

10 Things You Should Know About Django



北京 上海 珠海 苏州 杭州

主办: PyChina.org, GDG

About Me

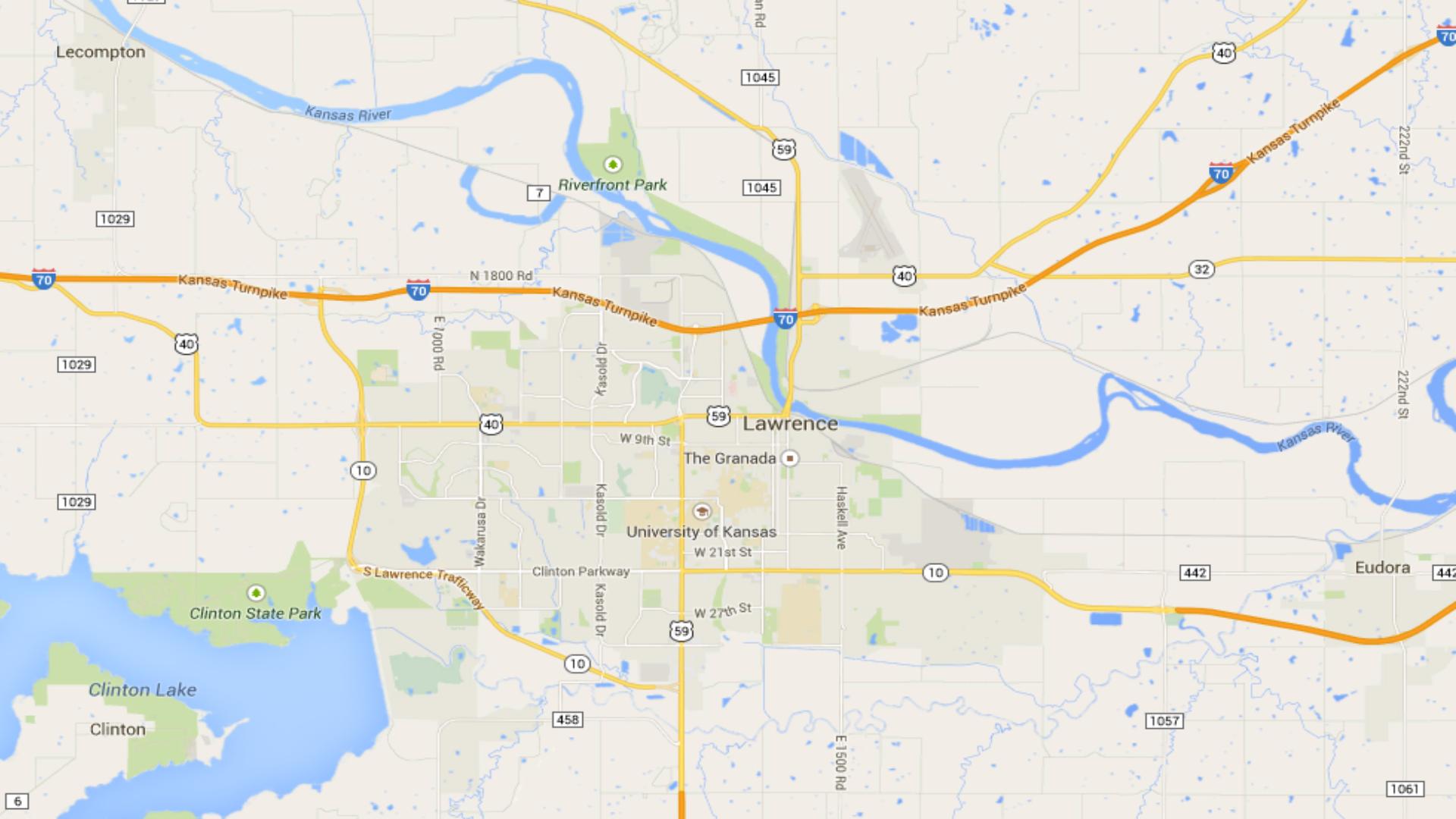
- Python user since 2007
- Build web based systems using Python + Django mostly
- Gain interests in infrastructure gradually
- Working on my own startup since 2014



twitter: [@adieu](https://twitter.com/adieu)
github: github.com/adieu
website: www.adieu.me

#1

History helps you understand





Gay marriage now legal in Kansas

16 comments



Photo by Nick Krug

The U.S. Supreme Court said Wednesday that same-sex marriages can go ahead in Kansas.

