# **一、**Django简介

## **1.Django的优点**

功能完善，容易上手，开发速度快，安全性高；

有完善的在线文档；

其模型自带数据库ORM(Object-Relational-Mapping，对象关系映射)组件，使得开发者无须学习其他数据库访问技术(如SQLAlchemy等)；

可以使用正则表达式管理路由映射，方便灵活。

## **MVC和MTV模式**

**MVC模式：**

目前的主流web框架，基本上都使用mvc模式开发web应用，其最大的优点是可以降低系统各模块间的耦合度。

Modle：模型，负责处理各个功能的实现，如增加，修改和删除，其中包含模型实体类和业务处理类；

View：视图，负责页面的显示和用户的交互，包含由HTML、CSS、JavaScript组成的各种页面；

Controller：控制器，用于将用户请求转发给相应的模型进行处理，并根据模型的处理结果向用户提供相应的相应。

**MTV模式：**

对传统的MVC模式进行了修改，修改后的模式称为MTV模式。

Model：模型，负责业务对象和数据库的关系映射；

Template：模版，负责页面的显示和用户的交互；

View：视图，负责业务逻辑，并在适当的时候调用Model和Template。

# **二、Django安装**

## 1.windows系统：

安装命令：

**pip install django==版本 -i <https://mirrors.aliyun.com/pypi/simple/>**

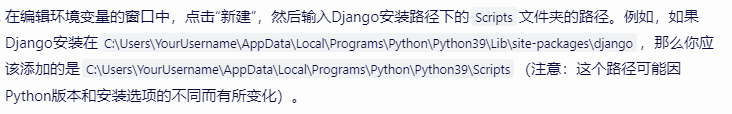
例子：

**pip install django==4.1 -i <https://mirrors.aliyun.com/pypi/simple/>**

查找Django的安装位置：

**pip show django**

配置环境变量：



查看Django版本：

**python -m django --version**

## **Linux系统：**

略

# **三、Django知识点**

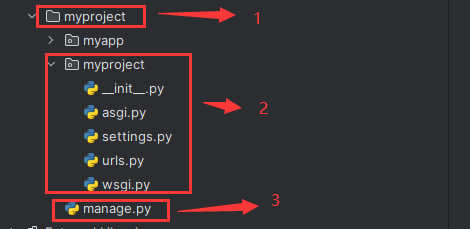
## 1.新建项目

命令：

django-admin startproject 项目名称

例子：

django-admin startproject myproject



**注意：**

manage.py：一个命令行实用程序，可以通过命令行方式与Django项目进行交互。

myproject：外面的为根目录，与Django无关，可以任意命名；里面的为项目目录，这个目录请勿修改。

asgi.py：与ASGI兼容的web服务器，为项目提供服务的入口。

settings：该项目的全局配置文件。

urls：该项目的路由配置文件。

