

# 2020 Kust

受[NEUP-Net-Department 2018 Mengxin Plan](#)启发。

该repo用于帮助昆工学生掌握一些基础技能,整理了一些资料。

可以初步学习到以下基本技能:

- [2020 Kust](#)
- [todolist](#)
  - [git/github](#)
  - [escape GFW](#)
  - [build your own blog](#)
  - [Markdown](#)
  - [other](#)

## todolist

### git/github

Git 是一个代码版本控制工具。在真实的开发环境中,是多人写作开发的,没有代码控制工具几乎不可能完成任何开发工具,因此是每一个程序员的**必备技能**。合格的程序员应该把使用去代码控制工具管理自己的代码作为一种自然而然的事情。

以下给出了几个 Git 学习资源:

- <https://try.github.io/>
- 廖雪峰 [Git教程](#)

Q: 什么是 github ?

A: [wiki](#)上的解释是:

GitHub是通过Git进行版本控制的软件源代码托管服务平台。

根据在2009年的Git用户调查, GitHub是最流行的Git访问站点。截止到2015年, GitHub已经有超过两千八百万注册用户和5700万代码库。事实上已经成为了世界上最大的代码存放网站和开源社区。

github 是程序员的社区, 开源的社区, 富含宝贵的学习资源。

有这样一种说法, 一个程序员的 GitHub 帐号相当于这个程序员的门面, 一个好的 GitHub 帐号是无声而最有力的简历和推荐信, 而事实上你注册的 GitHub 帐号 (如不出意外) 将会陪伴你整个代码生涯, 所

以请慎重地为自己选择一个 GitHub ID，然后给自己上传一个合适的头像（也可以使用 GitHub 自动为你生成的像素风头像）。快把使用 `github` 作为你的日常活动吧！相信通过使用这个网站能够充分拓展你的视野。

- [如何使用github-知乎](#)
- [GitHub Learning Lab](#), `github` 的官方教程。

建议的实践：

完成注册 GitHub 之后，请学习并完成有关 Git/GitHub 的命令行教程。同时你也可以参考网络上各种优质的 Git/GitHub 学习资料，自学 Git 版本控制系统，自学有关如何在 GitHub 上进行协作的内容。

## escape GFW

为什么要科学上网？

“You shouldn’t let a wall decide what you know.”

如果想做程序员，至少得把这个技能点到够用，不会因为被限制，就视野狭隘，技术文档第一线的东西都应该到外网去找。

首先了解下什么是[GFW](#)。以及一些基本的概念[浅谈vpn、vps、Proxy以及shadowsocks之间的联系和区别](#)

给出自建VPS的几个链接，同样的，若对给出内容不满意，务必自行搜索：

- [使用酸酸乳方式](#)
- [新一代科学上网利器：V2Ray扫盲教程](#)
- [V2Ray白话文教程](#)

如果不想折腾，推荐租别人搭好的，若需要请私聊。

建议的实践：

能够成功访问到 `www.google.com`，创建一个自己的谷歌账号。浏览只能在外网浏览的优秀社区如 `StackOverflow`。

## build your own blog

Q: 为什么要自建个人博客，`csdn` 它不好吗？

A: 对自己的查漏补缺、积累经验，交友发展到自己的技术圈子。

相比于 `csdn` 等等的博客平台，个人自建博客自主权比较高，而博客平台的话会有这样那样的限制。

最为重要的是，个人博客记录自己成长过程。想融入程序员的圈子至少得有自己的**明信片**，而个人的博客和github主页就是这样的证明。

那么一个个人好的博客具备哪些功能和特质呢？

- 必须能够支持 Markdown 语法
- 推荐能够支持 MathJax，拓展链接：[什么是MathJax](#)
- 能够方便的更新和管理博客内容，例如通过 git 管理
- 方便备份和部署
- 页面清爽、干净，看着博客不会觉得累
- 排版合理

满足上述的博客搭建方式有很多，希望能够自己思考并多折腾折腾。随便google搜索会得到例如 [wordpress](#) 等等的搭建方式。

其中，比较推荐和方便的是 [github pages](#) 提供的博客托管服务。而 [github](#) 官方默认的渲染框架是 [jekyllrb](#)。

以下给出几个教程：

- [Github Pages + jekyll 全面介绍极简搭建个人网站和博客](#)
- [极客学院](#)
- [知乎：github上利用jekyll搭建自己的blog的操作顺序？](#)

给出几个好看的模板：

- [码志](#)，预览效果：<https://mazhuang.org/>
- [Hux Blog](#)，预览效果：<http://huangxuan.me/huxblog-boilerplate/>
- [Agency Jekyll theme](#)，预览效果：<https://y7kim.github.io/agency-jekyll-theme/>

如果对上面给出的模板不满意，可以使用[jekyllthemes](#)或者[github](#)搜索其他模板，

建议的实践：

搭建一个属于自己的博客，并且大方地向朋友展示！

## Markdown

Q: 什么是[Markdown](#)？

A: Markdown是一种轻量级标记语言，创始人约翰·格鲁伯（英语：John Gruber）。它允许人们“使用易读易写的纯文本格式编写文档，然后转换成有效的XHTML（或者HTML）文档”。这种语言吸收了很多在电子邮件中已有的纯文本标记的特性。

Q: 为什么要使用 Markdown ?

A: 由于其具有轻量化、易读易写特性，是大部分项目文档、个人博客甚至是小说的首选编辑"语言"。

Q: Markdown 语法的通用性？

A:

- 知乎
  - SourceForge
  - StackOverflow
  - v2ex
- 等等，主流技术的社区几乎都支持 Markdown 语法进行评论、撰写博客。

Q: 如何掌握？

A:

- [Markdown命令查询手册](#)
- [Markdown 基本语法](#)

Q: 在哪里写 Markdown （用什么软件编辑 Markdown ）？

A:

1. 直接在上述的支持网页中评论使用即可。
2. vscode 中编辑 .md 文件，使用插件[Markdown Preview Enhanced](#)进行阅览。推荐！
3. MarkdownPad
4. [vnote](#),强烈推荐！

建议的实践：

写一些 Markdown ,如果已经完成了博客搭建，尝试用 Markdown 写点东西然后发布到博客上。

## other

很多时候，在 Google 上查找了大量的资料，也查阅了 Github 上 issue 诸多问题，也无法得到有效的解决办法。此时就需要在某个社区/群聊/issue上提出新的问题。为了自己的问题有较大概率得到回复、为被提问者提供尽可能多的有效信息以及保持有效的沟通方式，就需要掌握提问的技巧，重视提问的技巧，往往能够得到其他人友好的帮助。

- [提问的艺术](#)

