
catsup Documentation

Release 0.0.8

whtsky

February 15, 2013

CONTENTS

Contents:

INSTALLATION

1.1 Setuptools and Pip

If you have pip installed

```
(sudo) pip install catsup
```

Or if you don't have pip, try easy_install instead

```
(sudo) easy_install catsup
```

1.2 Git

Install with git can always have the latest code

```
git clone git://github.com/whtsky/catsup.git
cd catsup

# We use git submodules to organize out theme.
# If you don't want the default theme(current version is sealscript)
# You can skip these command.
git submodule init
git submodule update

python setup.py install
```

1.3 Attention

Catsup uses misaka as the markdown engine.It requires C compiler.On Ubuntu, you may run

```
(sudo) apt-get install python-dev
```

If you're using a Mac, you need to install XCode and its Command Line Tools.

1.4 Windows User?

Sorry that I don't use Windows and I also don't know how to install it.That should works well with Cygwin. If you're using Windows and know how to install catsup on Windows, please fork this doc , add it and send a pull request.

GET STARTED

This section needs you have catsup installed.If you don't, please go and *Install it*

2.1 Create a blog

It's pretty simple to create a blog using catsup

```
cd ~
mkdir blog
cd blog
catsup init
```

You'll be asked some questions. These questions have default value, so if you don't want to change them, just hit enter.

2.2 Create a post

Create a post

```
touch posts/2013-02-08-helloworld.md
```

Attention: Post's filename should like `year-month-day-title.md`. `2013-2-2-hi.md` is not vailed.

Then edit it with your favorite editor

```
mate posts/2013-02-08-helloworld.md
```

2.3 Build your blog

Now you have written your first post.Let's build your blog

```
cd ~/blog
catsup build
```

You can also preview your blog before build it using *Preview Server*

WRITING

3.1 Markup

Catsup support markup of Markdown.

A post consists of three parts:

- Title
- Meta
- Content

Here's an example:

```
#Hello World!           <---- This is title
- tags: hello, world    <---- This is meta
- comment disabled
---
Hello, World!           <---- This is content
```

Title should always on the first line and starts with #

Meta is some information about the post. It's below title and above the first ---

Content is everything below the first ---

3.2 Meta

Metadata that catsup supports natively:

- tags - tags are separated by comma
- summary - used in pages
- description - used for SEO and Twitter Card
- comment - set to `disabled` for forbidding comment

Note that if you don't write `summary` and `description` in post's meta, catsup will create it for you.

CONFIGURATION

Catsup support json format config file. You can create a config file with

```
catsup init
```

4.1 Overview

The default json format config file:

```
{
  "site": {
    "name": "blogname",
    "url": "http://blog.com"
  },
  "author": {
    "name": "name",
    "twitter": "twitter"
  },
  "config": {
    "source": "posts",
    "output": "deploy",
    "static": "deploy/static",
    "static_prefix": "/static/",
    "feed": "/feed.xml",
    "per_page": 3,
    "posts_on_feed": 5,
    "analytics": "",
    "display_summary": false,
    "escape_md": false
  },
  "comment": {
    "allow": true,
    "system": "disqus",
    "disqus": "catsup",
    "duoshuo": "catsup"
  },
  "deploy": {
    "default": "rsync",
```

```
"git": {
  "repo": "repo url here",
  "branch": "master",
  "delete": true
},

"rsync": {
  "ssh_port": 22,
  "ssh_user": "username",
  "ssh_host": "123.45.6.78",
  "document_root": "~/website.com/",
  "delete": true
}
},

"theme": {
  "name": "sealscript",
  "vars": {
    "description": "a blog",
    "github": "whtsky",
    "links": [
      {
        "name": "catsup",
        "url": "https://github.com/whtsky/catsup",
        "description": "Awesome!"
      }
    ]
  }
}
}
```

THEME

5.1 Overview

Install a theme

```
catsup install git_repo
```

Install a theme in global

```
catsup install git_repo -g
```

Update a installed theme

```
catsup install theme_name
```

List all themes installed

```
catsup themes
```

5.2 Structure

Catsup uses Jinja2 as a Template Engine. You need to learn it if you want to design your own theme.

