Django 视图与网址

Django中网址是写在 urls.py 文件中，用正则表达式对应 views.py 中的一个函数(或者generic类),我们用一个项目来演示。

一，首先，新建一个项目(project), 名称为 mysite

django-admin startproject mysite

运行后,如果成功的话, 我们会看到如下的目录样式

mysite

├── manage.py

└── mysite

    ├── \_\_init\_\_.py

    ├── settings.py

    ├── urls.py

    └── wsgi.py

我们会发现执行命令后，新建了一个 **mysite** 目录，其中还有一个 **mysite** 目录，这个子目录 mysite 中是一些项目的设置**settings.py** 文件，总的urls配置文件 **urls.py** 以及部署服务器时用到的**wsgi.py** 文件， **\_\_init\_\_.py**是python包的目录结构必须的，与调用有关。

我们到外层那个 mysite 目录下(不是mysite中的mysite目录)

二, 新建一个应用(app), 名称叫 learn

python manage.py startapp learn # learn 是一个app的名称

我们可以看到mysite中多个一个 learn 文件夹，其中有以下文件。

learn/

├── \_\_init\_\_.py

├── admin.py

├── models.py

├── tests.py

└── views.py

**注：Django 1.8.x 以上的，还有一个 migrations 文件夹。把我们新定义的app加到settings.py中的INSTALL\_APPS中**

修改 mysite/mysite/settings.py

INSTALLED\_APPS = (

    'django.contrib.admin',

    'django.contrib.auth',

    'django.contrib.contenttypes',

    'django.contrib.sessions',

    'django.contrib.messages',

    'django.contrib.staticfiles',

    'learn',

)

备注,这一步是干什么呢? 新建的 app 如果不加到 INSTALL\_APPS 中的话, django 就不能自动找到app中的**模板文件**(app-name/templates/下的文件)和**静态文件**(app-name/static/中的文件) , 后面你会学习到它们分别用来干什么.

**定义视图函数（访问页面时的内容）**

我们在learn这个目录中,把views.py打开,修改其中的源代码,改成下面的

#coding:utf-8

from django.http import HttpResponse

def index(request):

    return HttpResponse(u"欢迎光临 千峰!")

第一行是声明编码为utf-8, 因为我们在代码中用到了中文,如果不声明就报错.

第二行引入HttpResponse，它是用来向网页返回内容的，就像Python中的 print 一样，只不过 HttpResponse 是把内容显示到网页上。

我们定义了一个**index()**函数，第一个参数必须是 **request**，与网页发来的请求有关，request 变量里面包含**get**或**post**的内容，用户浏览器，系统等信息在里面（后面会讲，先了解一下就可以）。

函数返回了一个 HttpResponse 对象，可以经过一些处理，最终显示几个字到网页上。

那问题来了，我们访问什么网址才能看到刚才写的这个函数呢？怎么让网址和函数关联起来呢？

定义视图函数相关的URL(网址)  （即规定 访问什么网址对应什么内容）

我们打开 mysite/mysite/urls.py 这个文件, 修改其中的代码:

from django.conf.urls import url

from django.contrib import admin

from learn import views as learn\_views  # new

urlpatterns = [

    url(r'^$', learn\_views.index),  # new

    url(r'^admin/', admin.site.urls),

]

以上都修改并保存后,我们来看一下效果!

在终端上运行 **python manage.py runserver**我们会看到类似下面的信息:

$ python manage.py runserver

Performing system checks...

System check identified no issues (0 silenced).

You have unapplied migrations; your app may not work properly until they are applied.

Run 'python manage.py migrate' to apply them.

December 22, 2015 - 11:57:33

Django version 1.9, using settings 'mysite.settings'

Starting development server at http://127.0.0.1:8000/

Quit the server with CONTROL-C.

