
Knowledge Documentation

Release 1.0

Tower Joo

March 23, 2016

1	About	3
1.1	About me	3
1.2	Contact me	5
1.3	About the docs	5
2	I love Python	7
2.1	Python on Web	7
3	Management is with life all the time	9
4	Some blog entries	11
4.1	About the Great Read The Docs	11
5	Indices and tables	13

I love python and python related stuffs, e.g django, sphinx and etc.

To make everything in a good state, I use sphinx to generate the docs to server as a knowledge management system.

The content will include the stuffs I learned in the college life and working life. And the tips to remind me to prevent forgetting.

Note: Thousands of times' reading is less than one time's writing.

– quotted from a Chinese old proverb

.

Contents:

About

Contents

- *About*
 - *About me*
 - *Contact me*
 - *About the docs*

1.1 About me

My name is Tower Joo(in Chinese), which have the following meaning:

- Tower means a great traditional building, which stands for thousands of years
- Joo is my family name to match my Chinese name's pronunciation.

I'm living in [Beijing](#), China and I graduated from [GUCAS](#) for my master degree, and [XIPT](#) for my bachelor degree, in which my major is Computer Sciences.



I love [Python](#), which a great elegant and *stupid* language. I use [Python](#) to work on *Social Game*, *scripts*, *prototype* and etc. [Django](#), [Sphinx](#) and other [Python](#) related stuffs are my favorites too.

You can find my blog from [mindsbook](#) and you can also follow me at [@towerjoo](#) twitter (for twitter), and [@towerjoo](#) sina (for sina).



1.2 Contact me

You can contact me from the following ways:

- Email : zhutao AT halfquest.com
- Gtalk/MSN : same with Email
- Twitter: @towerjoo twitter
- Sina microblog: @towerjoo sina
- blog : mindsbook

Anything with the site or share of python/social game/startup related stuffs will be welcomed.

1.3 About the docs

These docs are for my knowledge management. Using [sphinx](#) to record *What I've learned, what should be noticed, what tips can be used to ease something.*

I hope the knowledge database will not just benefit myself but also can help others to avoid some difficulties and traps.

All docs will evolve.

I love Python

Python is a great language.

Contents:

2.1 Python on Web

Contents:

2.1.1 Gunicorn

Gunicorn ‘Green Unicorn’ is a Python WSGI HTTP Server for UNIX. It’s a pre-fork worker model ported from Ruby’s Unicorn project. The Gunicorn server is broadly compatible with various web frameworks, simply implemented, light on server resources, and fairly speedy.

Gunicorn and Django

1. you can install **Gunicorn** using `easy_install` or `pip`
2. add `gunicorn` to `INSTALLED_APPS` in django’s `settings.py`
3. run `python manage.py run_gunicorn -b 127.0.0.1:1234 --daemon` to start
4. configure `nginx` to make it proxy requests to gunicorn by adding `proxy_pass http://127.0.0.1:1234` in the config file

Then everything is done and you can visit your django app in `http://hostname:port` based on the `nginx` file config.

2.1.2 Django is great

tips

Note: when you got the queryset from calling `ModelClass.objects.filter(condition)` and refer the element using `[]` and save it at last, which **WON’T hit the database**, and it will beyond your expectation. See the below codes:

```
a = Blog.objects.filter(title__contains="django")
a[0].last_update_time = datetime.now()
a[0].save() # won't work as you expect
print a[0].last_update_time # this will print the old one

# but you can
b = a[0]
b.last_update_time = datetime.now()
b.save() # will work as expected
```

Management is with life all the time

Management is to make choices.

How can we make a cleverer choice? time and experience can tell you.

Contents:

Some blog entries

Blog can be a great way to write something in a free style or in a serious way. Just make it fit you~

Contents:

4.1 About the Great Read The Docs

“Hey, I’ve started to use [RTD](#) to do my knowledge management.”

I read the blog article from [Eric’s Announcing Read The Docs](#) days ago and I set up my online doc system 2 days ago. And [RTD](#) is really amazing! You can:

- write your docs using [sphinx](#)
- host your doc in any service with mainstream source management system support([git](#), [mercurial](#) or [svn](#))
- push to the server and the online doc will update in real time
- it’s written in [Django](#) (Check out the source code from [github](#))!

Why not to set up your own and it’s free!

Indices and tables

- `genindex`
- `modindex`
- `search`