

利用Django 2.0构建强大、可靠的Web应用程序

Django

项目实例精解 (第2版)

[美] 安东尼奥·米勒 著 李 伟 译

Django 2 by Example



清华大学出版社

Django 项目实例精解

(第2版)

[美] 安东尼奥·米勒 著

李伟 译

清华大学出版社

北京

内 容 简 介

本书详细阐述了与 Django 开发相关的基本解决方案, 主要包括构建博客应用程序、利用高级特性完善博客程序、扩展博客应用程序、构建社交型网站、共享网站中的内容、跟踪用户活动、构建在线商店、管理支付操作和订单、扩展在线商店应用程序、打造网络教学平台、显示和缓存内容、构建 API、部署项目等内容。此外, 本书还提供了相应的示例、代码, 以帮助读者进一步理解相关方案的实现过程。

本书适合作为高等院校计算机及相关专业的教材和教学参考书, 也可作为相关开发人员的自学教材和参考手册。

Copyright © Packt Publishing 2018. First published in the English language under the title *Django2 by Example*.

Simplified Chinese-language edition © 2019 by Tsinghua University Press. All rights reserved.

本书中文简体字版由 Packt Publishing 授权清华大学出版社独家出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

北京市版权局著作权合同登记号 图字: 01-2018-4940

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目(CIP)数据

Django 项目实例精解: 第2版/ (美) 安东尼奥·米勒 (Antonio Mele) 著; 李伟译. —北京: 清华大学出版社, 2019

书名原文: Django 2 by Example
ISBN 978-7-302-52655-1

I. ①D… II. ①安… ②李… III. ①软件工具-程序设计 IV. ①TP311.561

中国版本图书馆 CIP 数据核字 (2019) 第 047103 号

责任编辑: 贾小红

封面设计: 刘超

版式设计: 魏远

责任校对: 马子杰

责任印制: 丛怀宇

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦 A 座 邮 编: 100084

社总机: 010-62770175 邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印装者: 三河市君旺印务有限公司

经 销: 全国新华书店

开 本: 185mm×230mm

印 张: 28

字 数: 561 千字

版 次: 2019 年 3 月第 1 版

印 次: 2019 年 3 月第 1 次印刷

定 价: 139.00 元

产品编号: 080276-01

译者序

Web 开发是 Python 语言应用领域的重要部分。Python 作为当前最火爆、最热门，也是最主要的 Web 开发语言之一，在其发展过程中出现了数十种 Web 框架。其中，Django 是一个功能强大的 Python Web 框架，支持快速开发过程以及简洁、实用的设计方案。

由于 Django 在近年来的迅速发展，其应用越来越广泛，同时也被认为是该领域的佼佼者。本书在 Django 2.0 的基础上引领读者构建真实的 Web 应用程序，其中涉及 Redis 和 Celery 等技术、开发插件式 Django 应用程序、优化代码并使用缓存框架、向 Django 项目中加入国际化特性、利用 JavaScript 和 AJAX 丰富用户体验、添加社交功能，以及针对应用程序构建 RESTful API，而这一切均通过颇具研究价值的项目实例的方式予以展现。

在本书的翻译过程中，除李伟外，张博、刘璋、王烈征、刘晓雪、刘祎、张华臻等人也参与了部分翻译工作，在此一并表示感谢。

由于译者水平有限，难免有疏漏和不妥之处，恳请广大读者批评指正。

译者

前 言

Django 是一个功能强大的 Python Web 框架，支持快速开发过程以及简洁、实用的设计方案。无论是对于初学者还是专家级程序员，这一特点颇具吸引力。

本书将引领读者学习专业 Web 应用程序的开发流程。除了框架知识之外，本书还将讲解如何将其他较为流行的技术整合至 Django 项目中。

本书将讨论真实应用程序的构建过程、常见问题的处理，并逐步实现多种最佳实践方案。

在阅读完本书后，读者将能够理解 Django 的工作方式，以及如何打造具有实用性的高级 Web 应用程序。

适用读者

本书是针对具备一定的 Python 知识，同时希望以一种实用的方式学习 Django 的读者而准备的。或许，Django 对于读者来说是一项全新的事物；抑或，读者对此稍有了解且希望进一步学习 Django。通过打造实用的开发项目，本书可帮助读者掌握大部分架构知识。另外，本书要求读者对某些编程概念有所了解，同时具备一些 HTML 和 JavaScript 方面的知识。

