**Sphinx+vscode+github+readthedocs环境搭建**

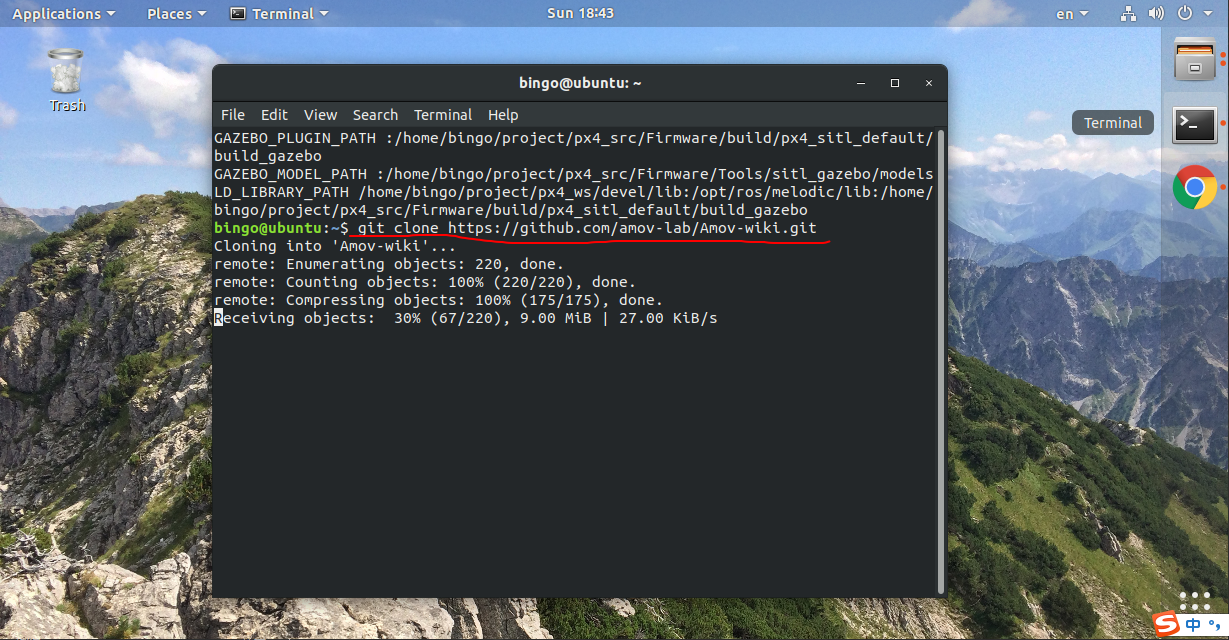
该文档的写作方式和Python官网文档是一致的，只是风格不同，所以使用的一些插件都是与Python相关的。该文档使用.rst编写的，类似于Markdown，语法上稍有不同。相比于Markdown，该文档能够保证本地于公布到网上的wiki一致性，方便多人共同协作，个人编写完成的文档，在本地可以先编译查看是否显示格式有问题，确定无误之后，再进行上传GitHub。上传至GitHub进行托管，一是解决图片存放问题，而是方便多人共同维护，能够保证团队之间的本地文档是一致的。

