

Flask Web开发实战



[Flask Web开发实战 下载链接1](#)

著者:李辉

出版者:机械工业出版社

出版时间:2018-8-1

装帧:平装

isbn:9787111606598

这是一本面向Python程序员的，全面介绍Python Web框架Flask的书。关于本书的详细介绍、相关资源等更多信息可以访问本书的官方

主页<http://helloflask.com/book>了解。

- 国内首本Flask著作，在内容上涵盖完整的Flask Web开发学习路径，在实践上包含完整的Flask Web程序开发流程。同时兼容Python2.7和Python3.6。
- 内容最新，基于Flask 1.0.2版本，所有的Python库、前端框架和JavaScript库均使用最新版本。
- 内容全面，除了基础知识，还包含Web API、全文搜索、国际化和本地化、第三方登录、AJAX、WebSocket、源码分析、扩展开发等实用内容。
- 实战性强，包含8个示例程序，5个实战项目和1个Flask扩展。实战项目包括留言板、个人博客、图片社交网站、待办事项程序和聊天室，分别对应不同的知识点，难度递增。
- 5个实战项目的Git标签经过精心设置，每个标签都对应新增的某个程序功能。
- 全书所有章节均经过精心设计和编排，力求满足不同水平读者的需要，并且让读者可以循序渐进的掌握Flask开发知识和技巧。
- 通过丰富的图表和代码示例讲解复杂知识点，尽量做到通俗易懂。

本书共16章，分为三部分。

第一部分 基础篇（第1~6章）

通过大量的程序实例详细介绍了Flask的所有基础知识，同时在每章的后面又提供了一些进阶技巧，供进阶读者学习。

第二部分 实战篇（第7~11章）

通过5个真实的项目案例来串接和阐释不同的知识点，难度逐渐递增。这5个案例分别为：留言板SayHello、个人博客Bluelog、图片社交网站Albumy、待办事项程序Todoism和聊天室CatChat。

第三部分 进阶篇（第12~16章）

首先介绍了Flask程序的部署流程：测试、性能优化和部署上线；然后通过一个真实的实例讲解了Flask扩展的开发，最后以源代码为切入点深入剖析了Flask的实现原理与主要工作机制。

作者介绍：

李辉（Grey Li），Flask官方开发团队（Pallets Team）成员，知乎专栏“Hello, Flask!”作者，HelloFlask.com网站创建者。对Flask有非常深入的研究，熟悉它的源代码，有丰富的项目经验。

他热爱分享和创造，曾为Flask等多个开源项目贡献过代码，你可以在GitHub（github.com/greyli）上看到他的开源活动。在他的个人网站（greyli.com）上，你可以了解到关于他的更多信息。

目录: 第1章 初识Flask

- 1.1 搭建开发环境
 - 1.1.1 Pipenv工作流
 - 1.1.2 安装Flask
 - 1.1.3 集成开发环境
- 1.2 Hello, Flask!
 - 1.2.1 创建程序实例
 - 1.2.2 注册路由
- 1.3 启动开发服务器
 - 1.3.1 Run, Flask, Run!
 - 1.3.2 更多的启动选项
 - 1.3.3 设置运行环境
- 1.4 Python Shell
- 1.5 Flask扩展
- 1.6 项目配置
- 1.7 URL与端点
- 1.8 Flask命令
- 1.9 模板与静态文件
- 1.10 Flask与MVC架构
- 1.11 本章小结

第2章 Flask与HTTP

- 2.1 请求响应循环
- 2.2 HTTP请求
 - 2.2.1 请求报文
 - 2.2.2 Request对象
 - 2.2.3 在Flask中处理请求
 - 2.2.4 请求钩子
- 2.3 HTTP响应
 - 2.3.1 响应报文
 - 2.3.2 在Flask中生成响应
 - 2.3.3 响应格式
 - 2.3.4 来一块Cookie
 - 2.3.5 session: 安全的Cookie
- 2.4 Flask上下文
 - 2.4.1 上下文全局变量
 - 2.4.2 激活上下文
 - 2.4.3 上下文钩子
- 2.5 HTTP进阶实践
 - 2.5.1 重定向回上一个页面
 - 2.5.2 使用AJAX技术发送异步请求
 - 2.5.3 HTTP服务器端推送
 - 2.5.4 Web安全防范
- 2.6 本章小结

