

目录

简介	1.1
安装	1.2
创作	1.3
命令	1.4
配置	1.5
插件	1.6
转PDF	1.7

- [简介](#)

简介

GitBook 是一个基于 Node.js 的命令行工具，可使用 Github/Git 和 Markdown 来制作精美的电子书，GitBook 并非关于 Git 的教程。

GitBook支持输出多种文档格式：

- 静态站点：GitBook默认输出该种格式，生成的静态站点可直接托管搭载Github Pages服务上；
- PDF：需要安装gitbook-pdf依赖；
- eBook：需要安装ebook-convert；
- 单HTML网页：支持将内容输出为单页的HTML，不过一般用在将电子书格式转换为PDF或eBook的中间过程；
- JSON：一般用于电子书的调试或元数据提取。使用GitBook制作电子书，必备两个文件：README.md和SUMMARY.md

- 安装GitBook
 - 1、环境要求
 - 2、安装命令行工具 `gitbook-cli`
 - 3、安装编辑器`gitbook-editor`
 - 1)、windows用户
 - 2)、mac用户

安装GitBook

1、环境要求

GitBook 是一个基于 Node.js 的命令行工具，所以要确保环境中已安装 Node.js.

检验Node.js是否安装成功

```
$ node -v  
v12.11.1
```

2、安装命令行工具 `gitbook-cli`

通过npm命令来安装gitbook-cli， 执行命令：

```
$ npm install gitbook-cli -g
```

上面命令运行完成之后，你可以使用下面的命令来检验是否安装成功。

```
$ gitbook -V  
CLI version: 2.3.2  
Installing GitBook 3.2.3
```

3、安装编辑器`gitbook-editor`

- 1)、windows用户

```
$ choco install gitbook-editor
```

- 2)、mac用户

```
$ brew install gitbook-editor
```

- 创作
 - 1、新建一本书 `gitbook init`
 - 2、编排书的目录结构
 - 3、编辑书的内容
 - 4、预览全貌

创作

GitBook用于创作电子书，主要有以下几个步骤：

1、新建一本书 `gitbook init`

为新书创建一个目录：`gitbook_tutorial`，进入`gitbook_tutorial`目录，初始化。

```
$ mkdir gitbook_tutorial
$ cd gitbook_tutorial
$ gitbook init
warn: no summary file in this book
info: create README.md
info: create SUMMARY.md
info: initialization is finished
```

运行`gitbook init`命令会创建 `README.md` 和 `SUMMARY.md` 这两个文件，`README.md` 是说明文档，`SUMMARY.md`是书的章节目录，其默认内容如下所示：

```
# Summary

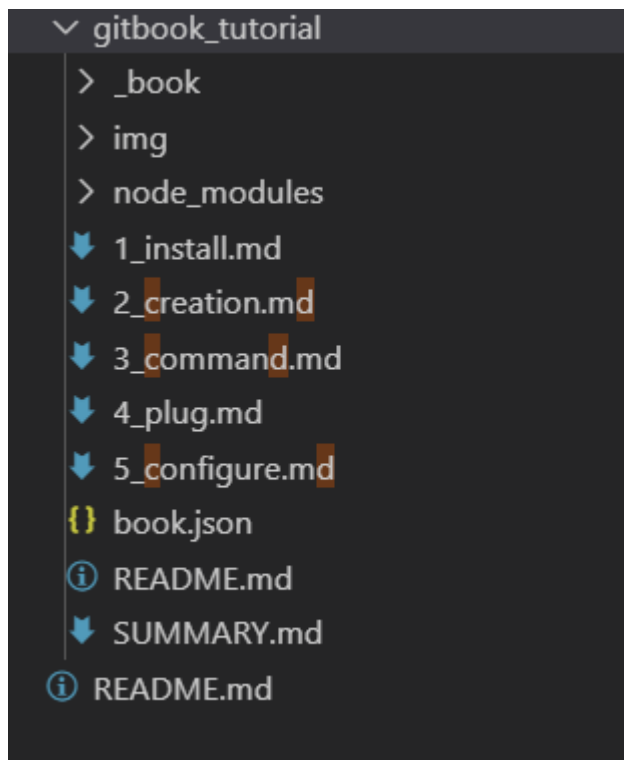
* [Introduction](README.md)
```

2、编排书的目录结构

进入目录`gitbook_tutorial`，在其中创建下列文件。

```
$ npx -p touch nodetouch 1_install.md
$ npx -p touch nodetouch 2_creation.md
$ npx -p touch nodetouch 3_command.md
$ npx -p touch nodetouch 4_plug.md
$ npx -p touch nodetouch 5_configure.md
```