**Ubuntu环境搭建下：**

1. 下载vscode，并安装

<https://code.visualstudio.com/Download>

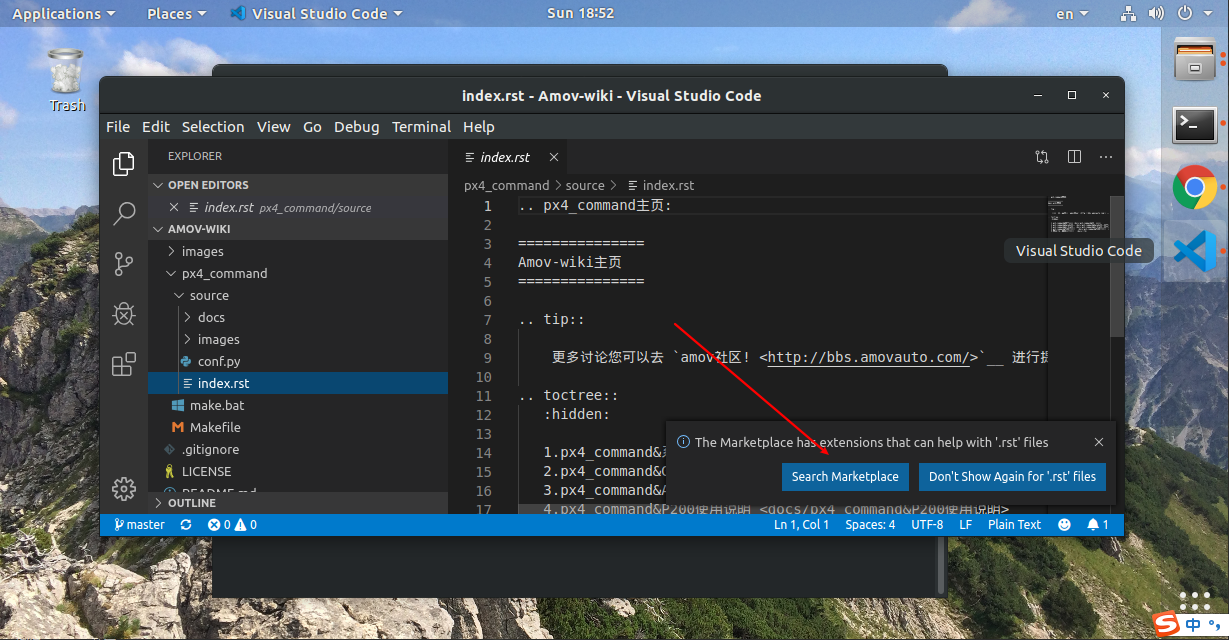
1. 下载wiki GitHub源码，如下图所示：

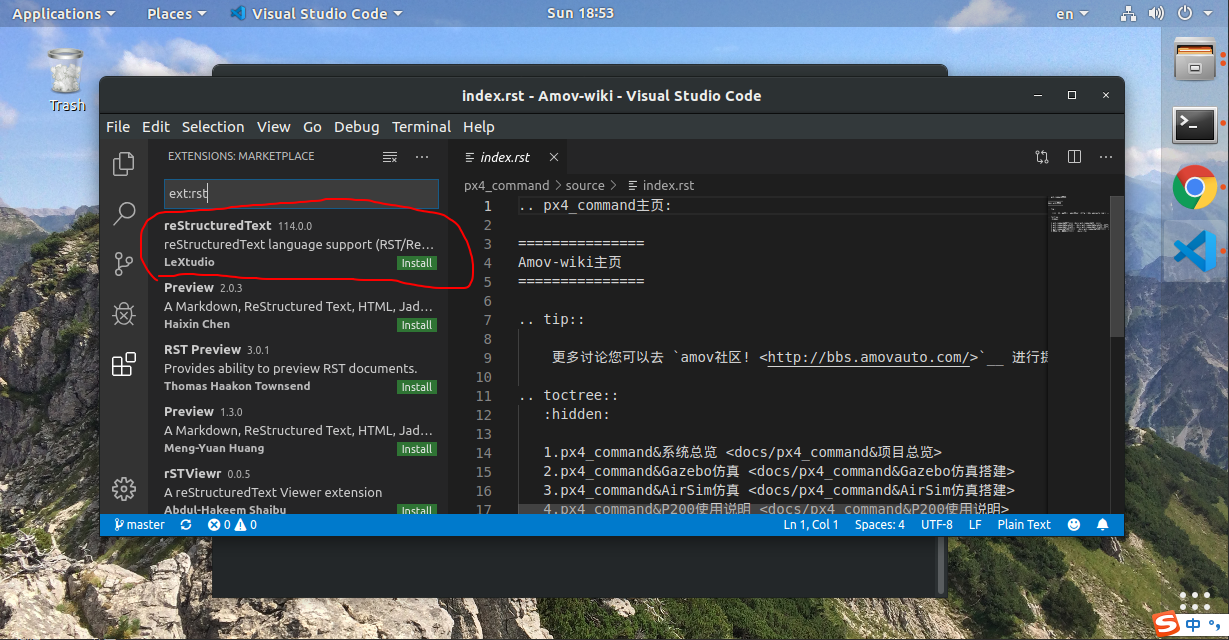


下载完成之后，用vsocde打开该文件夹。

1. 安装相关插件

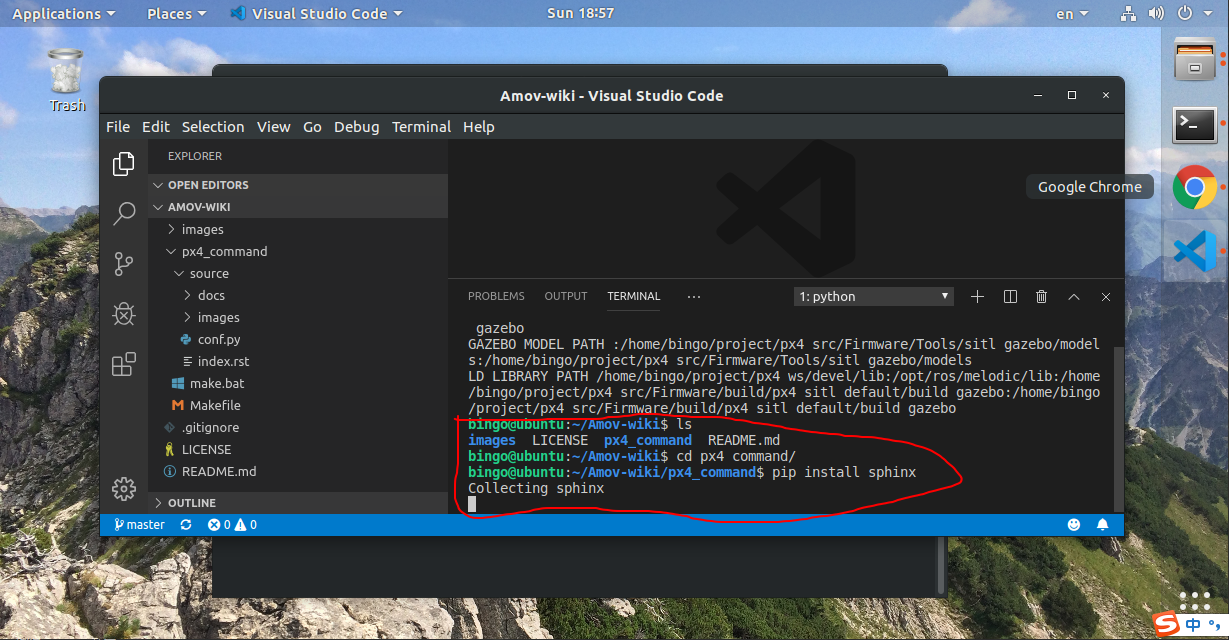
自动寻找.rst相关插件