我们打开浏览器,访问 <http://127.0.0.1:8000/>

**注意：**如果是在另一台电脑上访问要用 python manage.py ip:port 的形式，比如监听所有ip:

python manage.py runserver 0.0.0.0:8000

监听机器上所有ip 8000端口，访问时用电脑的ip代替 127.0.0.1

Django中的 urls.py 用的是正则进行匹配的。

Django 视图与网址进阶

一 、在网页上做加减法

1. 采用 **/add/?a=4&b=5**这样GET方法进行

django-admin.py startproject qf\_views

cd qf\_views

python manage.py startapp calc

自动生成目录大致如下（因不同的 Django 版本有一些差异，暂时可以忽略他们之间的差异）：

qf\_views/

├── calc

│   ├── \_\_init\_\_.py

│   ├── admin.py

│   ├── models.py

│   ├── tests.py

│   └── views.py

├── manage.py

└── qf\_views

    ├── \_\_init\_\_.py

    ├── settings.py

    ├── urls.py

    └── wsgi.py

我们修改一下 **calc/views.py**文件

from django.shortcuts import render

from django.http import HttpResponse

def add(request):

    a = request.GET['a']

    b = request.GET['b']

    c = int(a)+int(b)

    return HttpResponse(str(c))

注：request.GET 类似于一个字典，更好的办法是用 request.GET.get('a', 0) 当没有传递 a 的时候默认 a 为 0

接着修改 qf\_views/urls.py 文件，添加一个网址来对应我们刚才新建的视图函数。

from django.conf.urls import url

from django.contrib import admin

from calc import views as calc\_views

urlpatterns = [

    url(r'^add/$', calc\_views.add, name='add'),  # 注意修改了这一行

    url(r'^admin/', admin.site.urls),

]

我们打开开发服务器并访问

python manage.py runserver

如果提示 Error: That port is already in use.在后面加上端口号8001,8888等

python manage.py runserver 8001

打开网址：[http://127.0.0.1:8000**/add/**](http://127.0.0.1:8000/add/) 就可以看到

**MultiValueDictKeyError at /add/**



**这是因为我们并没有传值进去，我们在后面加上 ?a=4&b=5，即访问**[**http://127.0.0.1:8000/add/?a=4&b=5**](http://127.0.0.1:8000/add/?a=4&b=5)

就可以看到网页上显示一个 9，试着改变一下a和b对应的值试试看？

2. 采用 /add/3/4/ 这样的网址的方式

前面介绍的时候就说过 Django 支持优雅的网址

我们接着修改 **calc/views.py**文件，再新定义一个add2 函数，原有部分不再贴出

def add2(request, a, b):

    c = int(a) + int(b)

    return HttpResponse(str(c))

接着修改 qf\_views/urls.py 文件，再添加一个新的 url

    url(r'^add/(\d+)/(\d+)/$', calc\_views.add2, name='add2'),

我们可以看到网址中多了 (\d+), 正则表达式中 \d 代表一个数字，+ 代表一个或多个前面的字符，写在一起 \d+ 就是一个或多个数字，用括号括起来的意思是保存为一个**子组**，**每一个子组将作为一个参数，被 views.py 中的对应视图函数接收**。

我们再访问 <http://127.0.0.1:8000/add/4/5/> 就可以看到和刚才同样的效果，但是这回网址更优雅了

# Django URL name详解

**1. 打开 qf\_views/urls.py**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | from django.conf.urls import url  from django.contrib import admin  from calc import views as calc\_views      urlpatterns = [      url(r'^add/$', calc\_views.add, name='add'),      url(r'^add/(\d+)/(\d+)/$', calc\_views.add2, name='add2'),      url(r'^admin/', admin.site.urls),  ] |

url(r'^add/$', calc\_views.add, **name='add'**), 这里的 **name='add'**是用来干什么的呢？

简单说，name 可以用于在 templates, models, views ……中得到对应的网址，相当于“给网址取了个名字”，只要这个名字不变，网址变了也能通过名字获取到。

为了进一步弄清这个问题，我们先建一个首页的视图和url

**2. 修改 calc/views.py**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9 | from django.http import HttpResponse  from django.shortcuts import render      def index(request):      return render(request, 'home.html')      ...此处省去一些代码 |

render 是渲染模板，不懂先照着打就好。

3. 将 'calc' 这个 app 加入到 qf\_views/settings.py 中

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | INSTALLED\_APPS = [      'django.contrib.admin',      'django.contrib.auth',      'django.contrib.contenttypes',      'django.contrib.sessions',      'django.contrib.messages',      'django.contrib.staticfiles',        'calc',  ] |

这样，使用render的时候，Django 会自动找到 INSTALLED\_APPS 中列出的各个 app 下的 templates 中的文件。

4. 我们在 calc 这个 app 中新建一个 templates 文件夹，在templates中新建一个 home.html （关于模板更详细的以后学习）

文件 calc/templates/home.html 中写入以下内容（**保存时用 utf8 编码**）

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | <!DOCTYPE html>  <html>  <head>      <title>千峰</title>  </head>  <body>    <a href="/add/4/5/">计算 4+5</a>    </body>  </html> |

