

django框架简介

一、django框架是什么

Django是一个由Python编写的具有完整建站能力的开源Web框架。使用Django，只要很少的代码，Python的程序开发人员就可以轻松地完成一个正式网站所需要的大部分内容，并进一步开发出全功能的Web服务。

Django本身基于MVC模型，即Model（模型）+View（视图）+ Controller（控制器）设计模式，因此天然具有MVC的出色基因：开发快捷、部署方便、可重用性高、维护成本低等。Python加Django是快速开发、设计、部署网站的最佳组合。

学习网站如下：

- （1）刘江Django教程
- （2）Python Django 的学习路径
- （3）Django学习专题
- （4）谢育政的博客
- （5）基于 Python 3.5 + Django 2.0 开发的运维管理系统
- （6）自强学堂Django

二、django优缺点

优点：

功能完善、要素齐全：该有的、可以没有的都有，自带大量常用工具和框架，无须你自定义、组合、增删及修改。

完善的文档：经过十多年的发展和完善，Django有广泛的实践案例和完善的在线文档。开发者遇到问题时可以搜索在线文档寻求解决方案。

强大的数据库访问组件：Django的Model层自带数据库ORM组件，使得开发者无须学习其他数据库访问技术（SQL、pymysql、SQLAlchemy等）。

灵活的URI映射：Django使用正则表达式管理URI映射，灵活性高。新版的2.0，进一步提高了URI编写的优雅性。

丰富的Template模板语言：类似jinja模板语言，不但原生功能丰富，还可以自定义模板标签，并且与其ORM的用法非常相似。

自带后台管理系统admin： 只需要通过简单的几行配置和代码就可以实现一个完整的后台数据管理控制平台。

完整的错误信息提示：在开发调试过程中如果出现运行错误或者异常，Django可以提供非常完整的错误信息帮助定位问题。

缺点：

框架庞大，被认为不够精简、捆绑的内容太多

首先，对于新手，Django集成好的工具和部件，让你无须再费脑力去学习如何安装、调试、集成、兼容别的工具。Django帮你把它们都集成好了，而且保证兼容性、可用性和方便性，就好比联想一体机，开机即用，效率也高。

而一些如flask的框架，虽然精简，但是你要自己安装各种工具、ORM、插件等等，好比DIY电脑，在用之前，要知道买什么配件，怎么搭配，怎么组装，怎么配置效率才高，将新手的热情大量消耗在非关键性的内容上。

其次，对于老手，Django也是开放的，你完全可以关闭不必要的功能，忽略不使用的组件，或者自定义希望的组件，包括ORM和Template在内，都可以自由选择。

在异步通信方面略有欠缺

从本质上来说，Tornado在异步协程机制实现高并发的服务上要更强一点。Django在这方面有追赶的目标，但这不是说Django就差到不能用了，对么？

三、django原理

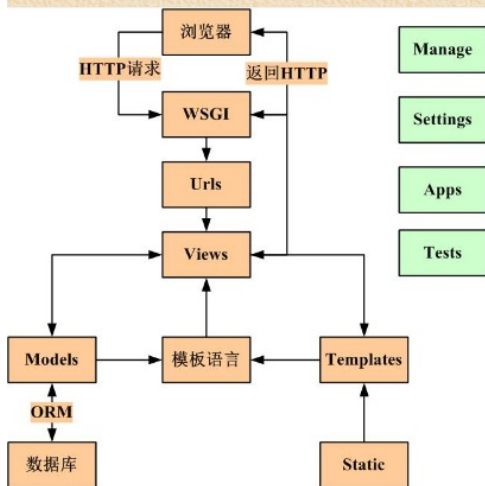
Django对传统的MVC设计模式进行了修改，将视图分成View模块和Template模块两部分，将动态的逻辑处理与静态的页面展现分离开。而Model采用了ORM技术，将关系型数据库表抽象成面向对象的Python类，将表操作转换成类操作，避免了复杂的SQL语句编写。MTV和MVC本质上是一样的。

