目录

[1. django（Python Web 框架） 1](#_Toc495920816)

[2. 框架介绍 1](#_Toc495920817)

[3. 架构设计 2](#_Toc495920818)

[4. 设计哲学 2](#_Toc495920819)

[5. 工作机制 2](#_Toc495920820)

[6. 部署 3](#_Toc495920821)

[7. 文档 3](#_Toc495920822)

[8. 学习网站 4](#_Toc495920823)

# django（Python Web 框架）

Django是一个开放源代码的Web应用框架，由Python写成。采用了MVC的框架模式，即模型M，视图V和控制器C。它最初是被开发来用于管理劳伦斯出版集团旗下的一些以新闻内容为主的网站的，即是CMS（内容管理系统）软件。并于2005年7月在BSD许可证下发布。这套框架是以比利时的吉普赛爵士吉他手Django Reinhardt来命名的。

# 框架介绍

Django 项目是一个python定制框架，它源自一个在线新闻 Web 站点，于 2005 年以开源的形式被释放出来。Django 框架的核心组件有：

1. 用于创建模型的[对象关系映射](https://baike.baidu.com/item/%E5%AF%B9%E8%B1%A1%E5%85%B3%E7%B3%BB%E6%98%A0%E5%B0%84)
2. 为最终用户设计的完美管理界面
3. 一流的 URL 设计
4. 设计者友好的模板语言
5. 缓存系统。

Django(发音：[`dʒæŋɡəʊ]) 是用[python](https://baike.baidu.com/item/python" \t "_blank)语言写的开源web开发框架(open source web framework)，它鼓励快速开发,并遵循[MVC](https://baike.baidu.com/item/MVC" \t "_blank)设计。Django遵守[BSD](https://baike.baidu.com/item/BSD" \t "_blank)版权，初次发布于2

005年7月, 并于2008年9月发布了第一个正式版本1.0 。

Django 根据比利时的爵士音乐家[Django Reinhardt](https://baike.baidu.com/item/Django%20Reinhardt" \t "_blank)命名，他是一个吉普赛人，主要以演奏吉它为主，还演奏过小提琴等。

由于Django在近年来的迅速发展，应用越来越广泛，被著名IT开发杂志SD Times评选为2013 SD Times 100，位列“API、库和框架”分类第6位，被认为是该领域的佼佼者[1]  。



# 架构设计

Django是一个基于[MVC](https://baike.baidu.com/item/MVC" \t "_blank)构造的框架。但是在Django中，控制器接受用户输入的部分由框架自行处理，所以 Django 里更关注的是模型（Model）、模板(Template)和视图（Views），称为 MTV模式。它们各自的职责如下：

|  |  |
| --- | --- |
| 层次 | 职责 |
| 模型（Model），即数据存取层 | 处理与数据相关的所有事务： 如何存取、如何验证有效性、包含哪些行为以及数据之间的关系等。 |
| 视图（View），即表现层 | 处理与表现相关的决定： 如何在页面或其他类型文档中进行显示。 |
| 模板(Template)，即业务逻辑层 | 存取模型及调取恰当模板的相关逻辑。模型与模板的桥梁。 |

从以上表述可以看出Django 视图不处理用户输入，而仅仅决定要展现哪些数据给用户，而Django 模板 仅仅决定如何展现Django视图指定的数据。或者说, Django将MVC中的视图进一步分解为 Django视图 和 Django模板两个部分，分别决定 “展现哪些数据” 和 “如何展现”，使得Django的模板可以根据需要随时替换，而不仅仅限制于内置的模板。

至于MVC控制器部分，由Django框架的URLconf来实现。URLconf机制是使用正则表达式匹配URL，然后调用合适的Python函数。URLconf对于URL的规则没有任何限制，你完全可以设计成任意的URL风格，不管是传统的，RESTful的，或者是另类的。框架把控制层给封装了，无非与数据交互这层都是数据库表的读,写,删除,更新的操作.在写程序的时候，只要调用相应的方法就行了，感觉很方便。程序员把控制层东西交给Django自动完成了。 只需要编写非常少的代码完成很多的事情。所以，它比MVC框架考虑的问题要深一步，因为我们程序员大都在写控制层的程序。现在这个工作交给了框架，仅需写很少的调用代码，大大提高了工作效率。

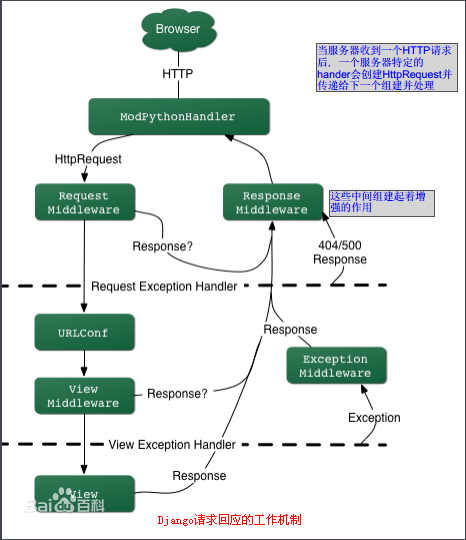
# 设计哲学

Django的主要目的是简便、快速的开发数据库驱动的网站。它强调代码复用,多个组件可以很方便的以“插件”形式服务于整个框架，Django有许多功能强大的第三方[插件](https://baike.baidu.com/item/%E6%8F%92%E4%BB%B6" \t "_blank)，你甚至可以很方便的开发出自己的工具包。这使得Django具有很强的可扩展性。它还强调快速开发和[DRY](https://baike.baidu.com/item/DRY" \t "_blank)(Do Not Repeat Yourself)原则。

