

前言

前面一篇将的是如果搭建自己的MkDocs来管理自己的文档，搭建完成后，可以使用内设的小型服务器来生成一个静态的站点，当然了，你可以将你的site文件夹直接放到服务器上。比如，放在本地的Tomcat: apache-tomcat-8.0.36\webapps路径下，开启自己的Tomcat，就可以直接访问：<http://localhost:8080/site/>，就会默认打开site下的index.html。同理，我们也可以将他放到自己公司的服务器或者托管到GitHub Pages上。

项目发布到GitHub Pages

GitHub Pages 官方网址：<https://pages.github.com/>

打开此链接，根据指示一步一步操作即可，这个时候，小编英语十三级的水平起到了至关重要的作用！

1.选择 “User or organization site” ，然后创建一个repository(仓库)，这个仓库的名称必须是username.github.io，比如，小编的仓库的名称是KevinDGK.github.io。

2.选择你的客户端，比如，小编的是GitHub for Windows，所以说以下都是windows上的操作，如果是其他的客户端，其实相差并不是很多，英语翻译的好，就没有问题。

3.将仓库clone到本地。

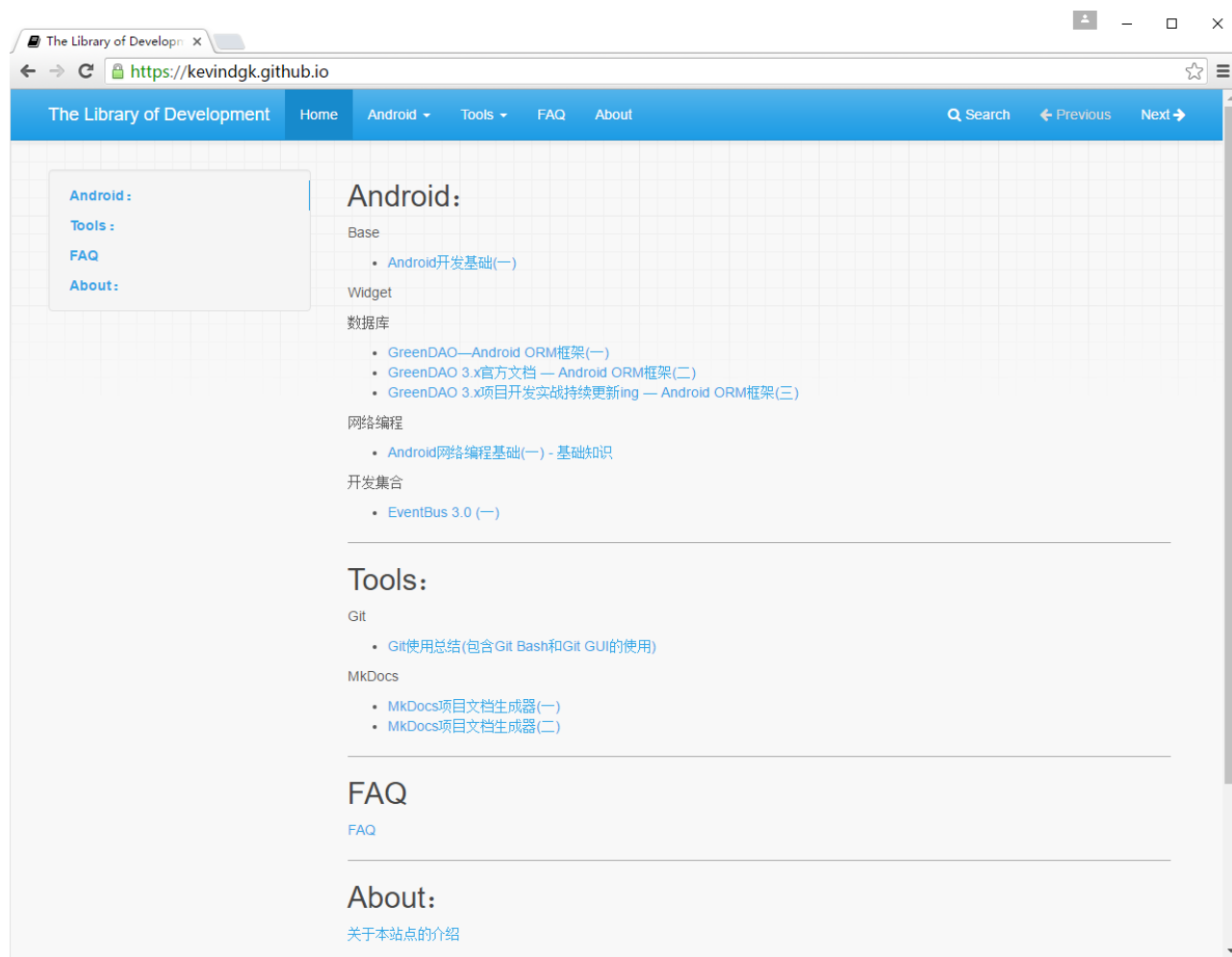
4.创建一个index.html，放到clone到本地的仓库中，然后推到github上。

