Web开发

# 创建项目

**特殊变量**

Urlpatterns = [

Path(route,view,kwargs,name)

]

Route：Route 是匹配URL的准则。当django响应一个请求是，它会从urlpatterns的第一项开始，按顺序依次匹配列表中的项，直到找到匹配的项。这些准则不会匹配get和post参数或域名。

View：当django站到一个匹配的准则，就会调用这个特定的视图函数，并传入一个HttpResponse对象作为第一个参数，被捕获的参数一关键字参数的形式传入。

Kwargs: 任意个关键字参数可以作为一个字典传递给目标视图函数。

Name：为你的 URL 取名能使你在 Django 的任意地方唯一地引用它，尤其是在模板中。这个有用的特性允许你只改一个文件就能全局地修改某个 URL 模式。

# 配置

打开 mysite/settings.py。这是个包含了Django项目设置的Python模块。

**数据库配置**

改变设置文件中[DATABASES](https://docs.djangoproject.com/zh-hans/2.1/ref/settings/" \l "std:setting-DATABASES) 'default' 项目中的一些键值：

[ENGINE](https://docs.djangoproject.com/zh-hans/2.1/ref/settings/#std:setting-DATABASE-ENGINE) --可选值有'django.db.backends.sqlite3'，'django.db.backends.postgresql'，'django.db.backends.mysql'，或 'django.db.backends.oracle'。其它 [可用后端](https://docs.djangoproject.com/zh-hans/2.1/ref/databases/#third-party-notes)。

[NAME](https://docs.djangoproject.com/zh-hans/2.1/ref/settings/#std:setting-NAME) - 数据库的名称。如果使用的是 SQLite，数据库将是你电脑上的一个文件，在这种情况下， [NAME](https://docs.djangoproject.com/zh-hans/2.1/ref/settings/#std:setting-NAME) 应该是此文件的绝对路径，包括文件名。默认值 os.path.join(BASE\_DIR, 'db.sqlite3') 将会把数据库文件储存在项目的根目录。

如果你不使用 SQLite，则必须添加一些额外设置，比如 [USER](https://docs.djangoproject.com/zh-hans/2.1/ref/settings/#std:setting-USER) 、 [PASSWORD](https://docs.djangoproject.com/zh-hans/2.1/ref/settings/#std:setting-PASSWORD) 、 [HOST](https://docs.djangoproject.com/zh-hans/2.1/ref/settings/#std:setting-HOST) 等等。

备注：如果你使用了 SQLite 以外的数据库，请确认在使用前已经创建了数据库。你可以通过在你的数据库交互式命令行中使用"CREATE DATABASE database\_name;" 命令来完成这件事。另外，还要确保该数据库用户中提供 mysite/settings.py 具有 "create database" 权限。这使得自动创建的[test database](https://docs.djangoproject.com/zh-hans/2.1/topics/testing/overview/" \l "the-test-database) 能被以后的教程使用。如果你使用 SQLite，那么你不需要在使用前做任何事——数据库会在需要的时候自动创建。

**时区配置**

编辑 mysite/settings.py 文件前，先设置 [TIME\_ZONE](https://docs.djangoproject.com/zh-hans/2.1/ref/settings/#std:setting-TIME_ZONE) 为你自己时区。

[**INSTALLED\_APPS**](https://docs.djangoproject.com/zh-hans/2.1/ref/settings/#std:setting-INSTALLED_APPS)**设置**

此外，关注一下文件头部的 [INSTALLED\_APPS](https://docs.djangoproject.com/zh-hans/2.1/ref/settings/#std:setting-INSTALLED_APPS) 设置项。这里包括了会在你项目中启用的所有 Django 应用。应用能在多个项目中使用，你也可以打包并且发布应用，让别人使用它们。

通常， [INSTALLED\_APPS](https://docs.djangoproject.com/zh-hans/2.1/ref/settings/#std:setting-INSTALLED_APPS) 默认包括了以下 Django 的自带应用：

[django.contrib.admin](https://docs.djangoproject.com/zh-hans/2.1/ref/contrib/admin/#module-django.contrib.admin) -- 管理员站点， 你很快就会使用它。

[django.contrib.auth](https://docs.djangoproject.com/zh-hans/2.1/topics/auth/#module-django.contrib.auth) -- 认证授权系统。

[django.contrib.contenttypes](https://docs.djangoproject.com/zh-hans/2.1/ref/contrib/contenttypes/#module-django.contrib.contenttypes) -- 内容类型框架。

[django.contrib.sessions](https://docs.djangoproject.com/zh-hans/2.1/topics/http/sessions/#module-django.contrib.sessions) -- 会话框架。

[django.contrib.messages](https://docs.djangoproject.com/zh-hans/2.1/ref/contrib/messages/#module-django.contrib.messages) -- 消息框架。

[django.contrib.staticfiles](https://docs.djangoproject.com/zh-hans/2.1/ref/contrib/staticfiles/#module-django.contrib.staticfiles) -- 管理静态文件的框架。

这些应用被默认启用是为了给常规项目提供方便。

[migrate](https://docs.djangoproject.com/zh-hans/2.1/ref/django-admin/#django-admin-migrate) 命令检查 [INSTALLED\_APPS](https://docs.djangoproject.com/zh-hans/2.1/ref/settings/#std:setting-INSTALLED_APPS) 设置，为其中的每个应用创建需要的数据表：

“python manage.py migrate”：这个 [migrate](https://docs.djangoproject.com/zh-hans/2.1/ref/django-admin/#django-admin-migrate) 命令检查 [INSTALLED\_APPS](https://docs.djangoproject.com/zh-hans/2.1/ref/settings/#std:setting-INSTALLED_APPS) 设置，为其中的每个应用创建需要的数据表，至于具体会创建什么，这取决于你的 mysite/settings.py 设置文件和每个应用的数据库迁移文件（我们稍后会介绍这个）。这个命令所执行的每个迁移操作都会在终端中显示出来。如果你感兴趣的话，运行你数据库的命令行工具，并输入 \dt (PostgreSQL)， SHOW TABLES; (MySQL)， .schema (SQLite)或者 SELECT TABLE\_NAME FROM USER\_TABLES; (Oracle) 来看看 Django 到底创建了哪些表。

# 创建模型

在 Django 里写一个数据库驱动的 Web 应用的第一步是定义模型 - 也就是数据库结构设计和附加的其它元数据。