Source document: Supreme Court denies Kansas' stay on lifting gay marriage ban



Adrian Holovaty



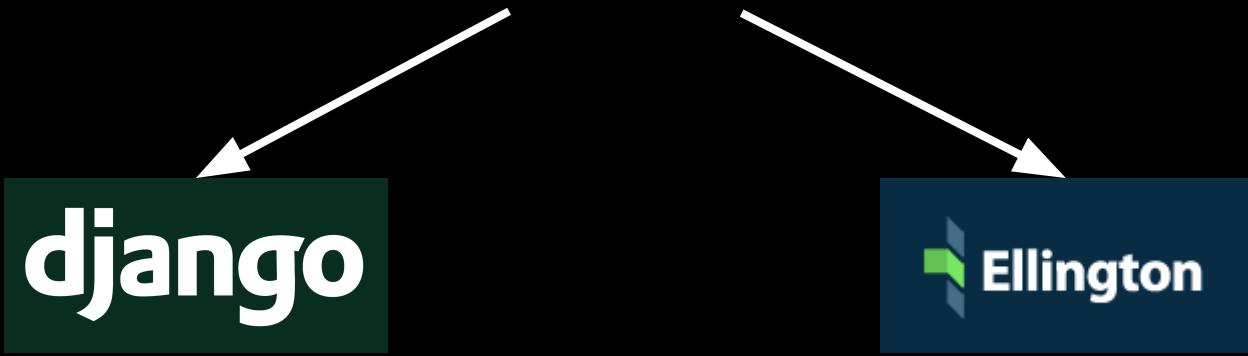
Simon Willison



Jacob Kaplan-Moss

Photo by Jason Samsa

The CMS



django

Ellington

Main Focus

Fast Development

Collaboration

Maintainability

Code Reuse

#2

Start with the Basics



w3schools.com

the world's largest web development site

educate yourself!

beginners and experts

Search w3schools.com:

Google™ Custom Search



HTML/CSS

- » Learn HTML
- » Learn HTML5
- » Learn CSS
- » Learn CSS3
- » Learn Bootstrap

JavaScript

- » Learn JavaScript
- » Learn jQuery
- » Learn jQueryMobile
- » Learn AngularJS
- » Learn AJAX
- » Learn JSON
- » Learn Google Maps

Server Side

- » Learn SQL
- » Learn PHP
- » Learn ASP
- » Learn ASP.NET
- » Learn VBScript



HTML

[HTML Tutorial](#)

[HTML Tag Reference](#)



CSS

[CSS Tutorial](#)

[CSS Reference](#)



JavaScript

[JavaScript Tutorial](#)

[JavaScript Reference](#)



SQL

[SQL Tutorial](#)

[SQL Reference](#)



PHP

[PHP Tutorial](#)

[PHP Reference](#)



JQuery

[JQuery Tutorial](#)

[JQuery Reference](#)



LEARN PYTHON THE HARD WAY

3RD
EDITION

Welcome to the 3rd Edition of Learn Python the Hard Way. You can visit the companion site to the book at <http://learnpythononthehardway.org/> where you can purchase digital downloads and paper versions of the book. The free HTML version of the book is available at <http://learnpythononthehardway.org/book/>.

Table Of Contents

- [Preface](#)
- [Introduction: The Hard Way Is Easier](#)
- [Exercise 0: The Setup](#)
- [Exercise 1: A Good First Program](#)
- [Exercise 2: Comments And Pound Characters](#)
- [Exercise 3: Numbers And Math](#)
- [Exercise 4: Variables And Names](#)
- [Exercise 5: More Variables And Printing](#)
- [Exercise 6: Strings And Text](#)
- [Exercise 7: More Printing](#)
- [Exercise 8: Printing, Printing](#)
- [Exercise 9: Printing, Printing, Printing](#)
- [Exercise 10: What Was That?](#)

1
2
3
4
MAIN



PLAY VIDEO



PREVIOUS



NEXT



HELP



Follow

A photograph of a spiral staircase viewed from the bottom, looking upwards. The stairs are white and curve upwards in a clockwise direction. In the center of the spiral is a large, octagonal skylight with a grid pattern. The ceiling above the skylight is also octagonal and has several recessed lights. The overall lighting is bright and airy.

The Upward Spiral

#3

It's just a Web Framework

A basic web server

```
import socket

HOST = ""
PORT = 80
listen_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
listen_socket.bind((HOST, PORT))
listen_socket.listen(1)
connection, address = listen_socket.accept()
request = connection.recv(1024)
connection.sendall("HTTP/1.1 200 OK
Content-type: text/html

<html>
    <body>
        <h1>Hello, World!</h1>
    </body>
</html>""")
connection.close()
```

WSGI

Web Server Gateway Interface

From Wikipedia, the free encyclopedia



This article's [lead section](#) may not adequately [summarize](#) key points of its contents. Please consider expanding the lead to [provide an accessible overview](#) of all important aspects of the article. *(April 2012)*

The Web Server Gateway Interface is a specification for simple and universal [interface](#) between [web servers](#) and [web applications](#) or frameworks for the [Python programming language](#). It was originally specified in PEP 333^[1] authored by Phillip J. Eby, and published on 7 December 2003. It has since been adopted as a standard for Python web application development. The latest version of the specification is v1.0.1, also known as PEP 3333, published on 26 September 2010.^[2]

Contents [hide]

- [1 Idea](#)
- [2 Specification overview](#)
- [3 Example application](#)
- [4 Example of calling an application](#)
- [5 WSGI-compatible applications and frameworks](#)
- [6 Wrappers](#)
- [7 WSGI and Python 3](#)
- [8 See also](#)
- [9 References](#)
- [10 External links](#)

Idea [edit]

Python web application frameworks have been a problem for new Python users because the choice of web framework would limit the choice of usable [web servers](#), and vice versa.^[3] Python applications were often designed for only one of [CGI](#), [FastCGI](#), [mod_python](#) or some other custom API of a specific web-server.