第3章 模板

- 3.1 模板基本用法
 - 3.1.1 创建模板
 - 3.1.2 模板语法
 - 3.1.3 渲染模板
- 3.2 模板辅助工具
 - 3.2.1 上下文

- 3.2.2 全局对象
- 3.2.3 过滤器
- 3.2.4 测试器
- 3.2.5 模板环境对象
- 3.3 模板结构组织
- 3.3.1 局部模板
- 3.3.2 宏
- 3.3.3 模板继承
- 3.4 模板进阶实践
- 3.4.1 空白控制
- 3.4.2 加载静态文件
- 3.4.3 消息闪现
- 3.4.4 自定义错误页面
- 3.4.5 JavaScript和CSS中的Jinja2
- 3.5 本章小结

第4章 表单

- 4.1 HTML表单
- 4.2 使用Flask-WTF处理表单
- 4.2.1 定义WTForms表单类
- 4.2.2 输出HTML代码
- 4.2.3 在模板中渲染表单
- 4.3 处理表单数据
- 4.3.1 提交表单
- 4.3.2 验证表单数据
- 4.3.3 在模板中渲染错误消息
- 4.4 表单进阶实践
- 4.4.1 设置错误消息语言
- 4.4.2 使用宏渲染表单
- 4.4.3 自定义验证器
- 4.4.4 文件上传
- 4.4.5 使用Flask-CKEditor集成富文本编辑器
- 4.4.6 单个表单多个提交按钮
- 4.4.7 单个页面多个表单
- 4.5 本章小结

第5章 数据库

- 5.1 数据库的分类
 - 5.1.1 SQL
 - 5.1.2 NoSQL
 - 5.1.3 如何选择?
- 5.2 ORM魔法
- 5.3 使用Flask-SQLAlchemy管理数据库
 - 5.3.1 连接数据库服务器
 - 5.3.2 定义数据库模型
 - 5.3.3 创建数据库和表
- 5.4 数据库操作
 - 5.4.1 CRUD
 - 5.4.2 在视图函数里操作数据库
- 5.5 定义关系
 - 5.5.1 配置Python Shell上下文
 - 5.5.2 一对多
 - 5.5.3 多对一
 - 5.5.4 一对一

- 5.5.5 多对多
- 5.6 更新数据库表
 - 5.6.1 重新生成表
 - 5.6.2 使用Flask-Migrate迁移数据库
 - 5.6.3 开发时是否需要迁移？
- 5.7 数据库进阶实践
 - 5.7.1 级联操作
 - 5.7.2 事件监听
- 5.8 本章小结

- 第6章 电子邮件**
 - 6.1 使用Flask-Mail发送电子邮件
 - 6.1.1 配置Flask-Mail
 - 6.1.2 构建邮件数据
 - 6.1.3 发送邮件
 - 6.2 使用事务邮件服务SendGrid
 - 6.2.1 注册SendGrid
 - 6.2.2 SendGrid SMTP转发
 - 6.2.3 SendGrid Web API转发
 - 6.3 电子邮件进阶实践
 - 6.3.1 提供HTML正文
 - 6.3.2 使用Jinja2模板组织邮件正文
 - 6.3.3 异步发送邮件
 - 6.4 本章小结

第二部分 实战篇

- 第7章 留言板**
 - 7.1 使用包组织代码
 - 7.1.1 配置文件
 - 7.1.2 创建程序实例
 - 7.2 Web程序开发流程
 - 7.2.1 程序功能设计
 - 7.2.2 前端页面开发
 - 7.2.3 后端程序开发
 - 7.3 使用Bootstrap-Flask简化页面编写
 - 7.3.1 加载资源文件
 - 7.3.2 快捷渲染表单
 - 7.4 使用Flask-Moment本地化日期和时间
 - 7.4.1 本地化前的准备
 - 7.4.2 使用Flask-Moment集成Moment.js
 - 7.4.3 渲染时间日期
 - 7.5 使用Faker生成虚拟数据
 - 7.6 使用Flask-DebugToolbar调试程序
 - 7.7 Flask配置的两种组织形式
 - 7.7.1 环境变量优先
 - 7.7.2 实例文件夹覆盖
 - 7.8 本章小结

- 第8章 个人博客**
 - 8.1 大型项目结构
 - 8.1.1 使用蓝本模块化程序
 - 8.1.2 使用类组织配置
 - 8.1.3 使用工厂函数创建程序实例
 - 8.2 编写程序骨架

