# 1.项目搭建,参考: [django的安装和开发环境搭建](django学习笔记1-markdown基础和django环境搭建以及项目创建.docx)

## 记住:初始化django项目可以使用django-admin startproject 项目名称或者直接使用pycharm专业版来创建

# 2.搭建好开发环境并且创建了一个mysite项目把项目清理后,我们来进一步学习

## 1>项目结构和文件的作用

|  |
| --- |
|  |

## 2>django中app的概念,这个不是手机应用,而是django里面的应用.适用于比较大型的项目,把大项目分割位为一个个单一功能的app.每一个app都可以有自己的表结构,函数和html模板

|  |
| --- |
|  |

## 3>如何创建这个app?通过manage.py来创建app

### a.在终端中进入项目根目录

|  |
| --- |
|  |

### b.在终端中输入python manage.py startapp app名称,如mySiteApp,如何回车,就会创建一个mySiteApp目录

|  |
| --- |
|  |
|  |

#### 你可以创建任意多个app,但是如果项目不是很大,我们只需要使用一个app就够了

### c.可以双击一下这个app文件夹.他里面的结构如下

|  |
| --- |
|  |

### d.在这个包里面各个文件的作用如图

|  |
| --- |
|  |

## 4>用django发送第一个请求,并且返回数据

### 1.首先,你需要在settings.py中注册你的app,打开mySite包里面的settings.py文件并且打开,找到INSTALLED\_APPS

|  |
| --- |
|  |

### 2.然后我们需要在它里面注册我们的mySiteApp,写法有点绕:

|  |
| --- |
|  |

### 3.在mySiteApp里面的views.py里面编写一个函数,用来和index/路由匹配和一个home函数匹配””,注意需要导入响应类

|  |
| --- |
|  |

### 4.在mysite包中的urls.py中编写url和函数的对应关系,打开urls.py在urlpatterns列表里面把django默认的路由注释掉,我们不需要他的.

|  |
| --- |
|  |

#### 然后怎么添加我们的路由呢?

### 5.我们需要导入应用里面的views模块,然后我们来添加路由配置

|  |
| --- |
| **from** django.contrib **import** admin **from** django.urls **import** path **from** mySiteApp **import** views     urlpatterns = [  *# path('admin/', admin.site.urls),* path(**''**, views.home),  path(**'index/'**, views.index),  ] |