WSGI^[2] was created as a low-level [interface](#) between web servers and web applications or frameworks to promote common ground for portable web application development.

A mini web framework

```
from importd import d
d(DEBUG=True)

@d("/")
def idx(request):
    return "index.html"

@d("/post/<int:post_id>/")
def post(request, post_id):
    return "post.html", {"post_id": post_id}

if __name__ == "__main__":
    d.main()
```

A simple Django site

```
# settings.py
```

```
DEBUG = True  
ROOT_URLCONF = 'urls'  
TEMPLATE_DIRS = ['.']
```

```
# urls.py
```

```
from django.conf.urls.defaults  
import *  
  
import views
```

```
urlpatterns = patterns("",  
    (r'^$', views.index),  
    (r'^test/(\d+)/$', views.test),  
)
```

```
# views.py
```

```
from django.shortcuts import render  
  
def index(request):  
    return render(request, 'index.html', {})
```

```
def test(request, id):  
    return render(  
        request, 'test.html', {'id': id}  
)
```

A single file Django site

```
# Setup
from django.conf import settings
if not settings.configured:
    settings.configure(
        DEBUG = True,
        ROOT_URLCONF = 'web',
        TEMPLATE_DIRS = [''],
    )

from django.conf.urls.defaults import patterns
urlpatterns = patterns("",
    (r'^$', 'web.index'),
    (r'^test/(\d+)/$', 'web.test'),
)

# Handlers
from django.shortcuts import render

def index(request):
    return render(request, 'index.html', {})

def test(request, id):
    return render(request, 'test.html', {'id': id})

# Running
if __name__ == '__main__':
    from django.core.management import execute_from_command_line
    execute_from_command_line()
```

Web Framework Basics

- The No. 1 issue web frameworks solve is mapping urls to your handler functions
- Web framework is all about common practice shared by a group of developers
- Choose a web framework when you share the same taste with the author
- Trust the author's choice and understand why it's implemented in its way
- Contribute a pull request instead of maintaining a custom fork

#4

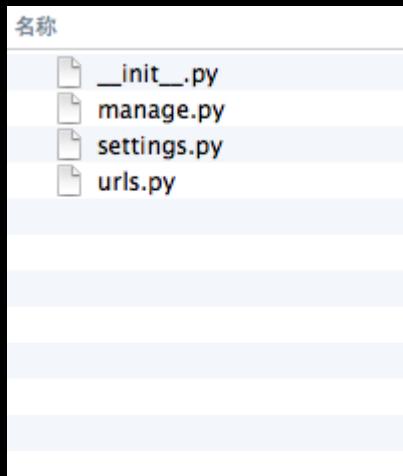
It's just a Python Package

Everything is Python Package

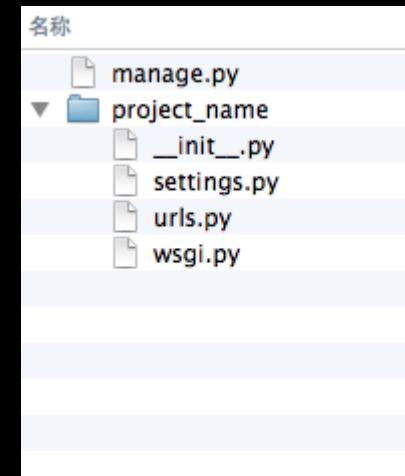
- Django
- Third Party Packages
- Your Application
- Your Site

The default project

Django 1.0



Django 1.7



Recommended project layout

名称	修改日期	大小	种类
module1	今天下午10:07	--	文件夹
module1	今天下午10:07	--	文件夹
__init__.py	今天下午10:07	0字节	Python Source
admin.py	今天下午10:07	63字节	Python Source
migrations	今天下午10:07	--	文件夹
models.py	今天下午10:07	57字节	Python Source
tests.py	今天下午10:07	60字节	Python Source
views.py	今天下午10:07	63字节	Python Source
setup.py	今天下午10:05	0字节	Python Source
module2	今天下午10:07	--	文件夹
module2	今天下午10:07	--	文件夹
setup.py	今天下午10:06	0字节	Python Source
mysite	今天下午10:08	--	文件夹
manage.py	今天下午10:05	249字节	Python Source
mysite	今天下午10:09	--	文件夹
__init__.py	今天下午10:05	0字节	Python Source
module	今天下午10:09	--	文件夹
settings.py	今天下午10:05	2 KB	Python Source
urls.py	今天下午10:05	275字节	Python Source
wsgi.py	今天下午10:05	387字节	Python Source
requirements.txt	今天下午10:08	0字节	纯文本文稿
setup.py	今天下午10:08	0字节	Python Source