修改 qf\_views/urls.py

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | ...此处省去一些代码    urlpatterns = [      url(r'^$', calc\_views.index, name='home'),      url(r'^add/$', calc\_views.add, name='add'),      url(r'^add/(\d+)/(\d+)/$', calc\_views.add2, name='add2'),      url(r'^admin/', admin.site.urls),  ] |

运行开发服务器，我们访问 <http://127.0.0.1:8000/>

我们计算加法的时候用的是 /add/4/5/ ，**后来需求发生变化**，比如改成 /4\_add\_5/，但在网页中，代码中很多地方都写死的/add/4/5/，比如模板中可能是这么写的

|  |  |
| --- | --- |
| 1 | <a href="/add/4/5/">计算 4+5</a> |

**如果这样写“死网址”，会使得在改了网址（正则）后，模板**（template)**，视图**(views.py，用以用于跳转)**，模型**(models.py，可以用用于获取对象对应的地址）**用了此网址的，都得进行相应的更改，修改的代价很大，一不小心，有的地方没改过来，就不能用了**。

**那么有没有更优雅的方式来解决这个问题呢？**当然答案是肯定的。

我们先说一下如何用 Python 代码获取对应的网址（可以用在 views.py，models.py等各种需要转换得到网址的地方）：

|  |  |
| --- | --- |
|  | python manage.py shell |

|  |  |
| --- | --- |
|  | >>> from django.core.urlresolvers import reverse  # django 1.4.x - django 1.10.x  >>> from django.urls import reverse  # django 1.10.x 新的，更加规范了    >>> reverse('add2', args=(4,5))  u'/add/4/5/'  >>> reverse('add2', args=(444,555))  u'/add/444/555/' |

**reverse 接收 url 中的 name 作为第一个参数**，我们在代码中就可以通过 reverse() 来获取对应的网址（这个网址可以用来跳转，也可以用来计算相关页面的地址），只要对应的 url 的name不改，就不用改代码中的网址。

在网页模板中也是一样，可以很方便的使用。

|  |  |
| --- | --- |
|  | 不带参数的：  {% url 'name' %}  带参数的：参数可以是变量名  {% url 'name' 参数 %}    例如：  <a href="{% url 'add2' 4 5 %}">link</a> |

上面的代码渲染成最终的页面是

|  |  |
| --- | --- |
| 1 | <a href="/add/4/5/">link</a> |

这样就可以通过 {% url 'add2' 4 5 %} 获取到对应的网址 /add/4/5/

当 urls.py 进行更改，前提是不改 name（这个参数设定好后不要轻易改），获取的网址也会动态地跟着变，比如改成：

|  |  |
| --- | --- |
| 1 | url(r'^new\_add/(\d+)/(\d+)/$', calc\_views.add2, name='add2'), |

注意看重点 add 变成了 new\_add，但是后面的 name='add2' 没改，这时 {% url 'add2' 4 5 %} 就会渲染对应的网址成 /new\_add/4/5/

用在 views.py 或 models.py 等地方的 reverse函数，同样会根据 name 对应的url获取到新的网址。

想要改网址的时候，修改 urls.py 中的正则表达式部分（url 参数第一部分），name 不变的前提下，其它地方都不需要修改。

另外，比如用户收藏夹中收藏的URL是旧的，如何让以前的 /add/3/4/自动跳转到现在新的网址呢？

**要知道Django不会帮你做这个，这个需要自己来写一个跳转方法**：

具体思路是，在 views.py 写一个跳转的函数：

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | from django.http import HttpResponseRedirect  from django.core.urlresolvers import reverse  # django 1.4.x - django 1.10.x  #  from django.urls import reverse  # new in django 1.10.x    def old\_add2\_redirect(request, a, b):      return HttpResponseRedirect(          reverse('add2', args=(a, b))      ) |

urls.py中：

|  |  |
| --- | --- |
| 1  2 | url(r'^add/(\d+)/(\d+)/$', calc\_views.old\_add2\_redirect),      url(r'^new\_add/(\d+)/(\d+)/$', calc\_views.add2, name='add2'), |

这样，**假如用户收藏夹中**有 /add/4/5/ ，访问时就会自动跳转到新的 /new\_add/4/5/ 了