### 6.启动项目,方式1,使用python manage.py runserver,如果一切配置都正确,效果如下

|  |
| --- |
|  |

### 7.点击链接,就会在浏览器中打开一个页面,默认是””也就是home页面,效果如下

|  |
| --- |
|  |

### 8.然后我们在网址后面输入index/,就会匹配到index函数,效果如下

|  |
| --- |
|  |

### 方式2.使用pycharm来启动,我们先把项目停止.非常麻烦还是命令行好用.这种方式,项目必须是用pycharm创建的.

|  |
| --- |
|  |
|  |
|  |

#### 注意:这只是最基本的的响应,在实际开发在,我们需要返回一个html页面给用户,这个有如何实现?

## 5>在django中使用template模板,如果你需要返回一个html模板页面,你不能够用HttpResponse类,需要使用render函数他的格式是这样子的,这里我们做两个路由函数:用户注册和用户列表

|  |
| --- |
|  |

## 6>然后,我们需要在urls中配置路由

|  |
| --- |
|  |

### 那么问题来了,我们应该把这个模板文件放在那里?

## 7>我们需要在app包里面新建应该templates文件夹

|  |
| --- |
|  |

## 8>然后在里面写我们的HTML模板文件

|  |
| --- |
|  |
|  |

## 9>此时还不行,我们需要注册这个template目录,否则无法访问,我们打开settings.py,找到TEMPLATES配置中的DIR一项,把templates目录的路径填写上(mac系统会自动寻找)

|  |
| --- |
|  |

### 也可以这么设置

|  |
| --- |
|  |

## 10>然后运行程序,效果如下

|  |
| --- |
|  |
|  |

## 11>其实我们可以在templates文件夹上面点击右键,然后选择Mark Directory as->template folder,这样子的话,即使我们不配置,django用找到这个文件夹

|  |
| --- |
|  |

## 12>然后,我们就可以在setting.py把设置删除

|  |
| --- |
|  |

## 13>然后重新运行项目,发现也是可以正常运行的

|  |
| --- |
|  |
|  |

## 13>然后我们可以配置应该静态文件夹,供html模板使用,我们一般把一些图片,css文件和js文件作为静态文件放到这个静态文件夹中,那么静态文件夹该如何设置?我们在app包里面新建应该static文件夹,然后在这里放置一张图片

|  |
| --- |
|  |

## 14>此时如果你在模板文件中直接使用,会报错,设置方法需要分2步

### 1>打开mysite包里面的settings.py文件,然后找到STATIC\_URL处,然后修改代码如下

|  |
| --- |
|  |

### 2>然后打开mysite包里面的urls.py,把这个静态文件夹设置一下

|  |
| --- |
|  |

## 15>然后我们在用户列表模板文件里面使用一个图片,写法如下

|  |
| --- |
|  |

### 然后我们刷新页面,图片就出来了

|  |
| --- |
|  |

## 16>现在我们尝试把样式文件分离出来,放置static/css/文件夹里面

|  |
| --- |
|  |

### 此时刷新页面,就会变为这样子,比较丑,因为我们移除了样式代码

|  |
| --- |
|  |

## 17>然后我们尝试导入这个css文件

|  |
| --- |
|  |

### 然后刷新页面,有恢复用来的样式

|  |
| --- |
|  |

## 18>然后我们在static文件夹里面新建一个js文件夹在里面编写一个utils.js文件,在里面定义一个获取任意多个数组的最大值的函数

|  |
| --- |
|  |

## 19>然后我们在userlist.html文件里面导入这个js文件然后在另外一个script并且里面使用这个函数,代码如下

|  |
| --- |
|  |

### 然后我们刷新页面,效果如下

|  |
| --- |
|  |

### 说明js文件的引入也是没有问题的.

## 20>在实际开发中我们还需要在static文件夹里面创建img和plugins文件夹,然后把图片放入img文件夹,plugins文件夹可以保存bootstrap等等的库

|  |
| --- |
|  |
|  |

## 21>当你把图片的位置修改了,你在使用图片的模板中也需要更新路径

|  |
| --- |
|  |

### 然后刷新页面,效果是一样的

|  |
| --- |
|  |

## 22>需要注意,在django中它不太推荐你使用html原生的引入方法,它有他特殊的语法,首先,需要使用一个指令来加载static文件夹,然后在下面有django语法来导入对应的资源

|  |
| --- |
|  |

## 23>然后图片和js也可以这么导入

|  |
| --- |
|  |

### 这样做的好处是,我们可以把静态文本放到任何位置,只需要配置一下即可,非常灵活,不用写死.可以避免不必要的麻烦

## 24>我们在register.html模板里面也用同样的方法来引入bootstrap.然后我们来给我们的sign up按钮添加boostrap样式

|  |
| --- |
|  |
|  |

## 25>然后我们进入reg路由,发现样式添加上了

|  |
| --- |
|  |

## 小结:

创建一个django项目其实需要使用2个命令:

用**django-admin startproject <projectname>** 来创建项目,然后进入项目根目录,

使用命令**python manage.py startapp <appname>**来创建应用程序.

在项目根目录里面有一个和项目同名的包.我们最常使用的就是settings.py和urls.py.

一个是用来做项目配置的,一个是用来做路由和视图函数匹配的.

然后我们在app包里面最常用的是views.py,你需要在这里编写视图函数实现功能.

然后在models的py里面编写数据库操作对应的类,实现orm功能,

在运行django项目之前,需要注册我们的app包.在views模块里面写函数,在urls模块里面匹配路由和对应的函数

运行项目的命令是python manage.py runserver,