如下图：



编辑SUMMARY.md文件：

```
# Summary

* [简介](README.md)
* [安装](1_install.md)
* [创作](2_creation.md)
* [命令](3_command.md)
* [插件](4_plug.md)
* [配置](5_configure.md)
```

3、编辑书的内容

以README.md为例：

简介

GitBook 是一个基于 Node.js 的命令行工具, 可使用 Github/Git 和 Markdown 来制作精美的电子书, GitBook 并非关于 Git 的教程。

GitBook支持输出多种文档格式:

- 静态站点: GitBook默认输出该种格式, 生成的静态站点可直接托管搭载Github Pages服务上;
- PDF: 需要安装gitbook-pdf依赖;
- eBook: 需要安装ebook-convert;
- 单HTML网页: 支持将内容输出为单页的HTML, 不过一般用在将电子书格式转换为PDF或eBook的中间过程;
- JSON: 一般用于电子书的调试或元数据提取。

使用GitBook制作电子书, 必备两个文件: README.md和SUMMARY.md

4、预览全貌

在终端输入

```
$ gitbook serve
```

然后在浏览器地址栏中输入 <http://localhost:4000> 便可预览书籍。

如下图:



- 命令
 - 1、查看gitbook的帮助信息
 - 2、启动服务器生成静态网页预览
 - 3、生成静态网页而不开启服务器
 - 4、生成静态网页时指定gitbook的版本，如果本地没有将先下载
 - 5、列出所有的gitbook版本
 - 6、列出远程可用的gitbook版本
 - 7、更新到gitbook的最新版本
 - 8、卸载对应的gitbook版本
 - 9、安装依赖
 - 10、指定log的级别
 - 11、输出错误信息

命令

GitBook常用的一些命令

1、查看gitbook的帮助信息

```
$ gitbook --help
```

2、启动服务器生成静态网页预览

```
$ gitbook serve
```

3、生成静态网页而不开启服务器

```
$ gitbook build
```

4、生成静态网页时指定gitbook的版本，如果本地没有将先下载

```
$ gitbook build --gitbook=3.2.3
```

5、列出所有的gitbook版本


```
$ gitbook ls
```

6、列出远程可用的gitbook版本

```
$ gitbook ls-remote
```

7、更新到gitbook的最新版本

```
$ gitbook update
```

8、卸载对应的gitbook版本

```
$ gitbook uninstall 3.2.3
```

9、安装依赖

主要指在 book.json 中加入的配置插件

```
$ gitbook install ./
```

10、指定log的级别

```
$ gitbook build --log=debug
```

11、输出错误信息

```
$ gitbook builid --debug
```

- 配置
 - 1、创建配置文件book.json
 - 2、编辑配置文件book.json
 - 3、配置文件book.json中各属性介绍

配置

1、创建配置文件book.json

进入gitbook_tutorial目录，创建一个文件 book.json

```
$ npx -p touch nodetouch book.json
```

2、编辑配置文件book.json

book.json文件主要用来存放配置信息，内容如下：

```

{
  "title": "GitBook简明教程",
  "author": "JackYang",
  "description": "GitBook的安装, gitbook-editor的安装及简单使用",
  "language": "zh-hans",
  "gitbook": "3.2.3",
  "styles": {
    "website": "./styles/website.css",
    "ebook": "./styles/ebook.css",
    "pdf": "./styles/pdf.css",
    "mobi": "./styles/mobi.css",
    "epub": "./styles/epub.css"
  },
  "structure": {
    "readme": "README.md"
  },
  "links": {
    "sidebar": {
      "GitBook简明教程": "https://legacy.gitbook.com/book/j
    }
  },
  "plugins": [
    "-sharing",
    "splitter",
    "expandable-chapters-small",
    "anchors",
    "github",
    "github-buttons",
    "donate",
    "sharing-plus",
    "anchor-navigation-ex",
    "favicon",
    "code"
  ],
  "pluginsConfig": {
    "code": {
      "copyButtons": false
    },
    "github": {
      "url": "https://github.com/JackYang3567"
    },
    "github-buttons": {
      "buttons": [{
        "user": "JackYang",
        "repo": "glory",
        "type": "star",
        "size": "small",
        "count": true
      }