wsgi.py：与WSGI兼容的web服务器，为项目提供服务的入口。

db.sqlit3：sqlit格式的数据库。新版本未显示。

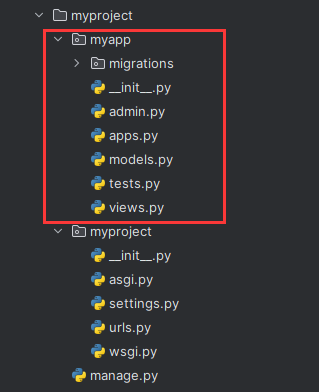
## **2.创建应用**

命令：

python manage.py startapp 应用名

例子：

python manage.py startapp myapp



**注意：**

myapp：应用目录，请勿修改这个目录名称。

admin.py：后台管理使用。

apps.py：应用管理文件。

models.py：模型文件。

tests.py：测试用例。

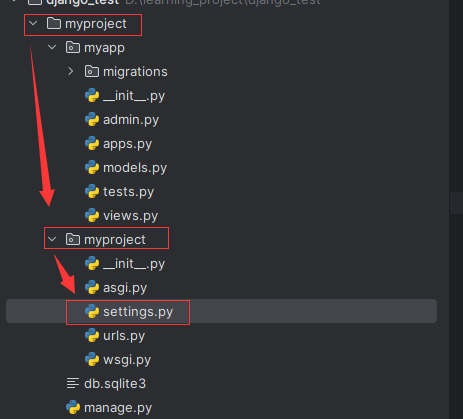
views.py：视图文件。

migrations：该目录包含数据的迁移文件。

创建应用时，需要切换至manage.py同级目录，再输入需要创建应用的命令。

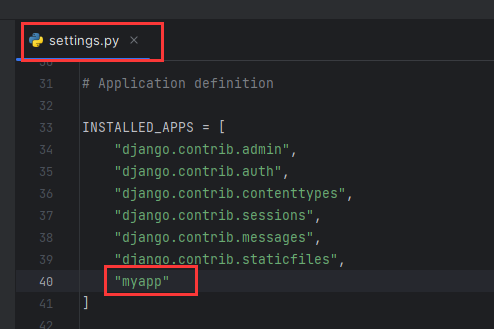
## **3.配置文件**

所有的配置信息都在setting.py文件中完成，该文件的位置如下：



### 3.1配置应用

在settings.py文件中找到INSTALLED\_APPS，将应用名称添加即可。



### **3.2配置数据库**

### **3.3配置静态文件**

### **3.4配置**

### **3.5配置**

### **3.6配置**

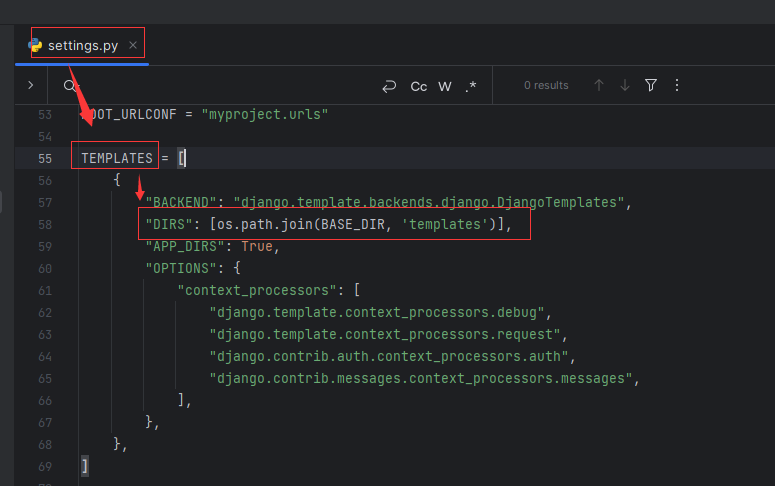
### **3.7配置**

## **4.创建模板**

使用命令行创建Django应用时，默认是没有模板目录，且目录名称为templates，可以更改，但建议保持不变。



模版创建完成后，还不能直接使用，需要在settings配置文件中进行注册。



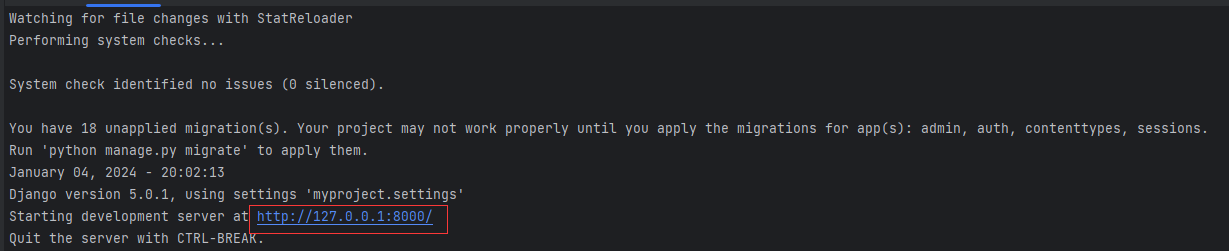
## **5.运行项目**

命令：

python manage.py runserver [port]

例子：

python manage.py runserver



显示上述信息，并在浏览器中输入“http://localhost:8000/”，如果页面有详细信息，表示Django应用运行成功。

## **6.Django的路由**

## **7.Django的视图**

## **8.Django的模版**

## **9.Django的模型**

## **10.用户认证**

## **11.Django的后台管理**

## **12.接口的设计与实现**

## **13.分层的自动化测试**

# **四、Django的开发流程**

# **五、Django的注意事项**

# **六、Django的常见问题**

# **七、学习网址**