Python Tricks

- virtualenv
- pip install -e
- python setup.py sdist upload
- gdb
- monkey patch
- pypy

#5

Battery included but replaceable

Batteries

django.db

django.forms

django.templates

django.views

django.middleware

django.test

django.db.migrations

django.contrib.auth

django.contrib.admin

django.contrib.sessions

django.contrib.gis

django.contrib.sites

The bench

Module	Package
Template	Jinja2
ORM	SQLAlchemy
Routing	Rhetoric
Admin	django-xadmin
Schema Migration	South

#6

Postgres is your friend

Improved PostgreSQL support in Django

by Marc Tamlyn

[Home](#)[Updates 1](#)[Backers 334](#)[Comments 11](#) Didcot, UK Web

Funded! This project was successfully funded on March 14, 2014.

Improved
PostgreSQL support for

django

334

Backers

£14,411

pledged of £2,500 goal

0

seconds to go



Project by

Marc Tamlyn

Didcot, UK

 First created · 2 backed

Postgres is the community choice

- django.contrib.postgres
- django.contrib.gis
- django-hstore
- django-tenant-schemas
- django-pgjson
- django-postgrespool
- django-ext-pgarray

#7

Performance is not your priority

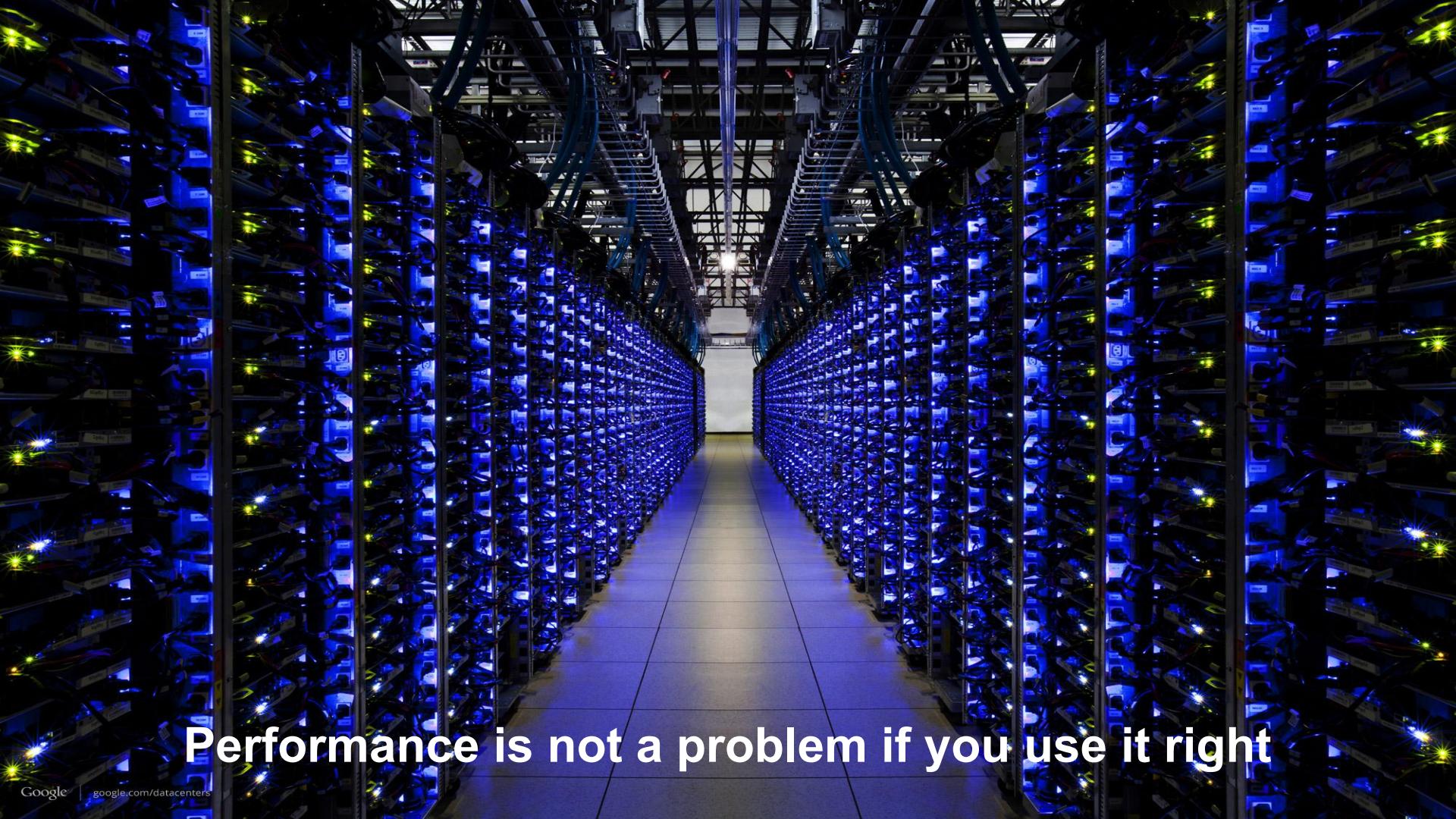
The slogan

The old one:

The web framework for perfectionists with deadlines.