- 8.2.1 数据库
- 8.2.2 模板
- 8.2.3 表单
- 8.2.4 视图函数
- 8.2.5 电子邮件支持
- 8.3 编写博客前台
 - 8.3.1 分页显示文章列表
 - 8.3.2 显示文章正文
 - 8.3.3 文章固定链接
 - 8.3.4 显示分类文章列表
 - 8.3.5 显示评论列表
 - 8.3.6 发表评论与回复
 - 8.3.7 支持回复评论
 - 8.3.8 网站主题切换
- 8.4 初始化博客
 - 8.4.1 安全存储密码
 - 8.4.2 创建管理员用户
- 8.5 使用Flask-Login管理用户认证
 - 8.5.1 获取当前用户
 - 8.5.2 登入用户
 - 8.5.3 登出用户
 - 8.5.4 视图保护
- 8.6 使用CSRFProtect实现CSRF保护
- 8.7 编写博客后台
 - 8.7.1 文章管理
 - 8.7.2 评论管理
 - 8.7.3 分类管理
- 8.8 本章小结

第9章 图片社交网站

- 9.1 项目组织架构
 - 9.1.1 功能式架构
 - 9.1.2 分区式架构
 - 9.1.3 混合式架构
 - 9.1.4 如何选择
- 9.2 编写程序骨架
 - 9.2.1 数据库模型与虚拟数据
 - 9.2.2 模板与静态文件
- 9.3 高级用户认证
 - 9.3.1 用户注册
 - 9.3.2 验证邮箱地址
 - 9.3.3 使用装饰器过滤未确认用户
 - 9.3.4 密码重置
- 9.4 基于用户角色的权限管理
 - 9.4.1 角色与权限模型
 - 9.4.2 设置角色与权限
 - 9.4.3 写入角色与权限
 - 9.4.4 验证用户权限
- 9.5 使用Flask-Dropzone优化文件上传
 - 9.5.1 配置Flask-Dropzone
 - 9.5.2 渲染上传区域
 - 9.5.3 处理并保存上传图片
- 9.6 使用Flask-Avatars处理用户头像
 - 9.6.1 默认头像

- 9.6.2 生成随机头像
- 9.7 图片展示与管理
 - 9.7.1 在用户主页显示图片列表
 - 9.7.2 图片详情页
 - 9.7.3 上一张下一张跳转
 - 9.7.4 删除确认模态框
 - 9.7.5 举报图片
 - 9.7.6 图片描述
 - 9.7.7 图片标签
 - 9.7.8 用户资料弹窗
- 9.8 收藏图片
 - 9.8.1 使用关联模型表示多对多关系
 - 9.8.2 添加和取消收藏
 - 9.8.3 收藏者和收藏页面
- 9.9 用户关注
 - 9.9.1 自引用多对多关系
 - 9.9.2 关注与取消关注
 - 9.9.3 显示关注用户列表
 - 9.9.4 使用AJAX在弹窗中执行关注操作
- 9.10 消息提醒
 - 9.10.1 提醒消息在数据库中的表示
 - 9.10.2 创建提醒
 - 9.10.3 显示和管理提醒
 - 9.10.4 通过轮询实时更新未读计数
- 9.11 用户资料与账户设置
 - 9.11.1 编辑個人資料
 - 9.11.2 自定义头像
 - 9.11.3 更改密码
 - 9.11.4 提醒消息开关
 - 9.11.5 将收藏设为仅自己可见
 - 9.11.6 注销账号
- 9.12 首页与探索
 - 9.12.1 获取正在关注用户的图片
 - 9.12.2 使用联结和分组查询获取热门标签
 - 9.12.3 使用数据库通用函数获取随机图片
- 9.13 使用Flask-Whooshee实现全文搜索
 - 9.13.1 创建索引
 - 9.13.2 搜索表单
 - 9.13.3 显示搜索结果
- 9.14 编写网站后台
 - 9.14.1 用户管理
 - 9.14.2 资源管理
 - 9.14.3 面向管理员的用户资料编辑
- 9.15 本章小结

第10章 待办事项程序

- 10.1 使用JavaScript和AJAX编写单页程序
 - 10.1.1 单页程序的模板组织
 - 10.1.2 在根页面内切换子页面
 - 10.1.3 生成测试账户
 - 10.1.4 添加新待办条目
- 10.2 国际化与本地化
 - 10.2.1 使用Flask-Babel集成Babel
 - 10.2.2 区域和语言

- 10.2.3 文本的国际化
- 10.2.4 文本的本地化
- 10.2.5 时间与日期的本地化
- 10.3 设计并编写Web API
- 10.3.1 认识Web API
- 10.3.2 设计优美实用的Web API
- 10.3.3 使用Flask编写Web API
- 10.3.4 使用OAuth认证
- 10.3.5 资源的序列化
- 10.3.6 资源的反序列化
- 10.3.7 Web API的测试与发布
- 10.4 本章小结