5.测试一下，访问<https://kevindgk.github.io/>，如果能够访问的到，那么恭喜你，完成了一大半！

6.接下来就是将mkdocs文档生成的site文件夹下的所有内容复制到本地的仓库中，然后推到github上。这个时候，如果再次访问上面的地址，默认就会打开仓库的根目录下的

index.html这个网页。恭喜你，大功告成！

下面是小编丰富后的界面：



编写文档

官方地址：<http://www.mkdocs.org/user-guide/writing-your-docs/>

中文文档：http://markdown-docs-zh.readthedocs.io/zh_CN/latest/user-guide/configuration/

其实简单的查看官方文档，很容易看出来如何配置整个站点，本文档给出大部分的翻译，具体的请查看文档。

- 配置页面和导航栏

在mkdocs.yml配置文件中定义的页面才会被mkdocs创建，然后显示在导航栏上。

简单的配置页面：

```
1 pages:
2 - Home: 'index.md'
3 - About: 'about.md'
4 1
5 2
6 3
```

- 多级导航栏

按照下面的代码创建的子选项会被以列表的形式，显示在所属的导航条下：

```
1 pages:
2 - Home: 'index.md'
3 - User Guide:
4   - 'Writing your docs': 'user-guide/writing-your-docs.md'
5   - 'Styling your docs': 'user-guide/styling-your-docs.md'
6 - About:
7   - 'License': 'about/license.md'
8   - 'Release Notes': 'about/release-notes.md'
9 1
10 2
11 3
12 4
13 5
14 6
15 7
16 8
```

- 文件的路径

如果MarkDown文件是在一个site中，文档的URL就是文档的路径。比如，文档的路径如下：

```
1 docs/
2   index.md
3   user-guide/getting-started.md
4   user-guide/configuration-options.md
5   license.md
6 1
7 2
8 3
9 4
10 5
```

getting-started.md这个文档的路径就是user-guide/getting-started.md，默认根目录就是docs。

这样的话，就可以在一个文档中链接另外一个文档了，即可以从文档A中打开文档B，见下面的内容。

- 内部链接

从一个文档中打开另外一个文档，

```
1 [如何开始构建文档](user-guide/getting-started.md)
2 1
```

其实最后会被转化成从一个网页打开另外一个网页。

- 显示图片

路径如下：

```
1 mkdocs.yml
2 docs/
3 CNAME
```

```
4      index.md
5      about.md
6      license.md
7      img/
8          screenshot.png
9  1
10 2
11 3
12 4
13 5
14 6
15 7
16 8
```

在MarkDown中写法如下，其实这个就是markdown的标准语法，圆括号中的就是图片的地址，可以是本地的地址，也可以是网络的地址：

```
1 ![Screenshot](img/screenshot.png)
2 1
```

- Markdown语法扩展

(此处省略一万字。。。不过还是可以推荐一个十分不错的MarkDown软件Typora)