The new one:

Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design.



Performance is not a problem if you use it right

Performance Improvements

django-debug-toolbar

Gevent

Gunicorn

Cache

DB tweaks

Application server farms

#8

Choose Django for the Ecosystem

1-60 of 71 results for Books : Computers & Technology : Web Development & Design : Programming : "django"

Sort by Relevance

Show results for

< Any Category
< Books
< Computers & Technology
< Web Development & Design
Programming
Python (58)
+ See more

Refine by

Eligible for Free Shipping
Free Shipping by Amazon

Delivery Day

Get It Today
 Get It by Tomorrow

New Releases

Last 30 days (2)
Last 90 days (5)
Coming Soon (2)

Book Format

Paperback (58)
Kindle Edition (13)

Book Language

English (41)

Author

www.amazon.com/Childrens-Books/b/ref=sv_b_4?ie=UTF8&node=4

Related Searches: django python, flask, python.

Book Format: Paperback | Kindle Edition

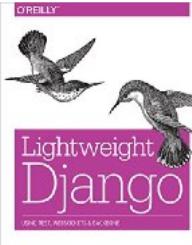


Two Scoops of Django: Best Practices For Django 1.6 Feb 1, 2014

\$35.20

Paperback

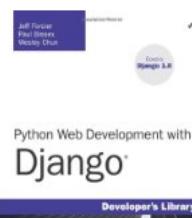
★★★★★ ▾ 38



Lightweight Django Nov 13, 2014

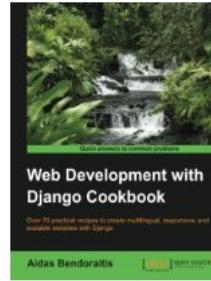
\$17.59 - \$32.59

Paperback, Kindle Edition



Python Web Development with Django

Developer's Library

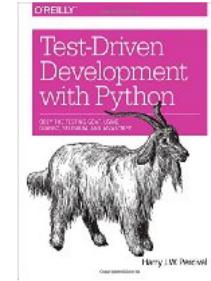


Web Development with Django Cookbook Oct 16, 2014

\$16.49 - \$44.99

Paperback, Kindle Edition

★★★★★ ▾ 2

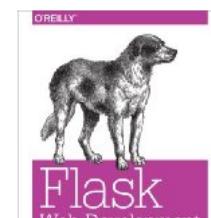


Test-Driven Development with Python Jun 29, 2014

\$13.49 - \$29.94

Paperback, Kindle Edition

★★★★★ ▾ 8



Flask Web Development

ACTIVITIES
ADMIN INTERFACE
ANALYTICS
ANTI-SPAM
API CREATION
ASSET MANAGERS
AUTHENTICATION

AUTHORIZATION
AUTO-COMPLETE
AWARDS AND BADGES
BLOGS
BOOTSTRAPS
CACHING
CALENDAR

CAPTCHA
CHAT
CMS
COMMENTING
CONFIGURATION
COOKIECUTTER P...
COUNTRIES

CUSTOM MODELS
DATABASE MIGRA...
DATA TOOLS
DEPLOYMENT
DESIGN
DEVELOPER TOOLS
DJANGO-CMS

DJANGO SHOP PL...
DOCUMENT MANAG...
E-COMMERCE
EMAIL
ERROR HANDLING
FEEDBACK
FIELDS

Django Packages is a directory of reusable apps, sites, tools, and more for your Django projects.

2562 packages and counting!

516 packages compatible with Python 3!

Add packages and grids

Know of any packages not listed here? Add them now!
It's quick and easy.

[add package »](#)

Or add a grid comparing the features of 2 or more similar packages.

[add grid »](#)

Featured Grid

Cookiecutter Packages

Cookiecutter-powered project, app, and utility bootstraps for Python, Go, Jinja, Node.js, etc.

Latest Python 3 Package Releases

django-email-confirm-la v0.2.2

Django email confirmation for any Model and any Field.

django-coupons v1.0.1

A reuseable Django application for coupon generation and handling

django-coupons v1.0.0

A reuseable Django application for coupon generation and handling

django-media-helper v0.2.2-bugfix

A drop-in app to manage, resize, and deliver uploaded images.

Waliki v0.3.1

A wiki engine powered by Django and Git

Latest 5 Packages Added

django-ftp-deploy

django-tablets

The ultimate database-driven Django template experience.

django-compat

For- and backwards compatibility layer for Django 1.4 to 1.7

django-email-confirm-la

Django email confirmation for any Model and any Field.

form_designer

A simple form designer for FeinCMS

Random 5

django-drip

Use the admin to manage drip campaign emails using

Package Categories

Apps (2063)



Search

django

Search

	Repositories	39,414
	Code	8,452,661
	Issues	77,822
	Users	246

Languages

Python	27,499
JavaScript	4,349
CSS	1,227
Shell	252
Ruby	170
PHP	101
VimL	68
Java	66
Puppet	32
C	18

We've found 39,414 repository results

Sort: Best match ▾

[django/django](#)

Python ★ 11,863 ⚡ 4,648

The Web framework for perfectionists with deadlines.