You can learn how to design your theme by demo:

- <https://github.com/whtsky/catsup-theme-sealscript>

A catsup theme should look like

```
-- README.md                <----- how to install/customize your theme.Used in GitHub.
-- static                   <----- static files
|   -- css
|   |   -- pygments_style.css <----- catsup uses Pygments to highlight code
|   |   -- style.css
-- templates                <----- template files
|   -- article.html
|   -- page.html
-- filters.py               <----- filters defined by theme
-- theme.py                 <----- meta file
```

Only `article.html` and `page.html` are required.

5.3 Meta File

A demo meta file

```
name = 'sealscript'
author = 'Lyric'
homepage = 'https://github.com/whtsky/catsup-theme-sealscript'
pages = [
    '404.html'
]
vars = {
    "github": "whtsky",
}
```

A theme meta consists of :

- name
- author
- homepage
- pages
- vars

5.3.1 What's Pages for?

By default, Catsup will render `article.html` for post, `page.html` for pagination, `tag.html` for tag page and `archive.html` for archive page. If you want to render more template(like `404.html`), please add template's name to pages.

5.3.2 What's Vars for?

Your theme may need some var that user defined in config file. But **they may miss some var**, so you need to give a default value in theme's meta file.

5.4 Variables

5.4.1 Global Variables

- site: `site` in user's config file.
- author: `author` in user's config file.
- config: `config` in user's config file.
- comment: `comment` in user's config file.
- theme: `“theme.vars“` in user's config file.

If you want more:

- `g.posts`: posts sorted with date.
- `g.tags`: tags sorted with posts' count

- `g.archives`: archives sorted with year

5.4.2 Templatable Variables

Templatable variables are only accessed in specify templates.

- `pagination`: available in `page.html`
- `post`: available in `article.html`
- `prev, next`: available in `article.html`, `tag.html` and `archive.html`

5.5 Filters and Functions

Catsup has a build-in function: `static_url`

```
<link rel="stylesheet" href="{{ static_url("css/style.css") }}" type="text/css" />
```

Every function in `filters.py` will be a filter. Catsup also has some build-in filter:

- `xmldatetime`
- `capitalize`

5.6 Template Marco

Catsup has some powerful marco to make your job easier

- `render_comment(post)`: render post's comment part.
- `meta(post)`: render post's meta tag. Should be used id `<head>`.
- `analytics()`: render google analytics code.

Use them like

```
<html>
  <head>
    <title>{{ post.title }}</title>
    {% from 'utils.html' import meta, analytics %}
    {{ meta(post) }}
    {{ analytics }}
  </head>
  <body>
    <article>
      <h1>{{ post.title }}</h1>
      {{ post.content }}
      {% from 'utils.html' import render_comment %}
      {{ render_comment }}
    </article>
  </body>
</html>
```

This is a simple `article.html` template using catsup's template marco. Isn't that simple?

GOODIES

6.1 Preview Server

Preview your blog **without build & deploy**

```
catsup server
catsup server -p 8000
```

6.2 Deploy Support

Help you deploy your blog via git or rsync

```
catsup deploy # Deploy via default way
catsup rsync # Deploy via rsync
catsup git # Deploy via git
```

Note: Catsup has GitHub Pages support. It will create a *CNAME* file when deploy via git.

6.3 Webhook

If you host your posts on GitHub or Bitbucket, catsup can generate your blog when you push to your repo.

You need to clone your repo and start a webhook server

```
git clone git://path/to/your/repo.git posts
catsup webhook -p 12580
```

Attention: Catsup webhook is not a daemon process. That means you may need to use [Supervisor](#) to turn it into daemon.

Then configure webhook on GitHub or Bitbucket. Here we use GitHub as an example:

- Go to the “admin” page for your project
- Click “Service Hooks”
- In the available service hooks, click “WebHook URLs”

- Type your url ¹
- Click “Update Settings”

Then when you push to GitHub, catsup will pull and generate your blog.

6.4 Twitter Card

Catsup has a build-in Twitter Card support via `meta marco`.

Click here if you don't know what Twitter Card is : <https://dev.twitter.com/docs/cards>

All you need to do is go to <https://dev.twitter.com/form/participate-twitter-cards> and submit your blog. Then wait for the magic works.

¹ If your server's ip is 1.2.3.4, you can type `http://1.2.3.4:12580/webhook`

CHANGELOG

7.1 Version 0.0.8

- Rewrite tag and archive code
- Add deploy support.(via git or rsync)

7.2 Version 0.0.7

Released on Feb. 7, 2013

- Add pagination for writing theme
- Rename excerpt to summary
- Add theme utils
- Support theme filters