

# 快捷键

## 前言

这里只记录一些重要的，我把那些我忘记的快捷键也删掉吧。

发现了 quicker 一些有趣的操作，可以选择打开软件，还可以选择激活已经打开的窗口，换言之，这样激活窗口就可以快捷键操作了。

我打算，至少常见的两个命令行，我希望直接快捷键操作。

## 左键

右键 复制，粘贴，很常用

Tap 回车，很好用

Space 粘贴，

大部分的和左键辅助的快捷键都和 `ctrl` 的效果相当，只是有时候拿着鼠标不方便使用 `ctrl`，再说几个比较特别的。

`c`，效果是，复制并且显示粘贴板上下文，需要长按，等一下，这个好像和 `ctrl + c` 效果还是一样，那就有一点乏味了。

那么，这样就把 `v` 放解放出来了，那么，`v` 用来做什么呢？因为 `v` 本来的功能已经被 `space` 实现了。想不到，就不去想了。

R 就打开 [zplan.md](#)

其实也没有什么，或许我可以选择默认打开就是 `zplan`，或许也不必要。如果默认就是 `zplan`，也好。还不能这样选择，也无妨。

没有什么其他特别的地方。

---

旧

1 ， 打开日志，对于目前来说也就是 11 月，可能就会到年底了，反正也没有多少事情

2 ， 打开快捷键，常看常新。

3 ， 全选复制，这样就省得自己弄两次了。 3 现在也是。

4 ， 全选粘贴，现在就完全了。

好荒唐啊，上面记录的四个数字键，其实我用得都不多，我感觉还是简单一点吧。就尽可能，让左键辅助的效果和 `Ctrl` 的效果相当。因为，全部都只是鼠标两个按键的组合，不需要再单独记忆。

## Capslk

`Caps` 需要很大的改动，参考pandoc的内容。也就是下面的[引号](#)。

另外就是其他按键的映射了，已经没有了 `A`，因为 `A` 用来做修饰键。

不同于左键，`Caps` 不需要承担 `Ctrl` 的功能，所以就更能够发挥快捷键的价值。

### 第一行

`Q`：智能识别，我一般默认视作链接，其他的东西我很少识别，还有就是搜索

`W`：也就是密码本，常用语，虽然我还没有发挥常用语真正厉害的功能

`R`：待设置。

`T`：待设置

`YUIOHKJL`，这一组就是用来文本操作的，仿照Vim的基础操作。

`P`：待设置

### 第二行

`S`：播放、暂停

`D`：Delete

`F`：Backspace

`G`：`Alt + F4`，你懂的，但是这个今天才设置，用得还不算熟练。

### 第三行

`Z`：`Win + 4`，我是把这个位置固定成了 Terminal，所以每一次都会打开PowerShell。

`X`：打开 **Flomo**的快捷键。(是的，我想明白了，不要使用“”表示强调，要使用加重符号)

`C`：也就是 **Quicker** 的设置。省得我自己去找。

`V`：待设置

`B`：待设置

N : IDEA, 从此我再也不需要点击IDEA就可以打开了。

M : Visual Studio Code,

毕竟, Capslk 实在是太强大, 它可以和26个字母组合成26个快捷操作, 还有标点符号, 还有数字, 组合太多了, 还有很多好玩的功能, 只不过暂时我还没有这样的需要。

## 引号

[别再用「六个点」当省略号了, 这些标点都有更规范的输入方式 - 少数派](#)

为什么会有这样的问题, 其实还是起源于前面那篇文章。看到了, 就过来折腾一下, 虽然也知道是完全不必要的事情。

这里是把其他的格式转换成 Markdown 的时候, 把直引号变成弯引号, 说起来, 这东西还是很难受。

提高效率, 不再折腾了, 也就是把符号好好弄一下。

我一度尝试在搜狗输入法中选择自定义标点, 但是, 显然, 我失败了, 这个东西是自带重置的效果, 每一次都会重置, 所以, 根本就无法持久化。最后, 我还是通过Quicker的扩展热点功能实现了引导的输入。

现在我可以手动输入各种引号了。

随便输入。

从今天起,  
我的「引用」就可以打破传统的限制,  
无论是“单引号”,  
还是‘双引号’,  
我都可以非常自在的输入,  
从前一直感觉 Capslk + A 用得很少  
现在才算是发挥出来它应该有的用途了。

我设置了一下, 还是Quicker非常好用。

原则上使用 Caps 当做引导键, 偶尔部分使用 A 作为修饰键, 虽然, 其实不是那么需要。

一些在中英文状态下可能会有歧义的按键, 默认加上 caps 就表示英文状态下的按键。

如,

/

``

</kbd>

\$

^

!