Updated 3 hours ago

[akulakov/django](#)

Python ★ 71 ⚡ 119

Updated on 12 Apr

[jpetazzo/django](#)

Python ★ 13 ⚡ 6

Django on DotCloud tutorial

Updated on 25 Feb 2012

Tagged Questions

info

newest

7

featured

frequent

votes

active

unanswered

Django is an open source web application framework, written in Python. Django is designed to reduce the effort required to create complex database-driven websites.

[learn more...](#) | [improve tag wiki](#) | [top users](#) | [synonyms](#) | [django jobs](#)

0

votes

0

answers

3 views

How to test view mixins in Django REST Framework?

In my Django site I use Django REST framework. To not repeat myself, I created a couple mixins to simplify writing my view sets (here is an example of a similar style of mixin). My question is how do ...

[django](#)[unit-testing](#)[integration-testing](#)[django-rest-framework](#)

asked 18 mins ago

Lubomír Sedlář

386 • 3 • 8

0

votes

2

answers

11 views

Django TypeError after installing tinymce

after installing tinymce and just including it to INSTALLED_APPS in settings.py I get this error when I'm trying to run local server: (django) \mypath\ python manage.py runserver Traceback (most ...

[python](#)[django](#)[tinymce](#)[typeerror](#)[nonetype](#)

asked 22 mins ago

Nhor

3 • 1

0

votes

0

answers

Why don't action_hooks written in windows for openshift work

I've written a web app in django using windows with a virtual env and doing it through powershell. Deploying the app using openshift has worked to a certain degree (thanks to the accepted answer on ...

[django](#)[git](#)[bash](#)[powershell](#)[openshift](#)

asked 40 mins ago

84,396

questions tagged

[django](#) [about »](#)

Blog

 Announcing Bosun, our new open source monitoring & alerting system

Featured on Meta

 Rolling question rate limits are now network-wide

Hot Meta Posts

9 [Flagging link-only questions](#)

17 [Should Docker questions go on StackOverflow or ServerFault or SuperUser?](#)

Related Tags

[python](#) × 37906[django-models](#) × 9423[django-forms](#) × 5174



django

Django (框架)

动态 精华 全部问题

热门排序 | 时间排序

Python 初学者想通过 Django 框架写一个博客，一个月内完成任务，大致的学习路线怎么安排？

8 周西, pythoner, 关注web开发技术

建议先弄清楚自己的目的：如果是为了学习，那么就好好学些python，然后学习http相关知识，django是开发时方便，但是对于学习web开发知识来说，并不是太合适；如果只是为了建个人博客，直接用wordpress，比自己写的好太多。

+ 关注问题 1条评论

59 vb, 知之甚少, 道不出来。hudongjie.com ...

我今天看到更新，真的佩服提问者

+ 关注问题 3条评论

121 志一, 互联网产品

一个小时看python的语法和基本数据类型：[woodpecker.org.cn/abyte...](#)；一个小时看HTTP基础，知道浏览器是怎么发请求的就行（Google搜“HTTP请求”）；一天时间过一遍django docs [docs.djangoproject.com/...](#)；同时搭环境让django应用能跑起来；开着python手册，开着dj...

显示全部

取消关注

1925 人关注了该话题

组织 · 管理 · 日志

描述

Django 是用 Python 语言写的开源 Web 开发框架，它鼓励快速开发，并遵循 MVC 设计和 BSD 版权。Django 根据比利时的爵士音乐家 Django Reinhardt 命名。 [修改](#)

