

# Django

xbZhong

2025-01-29

[本页PDF](#)

## 项目结构

1. `manage.py` : 与项目交互的文件, 一般在终端输入 `python manage.py` [子命令]
2. `settings.py` : 项目的设置项
3. `urls.py` : 配置url路径
4. `wsgi.py` : 项目与WSGI协议兼容的web服务器入口

## 入门

- Django中定义函数时 `request` 是一个必加的参数, 它包含了向服务器发送请求的请求体的所有信息
- 给请求体返回数据时要导入 `HttpResponse` 对象
- `app` 是项目的一个模块, 各种各样的 `app` 组成了项目
- `urlpatterns` 位于项目的 `urls.py` 当中, 可以将 `url` 与视图函数 (也就是访问这个url要执行的函数) 进行绑定

```
1  from django.contrib import admin
2  from django.http import HttpResponse
3  from django.urls import path
4  ## 视图函数
5  def welcome(request):
6      return HttpResponse("Welcome to DjangoProject")
7  urlpatterns = [
8      path('admin/', admin.site.urls),
9      path('', welcome)
10 ]
```

- 启动: `python manage.py runserver`
- 创建app: `python manage.py startapp [app名字]`

## URL携带参数

- 从请求信息中解析参数并获取
  - 使用 `request.GET.get` 解析参数, 如解析 `http://127.0.0.1:8000/app1?id=3` , 会得到3这个数字
  - 直接在网址中解析, 如 `http://127.0.0.1:8000/app1/水浒传` , 使用 `/app1/<book_name>` 的形式作为path, 调用视图函数时会进行参数传递, 只需要在视图函数中进行输出即可
    - `/app1/<book_name>` 中可以对book\_name进行类型定义, 如 `/app1/<int:book_id>`

```

1  ## app1/views.py
2  from django.http import HttpResponse
3  from django.shortcuts import render
4
5  def book_id(request):
6      id = request.GET.get('id')
7      return HttpResponse(f"您查找的图书id是{id}")
8
9  def book_name(request,book_name)
10     return HttpResponse(f'您查找的图书名称为{book_name}')
11
12
13  ## urls.py
14  from django.contrib import admin
15  from django.http import HttpResponse
16  from django.urls import path
17  ## 导入app1的views文件，将网址和视图函数进行映射
18  from app1 import views
19  def welcome(request):
20      return HttpResponse("Welcome to DjangoProject")
21  urlpatterns = [
22      path('admin/', admin.site.urls),
23      path('', welcome),
24      # http://127.0.0.1:8000/app1?id=3
25      path('app1/',views.book_id),
26      # http://127.0.0.1:8000/app1/1
27      path('app1/<book_name>',views.book_name)
28  ]

```

## path函数

- 定义为 `path(route,view,name=None,kwargs=None)`
  - route参数：这个参数可以指定url中需要传递的参数，从而在数据库中获取需要的信息。
  - view参数：可以是一个视图函数或者时 `类视图`，或者是函数返回值、
  - name参数：给这个url取个别名

## 路由模块化

- 使用 `include` 方法在项目的主 `urls` 导入app的 `urls.py` 文件，便于统一管理

```

1  # app2/urls.py 路由文件
2
3  from django.urls import path
4  from . import views
5  urlpatterns = [
6      path('list/', views.movie_list, name='movie_list'),
7      path('detail/<int:movie_id>', views.movie_detail, name='movie_detail')
8  ]
9
10 # app2/views.py 视图函数文件
11
12 from django.http import HttpResponse
13 from django.shortcuts import render
14
15 def movie_list(request):
16     return HttpResponse('电影列表')
17
18 def movie_detail(request, movie_id):
19     return HttpResponse(f'您查找的电影id为: {movie_id}')
20
21
22 # urls.py
23 from django.contrib import admin
24 from django.http import HttpResponse
25 from django.urls import path, include
26 from app1 import views
27 import app2
28
29 urlpatterns = [
30     # app2
31     path('app2/', include("app2.urls"))
32 ]

```

## 路由反转

- 使用 `reverse` 通过视图函数得到 `url`
  - 里面传递的参数是视图函数的别名和 `url` 要传递的参数

```
1  def welcome(request):
2      print(reverse('book_id_search'))
3      print(reverse('book_name_search',kwargs={'book_name':'sanguoyanyi'}))
4      return HttpResponse("Welcome to DjangoProject")
5
6  urlpatterns = [
7      path('admin/', admin.site.urls),
8      path('', welcome),
9      # http://127.0.0.1:8000/app1?id=3
10     path('app1/',views.book_id,name='book_id_search'),
11     # http://127.0.0.1:8000/app1/1
12     path('app1/<book_name>',views.book_name,name='book_name_search'),
13     # app2
14     path('app2/',include("app2.urls"))
15 ]
```

## DTL（模板渲染）

---