```

```

    }
  ]
},
"donate": {
    "wechat": "./src/img/微信打赏二维码.jpg",
    "alipay": "./src/img/支付宝打赏二维码.jpg",
    "title": "",
    "button": "赞赏",
    "alipayText": "支付宝打赏",
    "wechatText": "微信打赏"
  },
"sharing": {
  "douban": false,
  "facebook": false,
  "google": false,
  "hatenaBookmark": false,
  "instapaper": false,
  "line": false,
  "linkedin": false,
  "messenger": false,
  "pocket": false,
  "qq": false,
  "qzone": false,
  "stumbleupon": false,
  "twitter": false,
  "viber": false,
  "vk": false,
  "weibo": false,
  "whatsapp": false,
  "all": [
    "google", "facebook", "weibo", "twitter",
    "qq", "qzone", "linkedin", "pocket"
  ]
},
"anchor-navigation-ex": {
  "showLevel": false
},
"favicon":{
  "shortcut": "./src/img/favicon.jpg",
  "bookmark": "./src/img/favicon.jpg",
  "appleTouch": "./src/img/apple-touch-icon.jpg",
  "appleTouchMore": {
    "120x120": "./src/img/apple-touch-icon.jpg",
    "180x180": "./src/img/apple-touch-icon.jpg"
  }
}
}

```

```
}  
}
```

3、配置文件book.json中各属性介绍

- 1、**title** 本书标题
- 2、**author** 本书作者
- 3、**description** 本书描述
- 4、**language** 本书语言，中文设置 **"zh-hans"** 即可

language": "zh-hans"

可选的语言如下：

```
en, ar, bn, cs, de, en, es, fa, fi, fr, he, it, ja,  
ko, no, pl, pt, ro, ru, sv, uk, vi, zh-hans, zh-tw
```

- 5、**gitbook** 指定使用的 **GitBook** 版本
- 6、**styles** 自定义页面样式

默认情况下各generator对应的css文件

```
"styles": {  
  "website": "styles/website.css",  
  "ebook": "styles/ebook.css",  
  "pdf": "styles/pdf.css",  
  "mobi": "styles/mobi.css",  
  "epub": "styles/epub.css"  
}
```

- 7、**structure** 指定 **Readme**、**Summary**、**Glossary** 和 **Languages** 对应的文件名
- 8、**links** 在左侧导航栏添加链接信息

```
"links": {  
  "sidebar": {  
    "GitBook简明教程": "https://legacy.gitbook.com/book/  
  }  
},
```

• 9、plugins 配置使用的插件

可以在插件前面加-符号删除默认插件，默认五种插件如下，更多插件

- highlight: 代码高亮
- search: 导航栏查询功能（不支持中文）
- sharing: 右上角分享功能
- font-settings: 字体设置（最上方的"A"符号）
- livereload: 为GitBook实时重新加载

• 10、pluginsConfig 配置插件的属性

- 插件
 - 1、GitBook 插件列表
 - `back-to-top-button`（返回顶部）
 - `code`（代码添加行号&复制按钮）
 - `search-pro`（高级搜索，支持中文）
 - `github`（在右上角添加github图标）
 - `splitter`（侧边栏宽度可调节）
 - `tbfed-pagefooter`（页面添加页脚，简单的）
 - `page-copyright`（页面页脚版权，复杂的）
 - `donate`（打赏插件）
 - `sharing-plus`（分享当前页面）
 - `custom-favicon`（修改标题栏图标）
 - `prism`（代码高亮）
 - `todo`（复选框）
 - `pageview-count`（阅读量计数）
 - `auto-scroll-table`（表格滚动条）
 - `image-captions`（显示图片名称）
 - `styles-sass`（使用sass替换css）
 - `styles-less`（使用less替换css）
 - `toggle-chapters`（目录折叠）
 - `multipart`（分章节展示）
 - 设置导航序号
 - 2、添加插件
 - 3、安装插件
 - 1）、执行gitbook install
 - 2）、npm安装
 - 3)、从GitHub下载源码，放到node_modules文件夹里

插件

1、GitBook 插件列表

- **back-to-top-button**（返回顶部）
- **code**（代码添加行号&复制按钮）
- **search-pro**（高级搜索，支持中文）
- **github**（在右上角添加github图标）
- **splitter**（侧边栏宽度可调节）