父话题

Python 框架

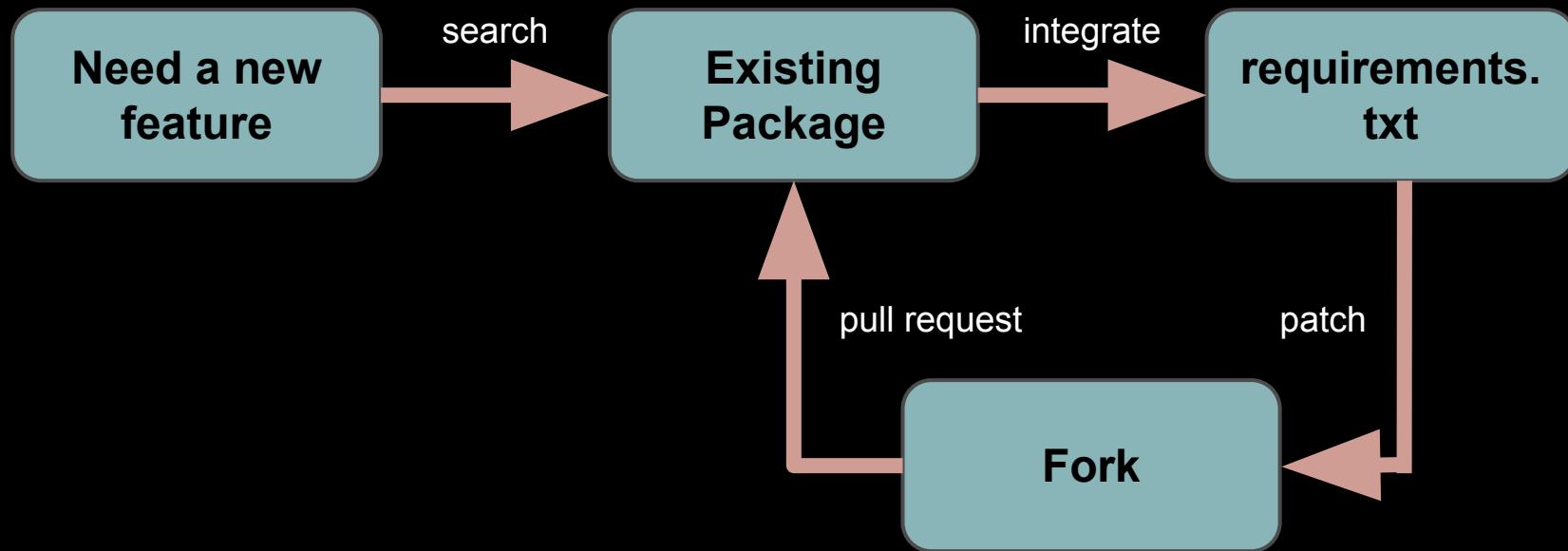
最佳回答者

 志一
互联网产品

回答了 1 个问题 · 关注

 松鼠奥利奥

Typical Django workflow



#9

Know the Limitations



Coding Style



Realtime

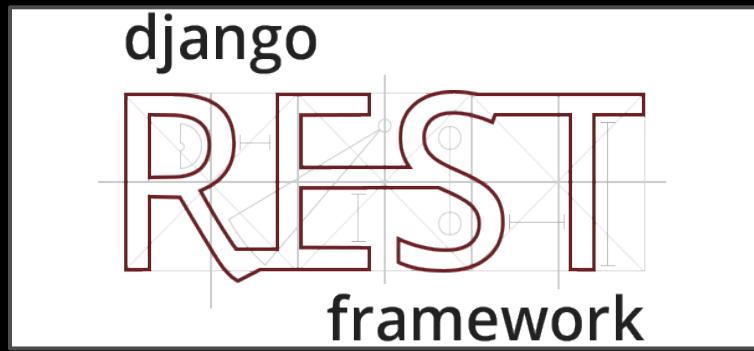
A wide-angle photograph of a large concrete dam during dusk or dawn. The dam features multiple arched gates at the base, from which a powerful stream of water is cascading down. Above the gates, the dam's structure rises, with a series of smaller arches and a flat top where several small buildings or control structures are visible. The sky is a mix of dark blues and light yellows, suggesting the time of day. In the foreground, a metal railing runs along the edge of a walkway or bridge.

Performance Critical

#10

Prepare for the Future

Restful API with multiple platform



SQL + NoSQL = ❤

The PostgreSQL logo is displayed on the left, featuring a blue icon of a person with a keyhole and the text "PostgreSQL". To the right, the slogan "The world's most advanced open source database." is written.

Home | About | Download | Documentation | Community | Developers | Support | Your

» Community
» Contributors
» Mailing Lists
 Subscribe
 User lists
 Developer lists
 pgsql-cluster-hackers
 pgsql-committers
 pgsql-hackers
 pgsql-reviewers
 pgsql-translators
 pgsql-www
 Regional lists
 Associations

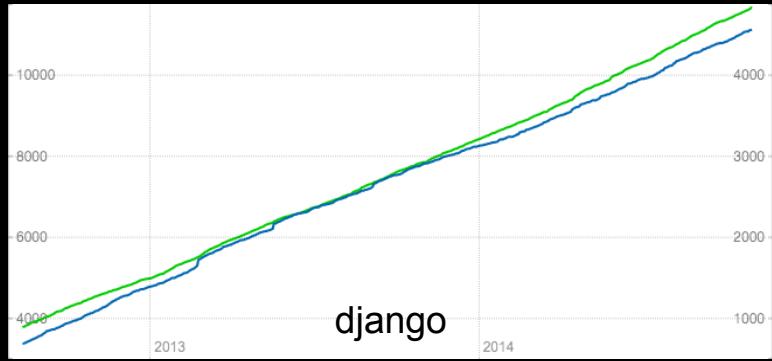
pgsql: Introduce jsonb, a structured format for storing json.