本书内容

第 1 章通过编写博客应用程序向读者介绍框架知识。其间，我们将构建基本的博客模型、视图、模板以及 URL 以显示博客内容。另外，读者还将学习如何利用 Django ORM 构建 QuerySets，并配置 Django 管理网站。

第 2 章将讨论如何处理表单问题、利用 Django 发送邮件以及第三方应用程序的整合操作。读者将尝试实现博客的评论系统，并通过电子邮件共享帖子内容。此外，本章还将讨论标签系统的构建处理过程。

第3章将介绍如何创建自定义模板标签和过滤器。除此之外，本章还将展示如何使用网站地图框架，并对帖子构建博客订阅功能。最后，通过 PostgreSQL 的全文本搜索功能构建搜索引擎，我们将完善博客应用程序。

第4章讨论如何构建社交网站，并使用 Django 身份验证框架构建用户的账户视图。另外，本章还将了解如何使用社交网络创建自定义用户配置文件模型，并将身份验证机制应用到项目中去。

第5章将讨论如何将社交应用程序转换为图像书签站点。其中，我们将针对模型定义多对多的关系，在 JavaScript 中创建一个 AJAX 书签，并将其集成到项目中。本章还进一步展示了如何生成图像缩略图和为视图创建自定义装饰器。

第6章介绍如何针对用户构建跟踪系统，并通过创建用户活动流应用程序完成图像书签站点，以及如何优化 QuerySets 并与信号协同工作。同时，本章还将 Redis 整合至项目中，以对图像视图进行计数。

第7章将讨论如何构建一个在线商店，其中包括目录模型、基于 Django 会话的购物车（并对此设置上下文处理器），以及通过 Celery 向用户发送异步通知。

第8章讨论如何将支付网关整合至在线商店中。除此之外，还将定制管理站点以将订单导出到 CSV 文件中，并动态生成 PDF 发票。

第9章将讨论如何创建优惠券系统并使用折扣订单。同时，本章还展示了如何在项目中实现国际化机制以及如何转换模型。此外，还将使用 Redis 构建一个产品推荐引擎。

第10章将设计一个电子教育平台，并向项目中添加某些固件、使用模型继承机制、设置自定义模型字段、使用类视图，以及管理分组和权限。此外，我们还将打造一个内容管理系统并处理表单集。

第11章将尝试构建一个学生注册系统，并管理学生的课程注册行为。该系统将显示不同的课程内容，同时还将学习如何使用缓存框架。

第12章将采用 Django REST 框架，进而针对项目构建 RESTful API。

第13章讨论如何通过 uWSGI 和 NGINX 设置产品环境，并利用 SSL 解决安全问题。此外，本章还解释了如何构建自定义中间件以及自定义管理命令。

背景知识

当阅读本书时，建议读者具备一定的 Python 知识，并熟悉 HTML 以及 JavaScript。另外，在阅读本书之前，建议读者阅读 Django 官方文档的 1~3 部分，对应网址为 <https://docs.djangoproject.com/>。

djangoproject.com/en/2.0/intro/tutorial01/。

资源下载

读者可访问 <http://www.packtpub.com> 并通过个人账户下载示例代码文件。在 <http://www.packtpub.com/support> 网站注册成功后，我们将以电子邮件的方式将相关文件发与读者。

读者可根据下列步骤下载代码文件：

- (1) 登录 www.packtpub.com 并在网站注册。
- (2) 选择 Support 选项卡。
- (3) 单击 CODE DOWNLOADS & ERRATA。
- (4) 在 Search 文本框中输入书名并执行后续命令。

当文件下载完毕后，确保使用下列最新版本软件解压文件夹：

- ☐ Windows 系统下的 WinRAR/7-Zip。
- ☐ Mac 系统下的 Zipeg/iZip/UnRarX。
- ☐ Linux 系统下的 7-Zip/PeaZip。

另外，读者还可访问 GitHub 获取本书的代码包，对应网址为 <https://github.com/PacktPublishing/Django-2-by-Example>。