Django基于[MVC](https://baike.baidu.com/item/MVC" \t "_blank)的设计十分优美：

* 对象关系映射 ([ORM](https://baike.baidu.com/item/ORM" \t "_blank),object-relational mapping)：以Python类形式定义你的数据模型，ORM将模型与关系数据库连接起来，你将得到一个非常容易使用的数据库API，同时你也可以在Django中使用原始的SQL语句。
* URL 分派：使用[正则表达式](https://baike.baidu.com/item/%E6%AD%A3%E5%88%99%E8%A1%A8%E8%BE%BE%E5%BC%8F" \t "_blank)匹配URL，你可以设计任意的URL，没有框架的特定限定。像你喜欢的一样灵活
* 模版系统：使用Django强大而可扩展的模板语言，可以分隔设计、内容和Python代码。并且具有可继承性。
* 表单处理：你可以方便的生成各种表单模型，实现表单的有效性检验。可以方便的从你定义的模型实例生成相应的表单。
* Cache系统：可以挂在内存缓冲或其它的框架实现超级缓冲 －－ 实现你所需要的粒度。
* 会话([session](https://baike.baidu.com/item/session))，用户登录与权限检查，快速开发用户会话功能。
* 国际化：内置国际化系统，方便开发出多种语言的网站。
* 自动化的管理界面：不需要你花大量的工作来创建人员管理和更新内容。Django自带一个ADMIN site,类似于[内容管理系统](https://baike.baidu.com/item/%E5%86%85%E5%AE%B9%E7%AE%A1%E7%90%86%E7%B3%BB%E7%BB%9F" \t "_blank)

# 工作机制

1.用manage .py runserver 启动Django服务器时就载入了在同一目录下的settings .py。该文件包含了项目中的配置信息，如前面讲的URLConf等，其中最重要的配置就是ROOT\_URLCONF，它告诉Django哪个Python模块应该用作本站的URLConf，默认的是urls .py。

2.当访问url的时候，Django会根据ROOT\_URLCONF的设置来装载URLConf。

3.然后按顺序逐个匹配URLConf里的URLpatterns。如果找到则会调用相关联的视图函数，并把HttpRequest对象作为第一个参数(通常是request)

4.最后该view函数负责返回一个HttpResponse对象。

# 部署

Django可以运行在[Apache](https://baike.baidu.com/item/Apache),[Nginx](https://baike.baidu.com/item/Nginx)上，也可以运行在支持[WSGI](https://baike.baidu.com/item/WSGI)，[FastCGI](https://baike.baidu.com/item/FastCGI" \t "_blank)的服务器上。支持多种数据库，已经支持[Postgresql](https://baike.baidu.com/item/Postgresql),MySql, Sqlite3,[Oracle](https://baike.baidu.com/item/Oracle)。[Google App Engine](https://baike.baidu.com/item/Google%20App%20Engine" \t "_blank)也支持Django的某些部分，国内支持的平台有（SAE）Sina App Engine，（BAE）百度应用引擎。

# 文档

Django1.0 建立了强大完整的文档体系，,涵盖了Django的方方面面,并且适合各种水平的读者和开发者. 其中还包含若干个简单的示例,你可以跟随他们一步步体验Django的优美。

截至目前，Django最新版本是1.7.7，1.7系列版本的文档非常的全面，主要由以下几部分组成：

1. First steps，提供一个快速起步的教程，可以很快开始使用Django
2. The model layer，介绍Django的抽象模型层
3. The view layer，介绍Django的视图层
4. The template layer，介绍Django的模板层
5. Forms，介绍Django提供的一系列用于帮助使用表单的工具
6. The development process，介绍Django提供的一系列用于开发和测试的工具
7. The admin，介绍Django提供的用于站点、内容管理的工具
8. Security，介绍Django提供的用于站点安全控制的工具
9. Internationalization and localization，介绍Django提供的用于全球化和本地化的工具
10. Performance and optimization，介绍Django提供的用于性能和优化的方面的工具以及建议
11. Python compatibility，介绍Django在不同Python环境下的兼容性（Jython和Python 3）
12. Geographic framework，介绍Django提供的地理位置相关的工具
13. Common web application tools，介绍Django提供的一些常用的Web应用程序工具（比如RSS等）
14. Other core functionalities，介绍Django提供的其它重要工具（比如跳转、路由等等）
15. The django open source project，介绍Django作为开源项目的一些内容，比如设计哲学，如何参与开发等等。

# 学习网站

Django官方网站[1]

Django documentation[2]

The Django Book[3]

Django 基础教程[4]