name='register'),
7     url(r'^login/$', views.log_in, name='login'),
8     url(r'^logout/$', views.log_out, name='logout'),
9     url(r'^(?P<article_id>[0-9]+)/comment/$', views.comment, name='comment'),
10    url(r'^(?P<article_id>[0-9]+)/keep/$', views.get_keep, name='keep'),
11    url(r'^(?P<article_id>[0-9]+)/poll/$', views.get_poll_article, name='poll'),
12    url(r'^(?P<article_id>[0-9]+)/$', views.article, name='article'),
13
14 ]
```

涉及到数字的正则匹配和命名组的知识，请自行查阅文档。

## 2. 添加各种表单

修改cms/focus/forms.py这个文件:

```
1 from django import forms
2
3 class LoginForm(forms.Form):
4     uid = forms.CharField(widget=forms.TextInput(attrs={'class':'form-control' , 'id':'uid', '
5     pwd = forms.CharField(widget=forms.PasswordInput(attrs={'class':'form-control' , 'id':'pwd'
6
7 class RegisterForm(forms.Form):
8     username = forms.CharField(label='username', max_length=100,
9         widget=forms.TextInput(attrs={'id':'username', 'onblur': 'authentication()'}))
10    email = forms.EmailField()
11    password1 = forms.CharField(widget=forms.PasswordInput)
12    password2 = forms.CharField(widget=forms.PasswordInput)
13
14 class SetInfoForm(forms.Form):
15     username = forms.CharField()
16
17 class CommentForm(forms.Form):
18     comment = forms.CharField(label='', widget=forms.Textarea(attrs={'cols': '60', 'rows': '6
19
20 class SearchForm(forms.Form):
21     keyword = forms.CharField(widget=forms.TextInput)
```

这个就不介绍了，之前有讲过，不熟的查文档。

## 3. 登入和登出:

在cms/focus/views.py这个文件中添加以下代码:

```
1 def log_in(request):
2     if request.method == 'GET':
3         form = LoginForm()
4         return render(request, 'login.html', {'form': form})
5     if request.method == 'POST':
6         form = LoginForm(request.POST)
7         if form.is_valid():
8             username = form.cleaned_data['uid']
9             password = form.cleaned_data['pwd']
10            user = authenticate(username=username, password=password)
11            if user is not None:
12                login(request, user)
13                url = request.POST.get('source_url', '/focus')
14                return redirect(url)
15            else:
16                return render(request, 'login.html', {'form': form, 'error': "password or usern
17
18        else:
19            return render(request, 'login.html', {'form': form})
20
21 @login_required
22 def log_out(request):
23
24     url = request.POST.get('source_url', '/focus/')
25     logout(request)
26     return redirect(url)
```

登录这个处理函数有两种可能，一是用户点击进入了登录页面login.html，请求方式是get，另外是ta在其它页面或者登录页面点击了‘登录’这个按钮，请求方式是post，所以需要先判断。

我们之前开始写views.py的时候，有导入三个django验证系统内置的方法

```
1 from django.contrib.auth import authenticate, login, logout
```

相关知识查阅 [django 文档](#)。

注意：我们定义登入，登出函数的时候不要命名为login,logout 这样会和引入的相关方法重名，会发生错误，平常命名也要注意是否该名字已被占用。另外，命名时也要尽量表示出这个函数/变量/类的作用和意图，让人一看就知道它是干什么的

@login\_required这个装饰器是django内置的，它的作用是使所装饰的函数必须是登录的用户才继续运行，不然进入指定的login\_url, 参见 [django 文档](#)，我们这里没有指定login\_url，所以在settings.py中添加以下代码：

```
1 LOGIN_URL = "/focus/login/?next='article_id'"
```

next表示登录后自动进入的页面。

标签： Django

««[django实战项目 内容管理系统cms \(五\)模版层](#)

[django实战项目 内容管理系统cms \(七\)视图层views.py3/4](#)»»

暂无评论

发表评论

已登入为D.Roger。登出？

评论

发表评论