第11章 在线聊天室

- 11.1 编写程序骨架
- 11.2 Gravatar头像
- 11.3 使用Flask-SocketIO建立实时双向通讯
- 11.3.1 建立Socket.IO连接
- 11.3.2 一条消息的旅程
- 11.3.3 在线人数统计
- 11.3.4 通讯频道分离
- 11.4 使用Flask-OAuthlib实现第三方登录
- 11.4.1 编写OAuth客户端
- 11.4.2 注册OAuth程序
- 11.4.3 处理OAuth2授权
- 11.4.4 处理OAuth1授权
- 11.5 聊天室功能增强
- 11.5.1 无限滚动加载历史消息
- 11.5.2 Markdown支持
- 11.5.3 代码语法高亮
- 11.5.4 标签页消息提醒
- 11.5.5 浏览器桌面通知
- 11.5.6 消息管理
- 11.6 本章小结

第三部分 高级篇

- ## 第12章 自动化测试
- 12.1 认识自动化测试
 - 12.2 Flask测试客户端
 - 12.3 使用unittest编写单元测试
 - 12.3.1 Flask程序的测试固件
 - 12.3.2 编写测试用例
 - 12.3.3 组织测试
 - 12.3.4 运行测试
 - 12.4 使用Selenium进行用户界面测试
 - 12.4.1 安装浏览器与驱动
 - 12.4.2 准备测试环境
 - 12.4.3 编写测试代码
 - 12.5 使用Coverage.py计算测试覆盖率
 - 12.5.1 基本用法
 - 12.5.2 获取测试覆盖率
 - 12.6 使用Flake8检查代码质量
 - 12.7 本章小结

第13章 性能优化

13.1 程序性能分析

13.1.1 函数性能分析

13.1.2 数据库查询分析

13.2 使用Flask-Caching设置缓存

13.2.1 缓存视图函数

13.2.2 缓存其他函数

13.2.3 更新缓存

13.2.4 使用Redis作为缓存后端

13.3 使用Flask-Assets优化静态资源

13.3.1 注册资源集

13.3.2 生成资源集文件

13.3.3 在模板中加载资源集

13.4 本章小结

第14章 部署上线

14.1 传统部署VS云部署

14.2 基本部署流程

14.3 部署前的准备

14.3.1 更新程序配置

14.3.2 创建生产环境专用的程序实例

14.3.3 设置迁移工具

14.3.4 程序日志

14.3.5 手动导入环境变量

14.3.6 HTTPS转发

14.4 部署到Linux服务器

14.4.1 使用OpenSSH登录远程主机

14.4.2 安装基本库和工具

14.4.3 安全防护措施

14.4.4 推送代码并初始化程序环境

14.4.5 使用Gunicorn运行程序

14.4.6 使用Nginx提供反向代理

14.4.7 使用Supervisor管理进程

14.4.8 更新部署后的程序

14.5 部署到PythonAnywhere

14.5.1 反向代理设置

14.5.2 创建PythonAnywhere程序

14.5.3 推送代码并初始化程序环境

14.5.4 创建数据库

14.5.5 设置虚拟环境

14.5.6 静态文件

14.5.7 运行和更新程序

14.6 部署到Heroku

14.6.1 通过Heroku Git部署

14.6.2 使用GitHub部署

14.7 下一步做什么？

14.8 本章小结

第15章 Flask扩展开发

15.1 扩展的命名

15.2 扩展项目骨架

15.3 编写扩展类

15.4 添加扩展配置

15.5 实现扩展功能

- 15.5.1 加载静态资源
- 15.5.2 创建分享组件
- 15.5.3 在移动设备上隐藏
- 15.6 开源发布前的准备
 - 15.6.1 添加文档字符串与注释
 - 15.6.2 编写README与文档
 - 15.6.3 为打包做准备
 - 15.6.4 编写示例程序
 - 15.6.5 编写单元测试
- 15.7 发布到PyPI
 - 15.7.1 创建PyPI账号
 - 15.7.2 使用setuptools打包
 - 15.7.3 使用twine上传
- 15.8 编写良好的扩展
- 15.9 本章小结

