# Django 简介

### 基本介绍

Django 是一个由 Python 编写的一个开放源代码的 Web 应用框架。

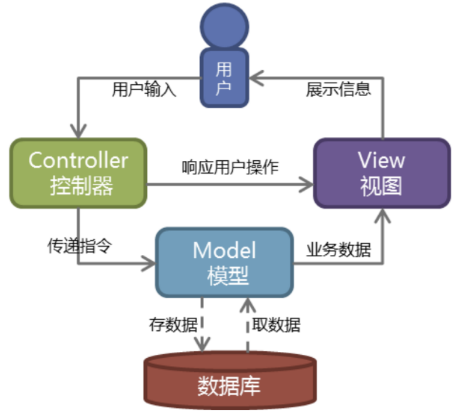
使用 Django，只要很少的代码，Python 的程序开发人员就可以轻松地完成一个正式网站所需要的大部分内容，并进一步开发出全功能的 Web 服务 Django 本身基于 MVC 模型，即 Model（模型）+ View（视图）+ Controller（控制器）设计模式，MVC 模式使后续对程序的修改和扩展简化，并且使程序某一部分的重复利用成为可能。

**MVC 模型**

MVC 模式（Model–view–controller）是软件工程中的一种软件架构模式，把软件系统分为三个基本部分：模型（Model）、视图（View）和控制器（Controller）。

MVC 以一种插件式的、松耦合的方式连接在一起。

* 模型（M）- 编写程序应有的功能，负责业务对象与数据库的映射(ORM)。
* 视图（V）- 图形界面，负责与用户的交互(页面)。
* 控制器（C）- 负责转发请求，对请求进行处理。



下面的图是Django的模式,最重要的部分应该是View这块

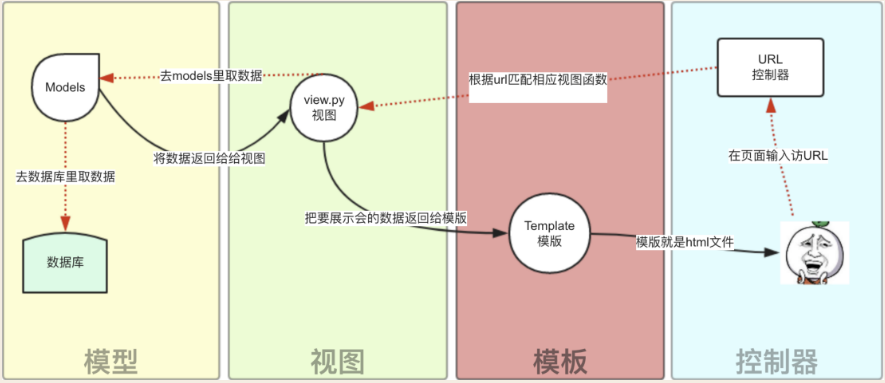
View负责业务逻辑，并在适当时候调用 Model和 Template。图中把数据传给了Template

然后用户输入也由View拿到

View和Model就是相当于和数据库打交道的感觉



除了以上三层之外，还需要一个 URL 分发器，(具体点就是urls.py这个文件)它的作用是将一个个 URL 的页面请求分发给不同的 View 处理，View 再调用相应的 Model 和 Template，MTV 的响应模式如下所示：



**解析：**

用户通过浏览器向我们的服务器发起一个请求(request)，这个请求会去访问视图函数：

* a.如果不涉及到数据调用，那么这个时候视图函数直接返回一个模板也就是一个网页给用户。
* b.如果涉及到数据调用，那么视图函数调用模型，模型去数据库查找数据，然后逐级返回。

视图函数把返回的数据填充到模板中空格中，最后返回网页给用户。

看一下目录

 最外层的:file: *mysite/* 根目录只是你项目的容器， Django 不关心它的名字，你可以将它重命名为任何你喜欢的名字。

 manage.py: 一个让你用各种方式管理 Django 项目的命令行工具。你可以阅读 [django-admin and manage.py](https://docs.djangoproject.com/zh-hans/2.1/ref/django-admin/) 获取所有 manage.py 的细节。

 里面一层的 mysite/ 目录包含你的项目，它是一个纯 Python 包。它的名字就是当你引用它内部任何东西时需要用到的 Python 包名。 (比如 mysite.urls).

 mysite/\_\_init\_\_.py：一个空文件，告诉 Python 这个目录应该被认为是一个 Python 包。如果你是 Python 初学者，阅读官方文档中的 [更多关于包的知识](https://docs.python.org/3/tutorial/modules.html#tut-packages)。

 mysite/settings.py：Django 项目的配置文件。如果你想知道这个文件是如何工作的，请查看 [Django settings](https://docs.djangoproject.com/zh-hans/2.1/topics/settings/) 了解细节。

 mysite/urls.py：Django 项目的 URL 声明，就像你网站的“目录”。阅读 [URL调度器](https://docs.djangoproject.com/zh-hans/2.1/topics/http/urls/) 文档来获取更多关于 URL 的内容。

 mysite/wsgi.py：作为你的项目的运行在 WSGI 兼容的Web服务器上的入口。阅读 [如何使用 WSGI 进行部署](https://docs.djangoproject.com/zh-hans/2.1/howto/deployment/wsgi/) 了解更多细