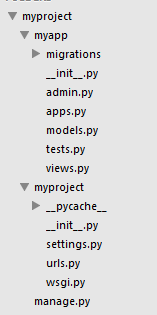
在Django框架中，提供了非常清晰简洁的url管理方法，在新建一个project之后(此处为myproject),然后在项目中建立一个app(此处为myapp)，会看到有如下的目录结构：



一般所熟知的就是在myproject/myproject/urls.py中的urlpatterns列表中来配置 url，每一个列表项就是一个由url函数的调用。例如嘉定我们想在myapp中定义一个主页，然后通过"http://localhost:8000/myapp/homepage"来访问，首先我们在myproject/myapp//view.py中定义一个叫homePage的函数（名字随意，不一定叫这名字）：

**[python]** [view plain](http://blog.csdn.net/ranyixu11/article/details/52986580) [copy](http://blog.csdn.net/ranyixu11/article/details/52986580)

1. from django.shortcuts import render
2. from django.http.response import HttpResponse
4. # Create your views here.
5. def homePage(request):
6. return HttpResponse("<h1>This is home page</h1>")

然后在myproject/myproject/urls.py的urlpatterns列表中添加一个url配置:

**[python]** [view plain](http://blog.csdn.net/ranyixu11/article/details/52986580) [copy](http://blog.csdn.net/ranyixu11/article/details/52986580)

1. from django.conf.urls import url
2. from django.contrib import admin
3. from myapp.views import homePage#记得导入
5. urlpatterns = [
6. url(r'^admin/', admin.site.urls),
7. url(r'^myapp/homepage', homePage)
8. ]

然后运行项目，就可以用浏览器通过http://localhost:8000/myapp/homepage来访问。

但假如一个project中有多个app，用以上的方式来管理url可能会造成比较混乱的局面，为了解决这个问题，我们可以用include的方法来配置url，首先我们在自己的app中建立一个urls.py，即myproject/myapp/目录下建立urls.py，然后在其中输入如下内容:

**[python]** [view plain](http://blog.csdn.net/ranyixu11/article/details/52986580) [copy](http://blog.csdn.net/ranyixu11/article/details/52986580)

1. from django.conf.urls import url
3. from myapp.views import homePage

6. urlpatterns = [
7. url(r'homepage', homePage),
8. ]

然后在项目的urls中包含刚刚app中添加的url配置，我们要做的是在myproject/myproject/urls.py输入如下内容:

**[python]** [view plain](http://blog.csdn.net/ranyixu11/article/details/52986580) [copy](http://blog.csdn.net/ranyixu11/article/details/52986580)

1. from django.conf.urls import url, include#导入了include函数
2. from django.contrib import admin
4. urlpatterns = [
5. url(r'^admin/', admin.site.urls),
6. url(r'^myapp/', include("myapp.urls"))#包含myapp中的urls
7. ]

然后通过刚刚相同的url(http://localhost:8000/myapp/homepage)访问发现也可以访问了，通过这样的url管理会发现更加整洁，可扩展性更强。