样式

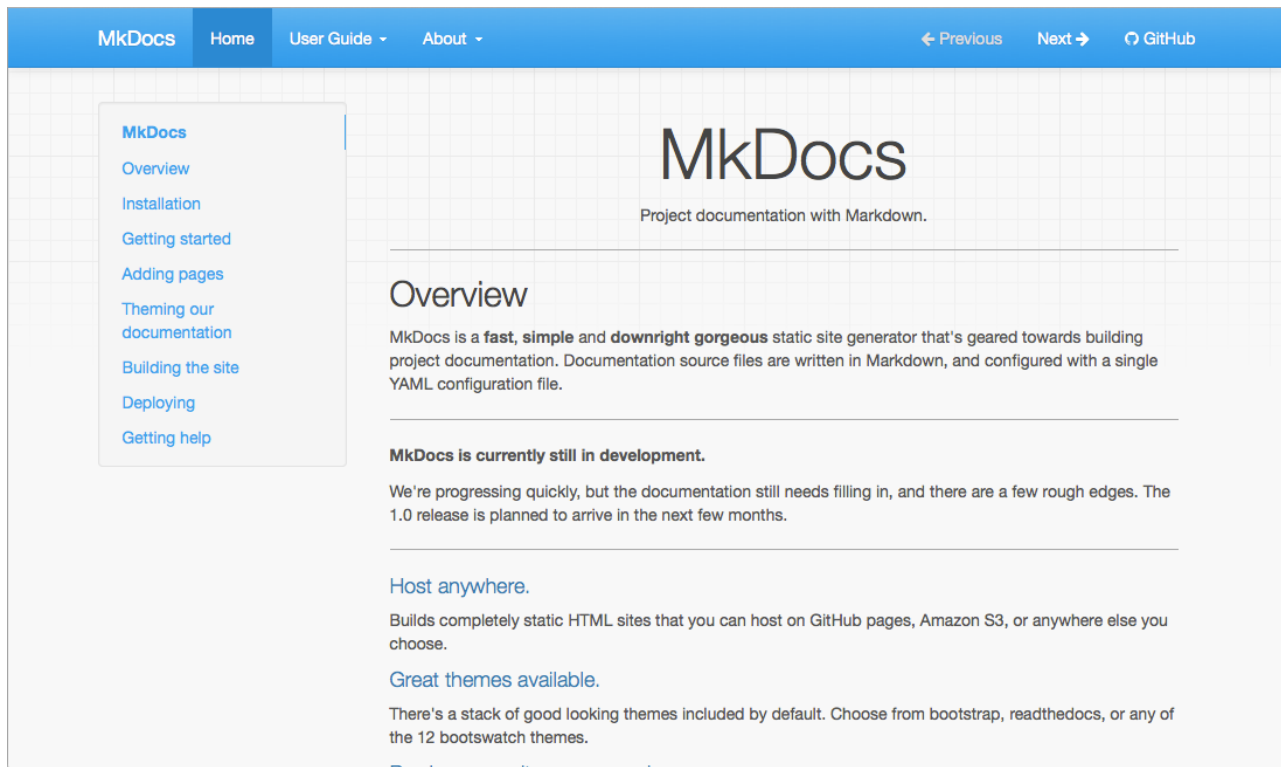
MkDocs包含许多不同的内置的样式和扩展的样式，也可以很容易的实现个性定制。

想要使用内置的样式，只要在配置文件中写入下列代码：

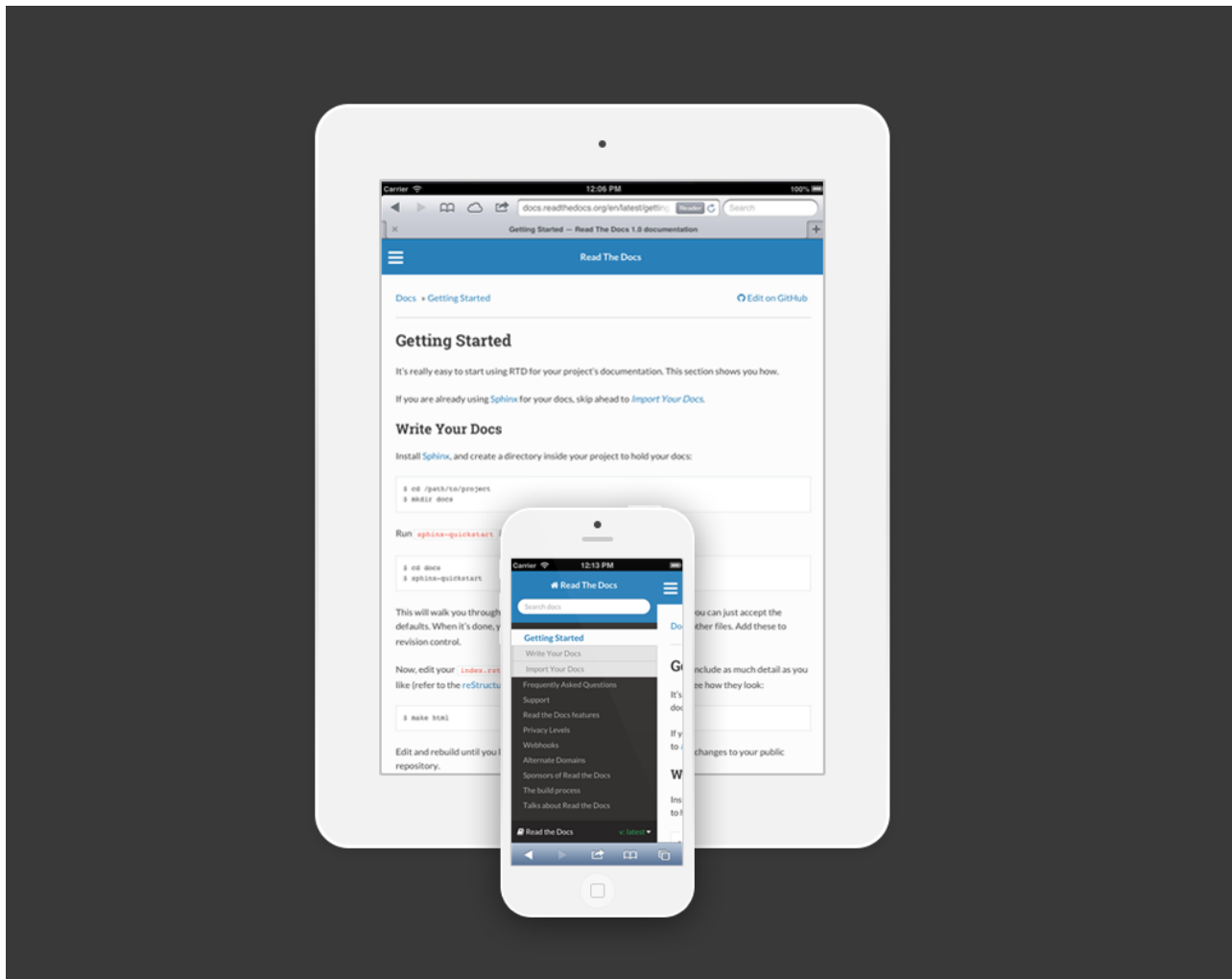
```
1 theme: readthedocs
2 1
```

theme: 样式的名称

- mkdocs



- readthedocs



(下面这几个主题需要安装，因为不会附带，但是安装的时候提示我升级，升级的时候提示我错误，所以小编忽略了，重点在文档的内容，主题什么的以后再说吧！)

配置

下面给出部分主要的配置信息：

项目信息

- `site_name` 站点的名称，这个配置是必须的，并且会显示在网页的顶部。

```
1 site_name: The Library of Development
2 1
```

- site_description 站点的描述

(配置信息太多了，可以查看中文文档的翻译，然后从官方网站上复制代码)

来源: <http://blog.csdn.net/kevindgk/article/details/52419819>