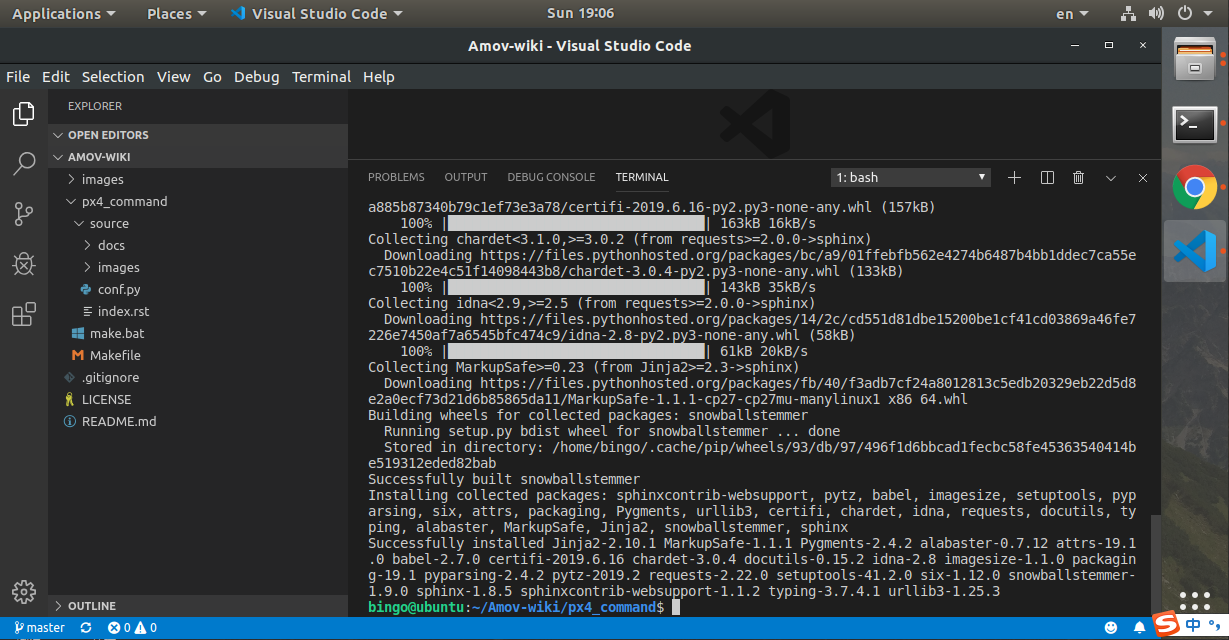


直接安装上图所示插件。

打开终端，进入git库下面的px4\_command文件目录，执行pip install sphinx



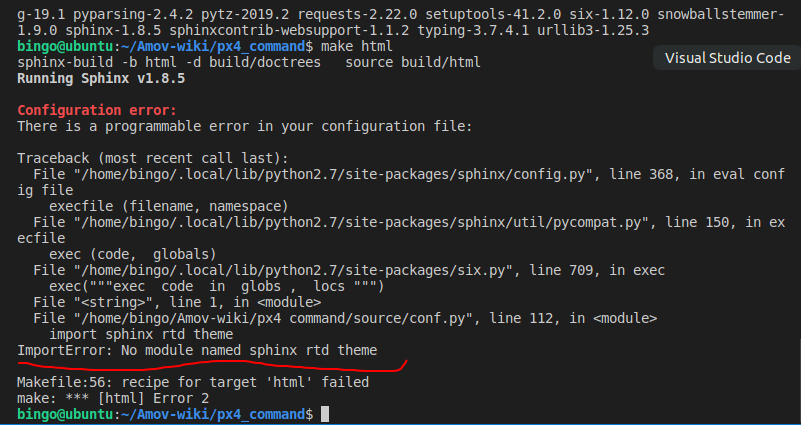
安装完成之后如下图：



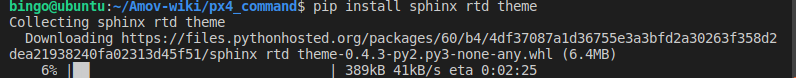
1. 本地编译查看所编写wiki样式

在px4\_command下执行 make html 生成本地网页，