此外，读者还可访问 <https://github.com/PacktPublishing/> 网站，以了解丰富的代码和视频资源。

本书约定

代码块则通过下列方式设置：

```
from django.contrib import admin
from .models import Post

admin.site.register(Post)
```

代码中的重点内容则采用黑体表示：

```
INSTALLED_APPS = [
    'django.contrib.admin',
```

```
'django.contrib.auth',  
'django.contrib.contenttypes',  
'django.contrib.sessions',  
'django.contrib.messages',  
'django.contrib.staticfiles',  
'blog.apps.BlogConfig',
```

命令行输入或输出如下所示：

```
$ python manage.py startapp blog
```



图标表示较为重要的说明事项。



图标则表示提示信息和操作技巧。

读者反馈和客户支持

欢迎读者对本书提出建议或意见并予以反馈。

对此，读者可向 feedback@packtpub.com 发送邮件，并以书名作为邮件标题。若读者对本书有任何疑问，均可发送邮件至 questions@packtpub.com，我们将竭诚为您服务。

勘误表

尽管我们希望做到尽善尽美，但错误依然在所难免。如果读者发现谬误之处，无论是文字错误抑或是代码错误，还望不吝赐教。对此，读者可访问 <http://www.packtpub.com/submit-errata>，选取对应书籍，输入并提交相关问题的详细内容。

版权须知

一直以来，互联网上的版权问题从未间断，Packt 出版社对此类问题异常重视。若读者在互联网上发现本书任意形式的副本，请告知我们网络地址或网站名称，我们将对此予以处理。关于盗版问题，读者可发送邮件至 copyright@packtpub.com。

若读者针对某项技术具有专家级的见解，抑或计划撰写书籍或完善某部著作的出版工作，则可访问 www.packtpub.com/authors。

问题解答

若读者对本书有任何疑问，均可发送邮件至 questions@packtpub.com，我们将竭诚为您服务。

目 录

第 1 章 构建博客应用程序	1
1.1 安装 Django	1
1.1.1 创建隔离的 Python 环境	2
1.1.2 利用 pip 安装 Django	3
1.2 创建第一个项目	3
1.2.1 运行开发服务器	5
1.2.2 项目设置	6
1.2.3 项目和应用程序	7
1.2.4 创建应用程序	8
1.3 设计博客数据方案	8
1.3.1 激活应用程序	10
1.3.2 设置并使用迁移方案	11
1.4 针对模型创建管理站点	12
1.4.1 创建超级用户	13
1.4.2 Django 管理站点	13
1.4.3 向管理站点中添加模型	14
1.4.4 定制模型的显示方式	16
1.5 与 QuerySet 和管理器协同工作	18
1.5.1 创建对象	18
1.5.2 更新对象	19
1.5.3 获取对象	20
1.5.4 删除对象	21
1.5.5 评估 QuerySet	21
1.5.6 创建模型管理器	22
1.6 构建列表和详细视图	22
1.6.1 生成列表和视图	23
1.6.2 向视图添加 URL 路径	24
1.6.3 模型的标准 URL	25

1.7	创建视图模板	26
1.8	添加分页机制	29
1.9	使用基于类的视图	31
1.10	本章小结	33
第2章	利用高级特性完善博客程序	35
2.1	通过电子邮件共享帖子	35
2.1.1	利用 Django 创建表单	35
2.1.2	处理视图中的表单	36
2.1.3	利用 Django 发送邮件	38
2.1.4	显示模板中的视图	40
2.2	构建评论系统	44
2.2.1	创建模型中的表单	46
2.2.2	处理视图中的 ModelForms	46
2.2.3	向帖子详细模板中添加评论	48
2.3	添加标签功能	51
2.4	根据相似性检索帖子	57
2.5	本章小结	59
第3章	扩展博客应用程序	61
3.1	创建自定义模板标签和过滤器	61
3.1.1	创建自定义模板标签	61
3.1.2	创建自定义模板过滤器	66
3.2	向站点添加网站地图	68
3.3	创建帖子提要	72
3.4	向博客中添加全文本搜索功能	74
3.4.1	安装 PostgreSQL	74
3.4.2	简单的查询操作	76
3.4.3	多字段搜索	76
3.4.4	构建搜索视图	77
3.4.5	对结果提取和排名	79
3.4.6	加权查询	80
3.4.7	利用三元相似性进行搜索	81
3.4.8	其他全文本搜索引擎	82