第16章 Flask工作原理与机制解析

- 16.1 阅读Flask源码
 - 16.1.1 获取Flask源码
 - 16.1.2 如何阅读源码
 - 16.1.3 Flask发行版本分析
- 16.2 Flask的设计理念
 - 16.2.1 “微”框架
 - 16.2.2 两个核心依赖
 - 16.2.3 显式程序对象
 - 16.2.4 本地上下文
 - 16.2.5 三种程序状态
 - 16.2.6 丰富的自定义支持
- 16.3 Flask与WSGI
 - 16.3.1 WSGI程序
 - 16.3.2 WSGI服务器
 - 16.3.3 中间件
- 16.4 Flask的工作流程与机制
 - 16.4.1 Flask中的请求响应循环
 - 16.4.2 路由系统
 - 16.4.3 本地上下文
 - 16.4.4 请求与响应对象
 - 16.4.5 session
 - 16.4.6 蓝本
 - 16.4.7 模板渲染
- 16.5 本章小结
- • • • • (收起)

[Flask Web开发实战 下载链接1](#)

标签

Python

Flask

Web

python

flask

web

编程

计算机

评论

超出预期，好书吹爆！

应该是学Flask最好最全面的书了，比起狗书内容更多实用性更强。

实体书和电子书要分别购买吗？已经买了实体书，很厚的一本，带着有点不方便，不知道电子书是不是需要再购买一次

目前来看，这是我看过的所有讲Flask中最全面清晰的了，当时没看到这本书可惜了

书没读过，因缘际会看了书中的几个demo源代码，很有启发，谢谢

强烈推荐的一本书，没什么废话。要是有涉及web安全的内容就好了

很棒

糙快猛的flask入门

内容丰富全面，项目可以有时间试试

说读完了，是读完了。说没读完，是因为还要读源码消化下。算是看过flask0.1"最小模型"的源码了吧

Flask入门必备，这本书的纸特别适合阅读学习！

繁琐，编码视频+源码 才是最好的学习编程方式

作者自己水平咋样不说，书写得肯定不好，写书绝对不是写手册。

后端一条龙，基本都讲到了，而且很详细，还提供了很多途径答疑。

入门好书

专业实践、毕业设计都参考了这本书，帮助挺大

这本书比那本开发实战高到不知哪里去了，条理清晰、代码详实，作者还在坚持维护更新示例代码，支持作者！！

非常全面，flask学习利器

小白看起来还是不错的，不单单是实例，也介绍运行原理。可以加深对于flask、乃至web运行的理解。

干货，而且很新。

[Flask Web开发实战 下载链接1](#)

书评

大家好，我是本书的作者。这篇文章列出了本书的勘误、源码等相关资源。衷心的希望这本书能够帮助到你，同时也为书中包含的错误为你带来的不便说声抱歉。勘误 & FAQ & 可改进实现 & 版本更新记录
在本书的Meta仓库helloflask (<https://github.com/greyli/helloflask>...)

经过这一段漫长的Flask之旅，你应该收获了不少东西。虽然本书结束了，但你的学习还没有止步。希望这本书可以让你的某些想法走进现实，带给人们一些特别的记忆。也希望你可以慷慨的分享你的代码、经验和思想，因为你正和其他人一样，用你的方式改变着这个世界。但愿这本书帮到...

刚刚入门行业，python爬虫xD。在某2上面抽奖得此书 有时候就特别相信命
书评一定要仔细看才能写（中间拖延不好意思讲，工作赶，回家懒。。。）
刚刚拿到书时候，没结合git源码核心基础知识一起看，单书看了大概五分之一，挺有意思 周末clone代码仔细看，发现都是干货。为啥说都...

1. 今天刚拿到书，我还特意买了签名版的，大致看了下目录，由浅入深，很赞
2. 了解作者的，应该都知道他在知乎上有个 helloflask 专栏，里面的文章质量很高
3. 现在只看到第二章，不能做过多的评价，不过可以看的出来作者的编程功底还是很深的。
。因为之前看过那本狗书，这本要...

书第一时间就入手了，写的非常详实。

更好的隐藏福利是：作者非常热情，作为一个自学编程的小白，我经常哪些很琐碎的小问题向他请教，作者大大都会不厌其烦的认真回答。

目前已经跟着书本学完了前面的基础知识，并且完成了sayhello和bluelog这2个项目，并且在阿里云部署上线， ...

买了实体书，电子书要再购买一次吗？实体书太厚带着不方便，并且实体书背的胶太薄，翻了几次现在几乎从中间裂开了，现在里面的勘误大部分已经改正过了，还没有看完，只看了前三章，看完再来评价买了实体书，电子书要再购买一次吗？实体书太厚带着不方便，并且实体书背的胶太薄...

[Flask Web开发实战 下载链接1](#)