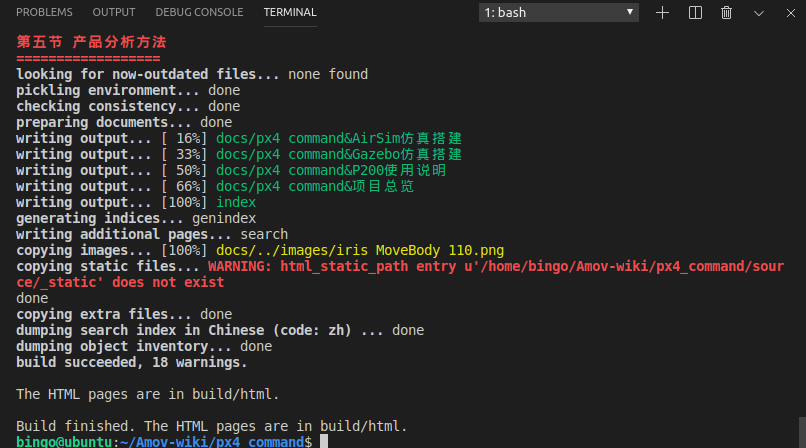
出现如下错误，显示缺少sphinx\_rtd\_theme，我们开始安装该主题



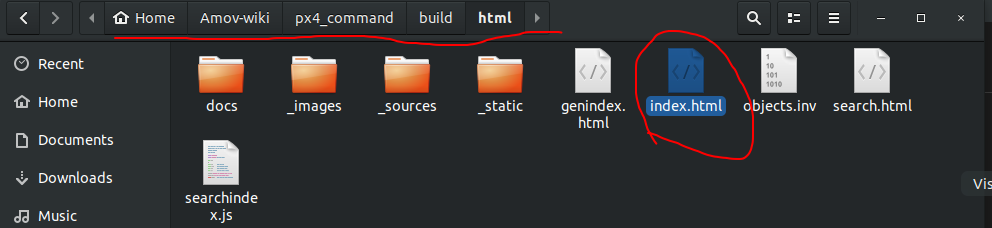
执行pip install sphinx\_rtd\_theme



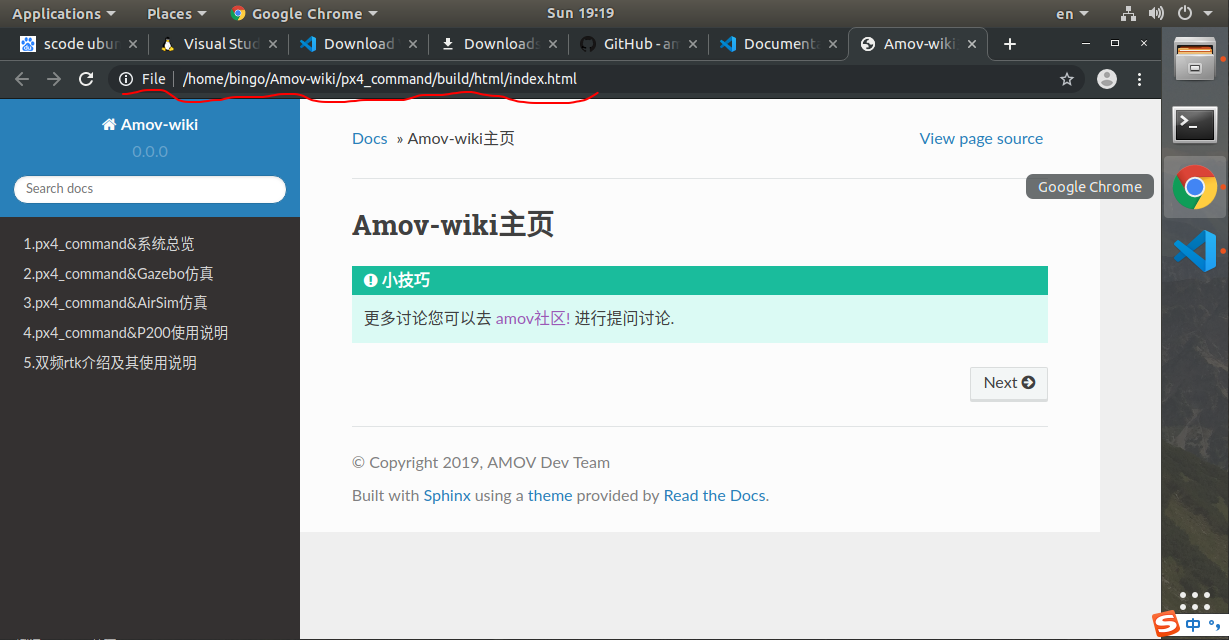
安装完成之后，再次执行make html



会有警告出现，是因为部分语法有点小瑕疵。可以看到生成的html已经出现在build目录下。



然后就是打开html文件下的index.html

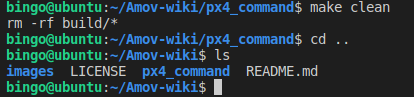


这样就可以查看你在本地修改之后的预览了。如果预览和你写的东西没有什么问题，就可以准备提交到GitHub上进行托管，在push到GitHub之前，需要删除掉刚才本地编译生成的预览。还是在px4\_command文件目录下执行，make clean



会看到，已经删除掉build文件夹。

1. 上传至GitHub托管



进入Amov-wiki目录下，进行上传

git add .

git commit –m “”