3.5 本章小结	82
第4章 构建社交型网站	83
4.1 设计社交型网站	83
4.2 使用 Django 验证框架	84
4.2.1 构建登录视图	85
4.2.2 使用 Django 验证视图	90
4.2.3 登录和注销视图	91
4.2.4 修改密码视图	96
4.2.5 重置密码视图	98
4.3 用户注册和用户配置	103
4.3.1 用户注册	103
4.3.2 扩展用户模型	107
4.3.3 使用消息框架	112
4.4 构建自定义验证后端	114
4.5 向站点中添加社交网站验证	116
4.5.1 基于 Facebook 的验证	118
4.5.2 基于 Twitter 的验证	123
4.5.3 基于 Google 的验证	125
4.6 本章小结	128
第5章 共享网站中的内容	129
5.1 构建图像书签网站	129
5.1.1 构建图像模型	130
5.1.2 生成多对多关系	131
5.1.3 在管理站点中注册图像模型	132
5.2 发布其他站点中的内容	132
5.2.1 清空表单字段	133
5.2.2 覆写 ModelForm 的 save() 方法	134
5.2.3 利用 jQuery 构建书签工具	138
5.3 创建图像的细节视图	147
5.4 利用 sorl-thumbnail 生成图像缩略图	149
5.5 利用 jQuery 添加 AJAX 操作	150
5.5.1 加载 jQuery	151

5.5.2	AJAX 请求中的跨站点请求伪造	152
5.5.3	利用 jQuery 执行 AJAX 请求	153
5.6	针对视图创建自定义装饰器	156
5.7	向列表视图中添加 AJAX 分页机制	158
5.8	本章小结	162
第 6 章	跟踪用户活动	163
6.1	构建关注系统	163
6.1.1	利用中间模型创建多对多关系	163
6.1.2	针对用户配置创建列表和详细视图	166
6.1.3	构建 AJAX 视图以关注用户	171
6.2	构建通用活动流应用程序	173
6.2.1	使用 contenttypes 框架	174
6.2.2	向模型中添加通用关系	175
6.2.3	避免活动流中的重复内容	178
6.2.4	向活动流中添加用户活动	179
6.2.5	显示活动流	180
6.2.6	优化涉及关系对象的 QuerySet	181
6.2.7	针对操作活动创建模板	182
6.3	利用信号实现反规范化计数	184
6.3.1	与信号协同工作	184
6.3.2	应用程序配置类	186
6.4	利用 Redis 存储数据项视图	188
6.4.1	安装 Redis	188
6.4.2	结合 Python 使用 Redis	190
6.4.3	将数据视图存储于 Redis 中	191
6.4.4	将排名结果存储于数据库中	192
6.4.5	Redis 特性	195
6.5	本章小结	195
第 7 章	构建在线商店	197
7.1	创建在线商店项目	197
7.1.1	创建商品目录模型	198
7.1.2	注册站点上的目录模型	200

7.1.3	构建目录视图	201
7.1.4	生成目录模板	204
7.2	创建购物车	208
7.2.1	使用 Django 会话	208
7.2.2	会话设置	209
7.2.3	会话过期	210
7.2.4	将购物车存储于会话中	210
7.2.5	创建购物车视图	214
7.2.6	针对购物车创建上下文处理器	221
7.3	注册客户订单	223
7.3.1	创建订单模型	224
7.3.2	在管理站点中包含订单模型	225
7.3.3	创建自定义订单	227
7.4	利用 Celery 启动异步任务	231
7.4.1	安装 Celery	231
7.4.2	安装 RabbitMQ	231
7.4.3	向项目中添加 Celery	232
7.4.4	向应用程序中添加异步任务	233
7.4.5	监视 Celery	235
7.5	本章小结	235
第 8 章	管理支付操作和订单	237
8.1	整合支付网关	237
8.1.1	创建 Braintree 沙箱账号	237
8.1.2	安装 Braintree Python 模块	239
8.1.3	集成支付网关	239
8.1.4	支付的测试操作	247
8.1.5	注意事项	249
8.2	将订单导出为 CSV 文件	249
8.3	利用自定义视图扩展管理站点	252
8.4	动态生成 PDF 发票	257
8.4.1	安装 WeasyPrint	257
8.4.2	创建 PDF 模板	257