From: Andrew Dunstan <[andrew\(at\)dunsLane\(dot\)net](mailto:andrew(at)dunsLane(dot)net)>
To: [pgsql-committers\(at\)postgresql\(dot\)org](mailto:pgsql-committers(at)postgresql(dot)org)
Subject: pgsql: Introduce jsonb, a structured format for storing json.
Date: 2014-03-23 21:17:51
Message-ID: [E1WRpmB-0002et-MT@gemulon.postgresql.org](#) (view [raw](#) or [flat](#))
Thread: 2014-03-23 21:17:51 from Andrew Dunstan <[andrew\(at\)dunsLane\(dot\)net](mailto:andrew(at)dunsLane(dot)net)>
Lists: [pgsql-committers](#)

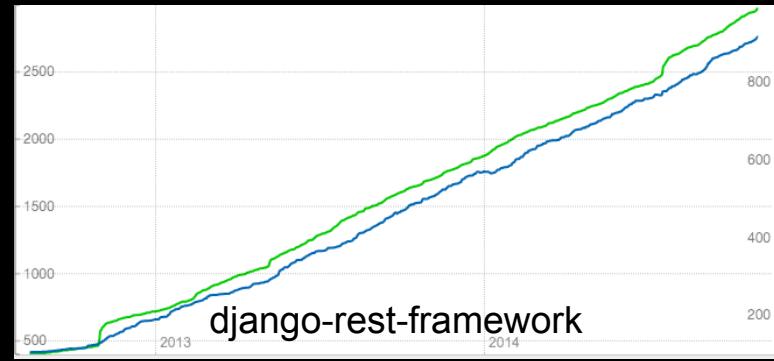
Introduce jsonb, a structured format for storing json.

The new format accepts exactly the same data as the json type. However, it is stored in a format that does not require reparsing the original text in order to process it, making it much more suitable for indexing and other operations. Insignificant whitespace is discarded, and the order of object keys is not preserved. Neither are duplicate object keys kept - the later value for a given key is the only one stored.

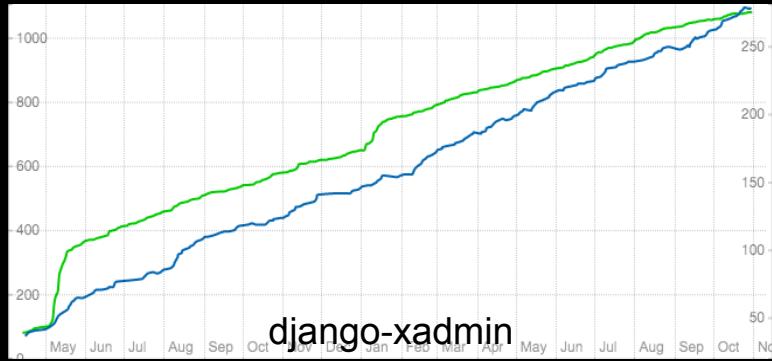
A bright future



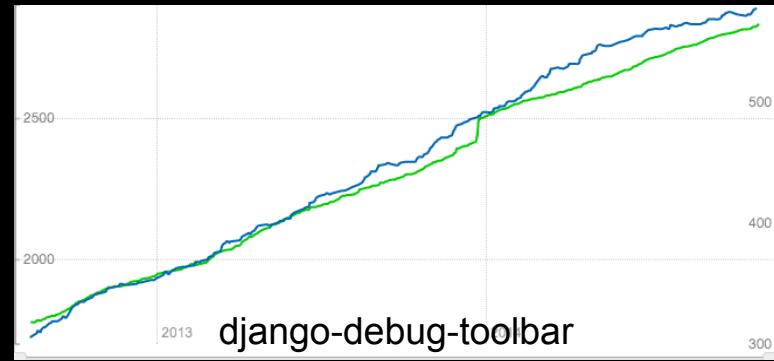
django



django-rest-framework



django-xadmin



django-debug-toolbar

Ready for the Open Source future?



contact@sebible.com

Links

<https://www.djangoproject.com/>

<http://www.ellingtoncms.com/>

<http://www.w3schools.com/>

<https://www.djangopackages.com/>

<http://learnpythononthehardway.org/book/>

<http://www.jeffknupp.com/blog/2014/03/03/what-is-a-web-framework/>

<https://github.com/amitu/importd>

<http://softwaremaniacs.org/blog/2011/01/07/django-micro-framework/en/>

<http://jinja.pocoo.org/docs/dev/>

<http://www.sqlalchemy.org/>

<https://github.com/avanov/Rhetoric>

<https://github.com/sshwsfc/django-xadmin>

<http://south.aeracode.org/>

<https://www.kickstarter.com/projects/mjtamlyn/improved-postgresql-support-in-django>

<https://github.com/djangonauts/django-hstore>

<https://github.com/bernardopires/django-tenant-schemas>

<https://github.com/djangonauts/django-pgjson>

<https://github.com/kennethreitz/django-postgrespool>

<https://github.com/niwibe/djorm-ext-pgarray>

<https://github.com/django-debug-toolbar/django-debug-toolbar>

<http://www.gevent.org>

[http://unicorn.org/](http://unicorn.org)

<https://www.djangopackages.com/>

<http://www.django-rest-framework.org/>

<http://www.postgresql.org/message-id/E1WRpmB-0002et-MT@gemulon.postgresql.org>

Thank You