- **tbfed-pagefooter**（页面添加页脚，简单的）
- **page-copyright**（页面页脚版权，复杂的）
- **donate**（打赏插件）
- **sharing-plus**（分享当前页面）
- **custom-favicon**（修改标题栏图标）
- **prism**（代码高亮）
- **todo**（复选框）
- **pageview-count**（阅读量计数）
- **auto-scroll-table**（表格滚动条）
- **image-captions**（显示图片名称）
- **styles-sass**（使用**sass**替换**css**）
- **styles-less**（使用**less**替换**css**）
- **toggle-chapters**（目录折叠）
- **multipart**（分章节展示）
- 设置导航序号

2、添加插件

在book.json中配置如下（这里用code插件作为例子）：


```
{
  "plugins": [
    "code"
  ],
  "pluginsConfig": {
    "code": {
      "copyButtons": false
    }
  }
}
```

3、安装插件

可以选择下列三种方式中的一种进行安装

• 1)、执行gitbook install

这其实是安装book.json中的配置

```
$ gitbook install ./
```

• 2)、npm安装

命令格式npm install gitbook-plugin-插件名字

```
$ npm install gitbook-plugin-code
```

• 3)、从GitHub下载源码，放到node_modules文件夹里

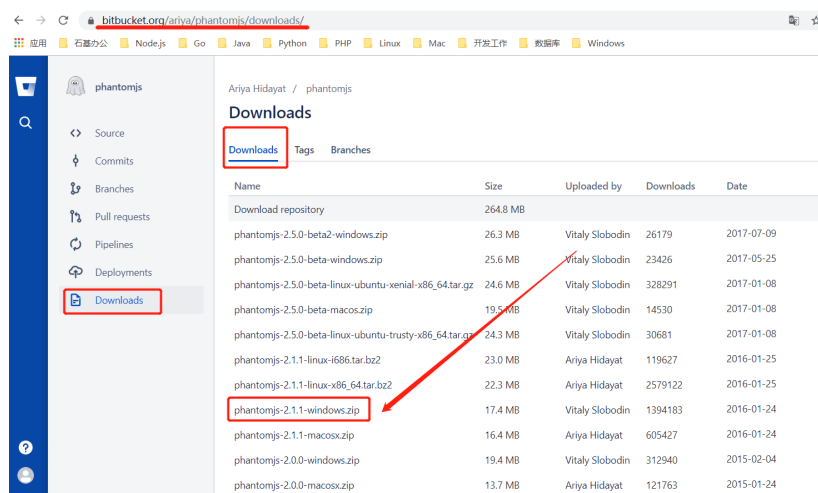
- GitBookI转成PDF
 - 1、下载相关插件
 - 2、安装插件
 - 1、phantomjs只需解压，并将解压后的目录添加到环境变量
 - 2、运行calibre安装程序，并将安装目录添加到环境变量
 - 3、用npm全局安装gitbook-pdf
 - 3、生成PDF文件

GitBookI转成PDF

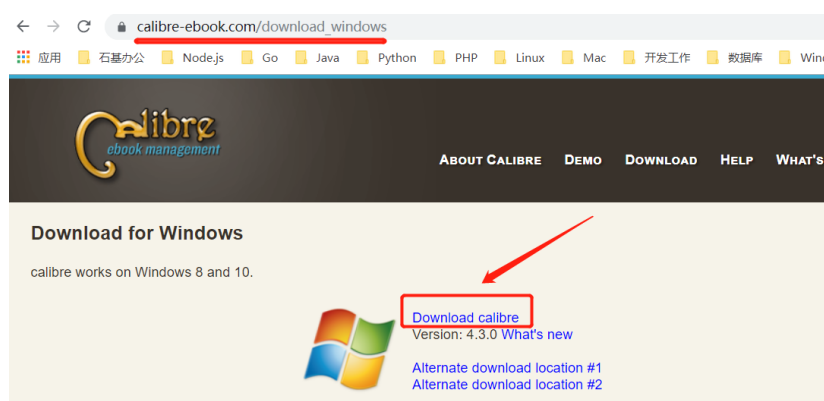
需要下载两个插件 1、phantomjs； 2、calibre

1、下载相关插件

- 下载phantomjs : <https://bitbucket.org/ariya/phantomjs/downloads/>



- 下载calibre: https://calibre-ebook.com/download_windows



2、安装插件

1、phantomjs只需解压，并将解压后的目录添加到环境变量

2、运行calibre安装程序，并将安装目录添加到环境变量

3、用npm全局安装gitbook-pdf

```
$ npm install gitbook-pdf -g
```

3、生成PDF文件

切换到你之前自己写的电子书文档目录，执行如下命令：

```
$ gitbook pdf ./pdffilename
```