8.4.3	显示 PDF 文件	258
8.4.4	通过电子邮件发送 PDF 文件	262
8.5	本章小结	263
第 9 章	扩展在线商店应用程序	265
9.1	创建优惠券系统	265
9.1.1	构建优惠券模型	266
9.1.2	在购物车中使用优惠券	268
9.1.3	在订单中使用优惠券	274
9.2	添加国际化和本地化机制	276
9.2.1	Django 的国际化处理	276
9.2.2	项目的国际化	278
9.2.3	翻译 Python 代码	279
9.2.4	翻译模板	285
9.2.5	使用 Rosetta	289
9.2.6	模糊翻译	291
9.2.7	国际化操作的 URL 路径	292
9.2.8	切换语言	294
9.2.9	利用 django-parler 翻译模块	295
9.2.10	本地化格式	303
9.2.11	使用 django-localflavor 验证表单字段	304
9.3	构建推荐引擎	305
9.4	本章小结	313
第 10 章	打造网络教学平台	315
10.1	设置网络教学项目	315
10.2	构建课程模型	316
10.2.1	在管理站点中注册模型	318
10.2.2	使用固定文件提供模型的初始数据	319
10.3	创建包含多样化内容的模型	321
10.3.1	使用模型继承机制	322
10.3.2	创建内容模型	324
10.3.3	创建自定义模型字段	326
10.3.4	向模块和内容对象中添加顺序机制	328

10.4 创建 CMS	332
10.4.1 添加验证系统	333
10.4.2 创建验证模板	333
10.4.3 设置基于类的模板	336
10.4.4 针对基于类的视图使用混合类	337
10.4.5 分组和权限	339
10.5 管理课程模块和内容	346
10.5.1 针对课程模块使用表单	346
10.5.2 向课程模块中添加内容	350
10.5.3 管理模块和内容	355
10.5.4 对模块和内容重排序	360
10.6 本章小结	363
第 11 章 显示和缓存内容	365
11.1 显示课程	365
11.2 添加学生注册机制	370
11.2.1 创建学生注册视图	370
11.2.2 注册课程	373
11.3 访问课程内容	376
11.4 使用缓存框架	382
11.4.1 有效的缓存后端	383
11.4.2 安装 Memcached	383
11.4.3 缓存设置	384
11.4.4 向项目中添加 Memcached	385
11.4.5 缓存级别	386
11.4.6 使用底层缓存 API	386
11.4.7 缓存模板片段	389
11.4.8 缓存视图	390
11.5 本章小结	392
第 12 章 构建 API	393
12.1 构建 RESTful API	393
12.1.1 安装 Django REST 框架	393
12.1.2 定义序列化器	394

12.1.3	理解解析器和渲染器	395
12.1.4	构建列表和详细视图	396
12.1.5	嵌套序列化器	399
12.1.6	构建自定义视图	400
12.1.7	处理授权问题	401
12.1.8	向视图中添加授权机制	402
12.1.9	创建视图集合路由器	404
12.1.10	向视图集添加附加操作	405
12.1.11	创建自定义授权	406
12.1.12	序列化课程内容	406
12.2	本章小结	409
第 13 章	部署项目	411
13.1	生成产品环境	411
13.1.1	针对多种环境管理设置内容	411
13.1.2	使用 PostgreSQL	414
13.1.3	项目检查	414
13.1.4	通过 WSGI 为 Django 提供服务	415
13.1.5	安装 uWSGI	415
13.1.6	配置 uWSGI	415
13.1.7	安装 NGINX	417
13.1.8	产品环境	418
13.1.9	配置 NGINX	418
13.1.10	向静态和媒体数据集提供服务	420
13.1.11	基于 SSL 的安全连接	421
13.2	创建自定义中间件	423
13.2.1	创建子域名中间件	424
13.2.2	利用 NGINX 向多个子域名提供服务	426
13.3	实现自定义管理命令	426
13.4	本章小结	429