模型(Model)：和MVC中的定义一样

模板(Template)：将数据与HTML语言结合起来的引擎

视图(View)：负责实际的业务逻辑实现

Django的MTV模型组织可参考下图所示：



四、django安装

1、通过pip进行安装

pip install django ----自动安装pypi提供的最新版本

pip install django==2.2 ----指定版本安装

2、验证安装

pip list 查看是否安装成功，安装成功如下图

```
C:\Users\zhaow>pip list
Package      Version
-----
certifi      2019.3.9
chardet      3.0.4
Click        7.0
Django       2.2.6
Faker        2.0.0
Flask        1.0.3
Flask-Cors   3.0.8
future       0.17.1
idna         2.8
```

五、django项目创建

1、命令行创建

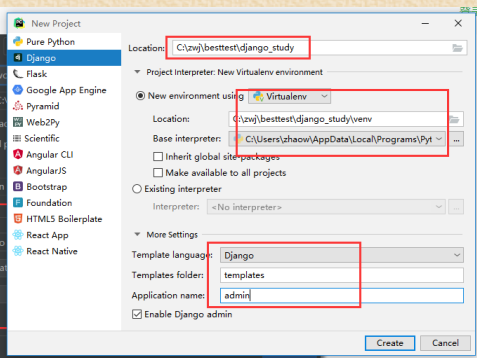
进入你想创建项目的目录

```
django-admin startproject dj_test #创建项目目录结构
```

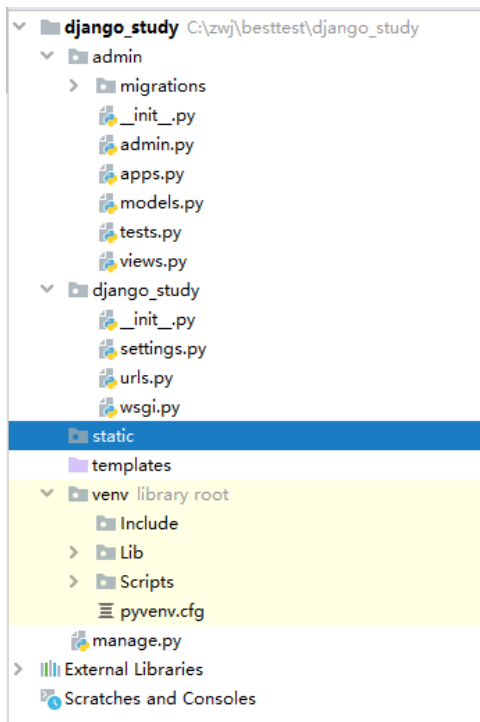
```
python manage.py startapp user #创建项目用户
```

```
python manage.py runserver # django会以127.0.0.1:8000这个默认配置启动开发服务器
```

2、pycharm创建Django项目



六、django项目目录简介



文件简介:

- **urls.py** : 网址入口, 关联到对应的views.py中的一个函数(或者generic类), 访问网址就对应一个函数。
- **views.py** : 处理用户发出的请求, 从urls.py中对应过来, 通过渲染templates中的网页可以将显示内容, 比如登陆后的用户名, 用户请求的数据, 输出到网页。
- **models.py** : 与数据库操作相关, 存入或读取数据时用到这个, 当然用不到数据库的时候 你可以不使用。
- **forms.py** : 表单, 用户在浏览器上输入数据提交, 对数据的验证工作以及输入框的生成等工作, 当然你也可以不使用。
- **templates 文件夹** : views.py 中的函数渲染templates中的Html模板, 得到动态内容的网页, 当然可以用缓存来提高速度。
- **static 文件夹** : views.py 中的函数渲染static中的静态图片, css和js等。
- **admin.py** : 后台, 可以用很少量的代码就拥有一个强大的后台。
- **settings.py** : Django 的设置, 配置文件, 比如 DEBUG 的开关, 静态文件的位置等