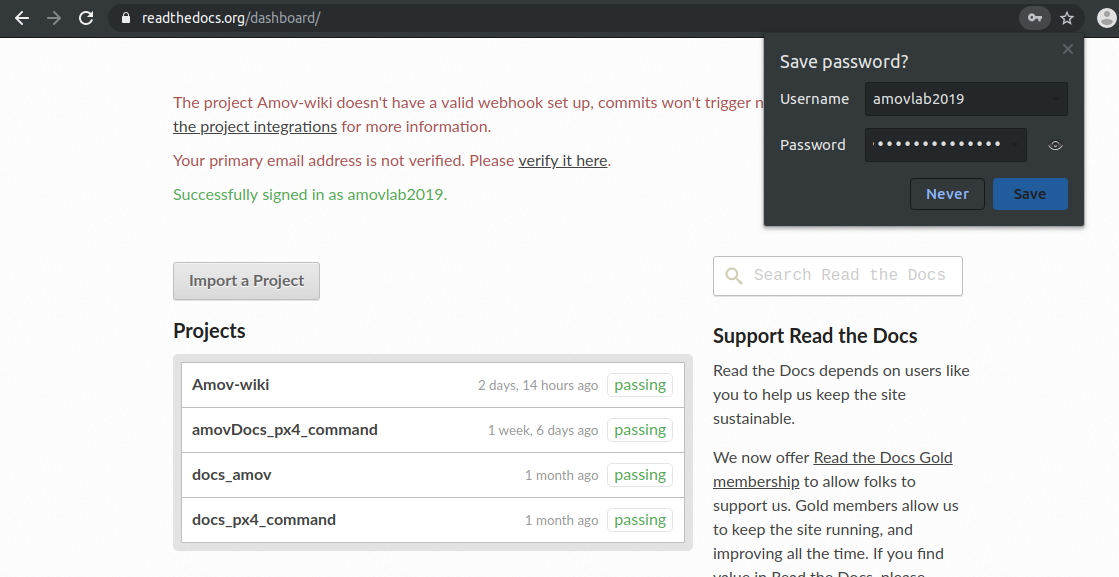
git push

每次更新源码的时候git pull即可实现同步最新的本地文档

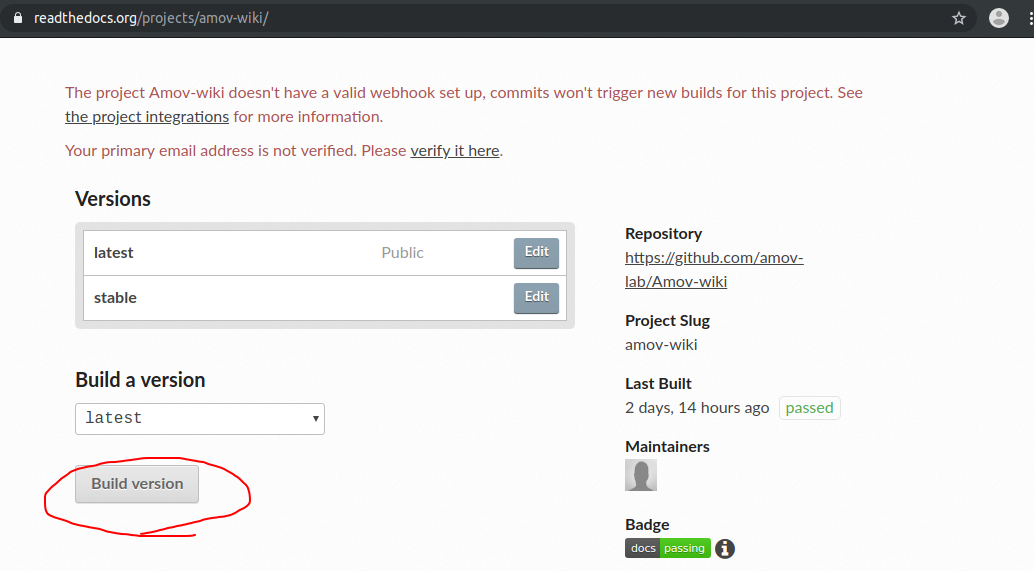
1. 在readthedocs上构建刚上传的文档，更新wiki上的显示。

登录账户是amovlab2019

密码是\*\*\*\*\*\*\*\*\*\*\*



登录之后可以看到有四个项目，我们与GitHub相关联的是第一个Amov-wiki



进入之后点击编译，等待编译构建完成，即可打开wiki官网查看我们的wiki，其效果和你在本地上查看的效果是一致的。

**Windows下环境搭建：**

基本步骤和Ubuntu下一样，有几点区别，

1、在Windows下需要安装Python，安装包放置到TOS下面的工具中，安装完之后需要添加环境变量，最后先测试Python是否安装成功。确保能够使用pip。

2、git安装包和vscode安装包均放在TOS里面，安装即可。

3、在Ubuntu下make html与make clean，但在Windows是.\make html和.\make clean

**最后说明：**

该方式编写文档具有一定逻辑性，建议大家命名图片时，命名为英文简写，方便好记。

目前的文档都是放在px4\_command文件下的，编写多个文件时，逻辑思路可能还不是很清晰，明了。之后可能继优化改文档的的规范性，合理性。

reStructuredText 语法可以参考： <https://3vshej.cn/rstSyntax/index.html>