

# KylinBlog开发注解-Part5

## 路由配置URLConf

Django的项目文件夹和每个应用(app)目录下都有 `urls.py` 文件，它们构成了Django的路由配置系统(URLconf)。

服务器收到用户请求后，会根据用户请求的url地址和urls.py里配置的url-视图映射关系，去调用执行相应的视图函数或视图类，最后由视图返回给客户端数据。

## URLConf工作原理

以建议博客为例：

### urls.py

```
# blog/urls.py
from django.urls import path
from . import views

urlpatterns = [
    path('blog/', views.index),
    path('blog/articles/<int:id>/', views.article_detail),
]

# blog/views.py
def index(request):
    # 展示所有文章

def article_detail(request, id):
    # 展示某篇具体文章
```

- 当用户在浏览器输入 `/blog/` 时，URL收到请求后会调用视图 `views.py` 里的 `index` 方法，展示所有文章
- 当用户在浏览器输入 `/blog/article/<int:id>/` 时，URL不仅调用了 `views.py` 里的 `article_detail` 方法，而且还把参数文章id通过 `<>` 括号的形式传递给了视图。int这里代表只传递整数，传递的参数名字是id。

注意：

当你配置URL时，别忘了把你的应用(blog)的urls加入到项目的URL配置里(mysite/urls.py),

## path和re\_path方法

Django提供了两种设计URL的方法: `path` 和 `re_path`，它们均支持向视图函数或类传递参数。

`path` 是正常参数传递，`re_path` 是采用正则表达式regex匹配。

- `path` 方法：采用双尖括号 `<变量类型:变量名>` 或 `<变量名>` 传递，例如 `<int:id>`，`<slug:slug>` 或 `<username>`。
- `re_path` 方法：采用命名组 `(?P<变量名>表达式)` 的方式传递参数。

```
# blog/urls.py
from django.urls import path, re_path
from . import views

urlpatterns = [
    path('blog/articles/<int:id>/', views.article_detail, name = 'article_detail'),
    re_path(r'^blog/articles/(?P<id>\d+)/$', views.article_detail, name='article_detail'),
]

# blog/views.py
def article_detail(request, id):
    # 展示某篇文章
```

在使用 `path` 和 `re_path` 方法设计urls需注意：

- url中的参数名要用尖括号，而不是圆括号；
- 匹配模式的最开头不需要添加斜杠 `/`，但建议以斜杠结尾；
- 使用 `re_path` 时不一定总是以 `$` 结尾，有时不能加。比如下例中把 `blog.urls` 通过 `re_path` 加入到项目urls中时就不能以 `$` 结尾，因为这里的 `blog/` 并不是完整的url，只是一个开头而已。

## KylinBlog系统中的urls.py

```
from django.urls import path
from django.contrib import admin
from kblog import views
from django.contrib.staticfiles.views import serve
from django.views.generic.base import RedirectView

urlpatterns = [
    # 首页
    path('', views.index, name='index'),
    # 文章详情页
    path('article/<int:id>/', views.article_detail, name='article_detail'),
    # 分类和标签页
    path('category_tag/', views.category_tag, name='category_tag'),
    # 文章分类详情页
    path('article_category/(?P<id>[0-9]+)$', views.article_category, name='article_category'),
    # 文章标签详情页
    path('article_tag/<int:id>', views.article_tag, name='article_tag'),
    # 关于
    path('about/', views.about, name='about'),
    # 登录
    path('admin/', admin.site.urls, name='login'),
    # 注册
    path('register/', views.register, name='register')
]
```

在实际开发中，KylinBlog使用了URL命名来提高代码可读性。