比较特别的是 8，因为这个不存在歧义，加上 Caps 就变成 ×，也就是标准的乘号，比 \*\* 要好看。

另外还有除号，这个我是没辙了，于是使用 A 作为修饰键，但是想着，或许也可以直接使用 9。

重要的当然是引号这一组。



Caps + 【】就是对应的「」，Caps + A + 【】就是『』，为了能够使用 Shift 修饰键，我也已经把它们的组合键去掉了。简简单单，快快乐乐。(这里我不太理解，当时输入 Caps + Shift 会变成默认的 Caps 的效果，但是现在又正常了，但是，我已经放弃使用 Shift 作为修饰键了，使用 A 也不错。)

普通的引号，就对应 ; 和 ‘ 这两兄弟。好吧，其实我现在还是有一点不太能够适应 Typora 的默认补全。

另外还有反引号，现在我改了一下，它就永远是反引号了！再也不需要受气了！

另外，还整的花活是←→，映射到了逗号和句号，现在感觉有一点浪费，它们应该映射成英文的标点。也不需要，我中文状态下就不要使用这两种符号了，但是 . 还是非常需要的，因为一些时候需要输入数字，就需要它。

其实选择 A 还是 Shift 作为修饰键都差不多，因为，默认的时候我使用左右手合作，我不会左手一次按下三个键，最多一个「引导键」+一个「修饰键」，所以之后搭配的一个键肯定在右边。

## Typora

### 基础快捷键

typora 用得挺多的，所以关于它的快捷键也可以记忆一些，比如说

左键辅助用得其实并不多，它最大的价值还是在，能够选中。其他的地方就无所谓了。

基本上也就是和 Ctrl 的那些常用的快捷操作。

Ctrl + Tab ，可以切换标签页，而且，切换之后，它还会记得光标的位置，挺好的。

但是，很多操作的快捷键也被别人占用了。所以，就这样吧。

`Ctrl + click` , 这个是新窗口打开。所以，就很适合分屏。

- 弹出标签的菜单选项：右键单击标签页，可以复制文件链接，适合在命令行操作。

多元文件搜索也是很常用很好用的功能。但是我还不会正则表达式，暂时也不会去学习的。快捷键不记得，但是可以有快速按钮选择。

`imageReviewer`: 图片查看器

右下角就有对应的按钮。

## **chineseSymbolAutoPairer: 中文符号自动补全**

功能：输入 《 【 《 ‘ “ 「 符号时自动补全。

确实能够补全，那么自己写的时候就注意一点就好了。

但是在代码块内不能补全。

所以，我还是启用吧，「」，我现在输入这些符号的能力强大得要死。

## **sortableOutline: 拖动大纲标题，调整文章结构**

| 此插件默认关闭，需手动开启。

我愿称之为最强，之前找不到。

## **resize\_table: 拖动调整表格大小**

功能：`ctrl+鼠标拖动`：修改表格的行高列宽。

## **resize\_image: 调整图片大小**

功能：`alt+鼠标滚轮滚动`：调整图片大小。

导出增强的用处实在是不大。

## **blockSideBySide: 并列显示活动块**

功能：活动块默认是上下显示，改为并列显示。

我现在知道为什么之前的显示会出现问题了，当时我修改了这个设置。

总之，结论就是，能够乱动就尽可能不要乱动。保持原有的按钮，

## **章节折叠**

- 折叠/展开单个章节：`ctrl+click`

- 折叠/展开父章节下所有同级的章节 : **ctrl+alt+click**
- 折叠/展开所有同级的章节 : **ctrl+shift+alt+click**

感觉也是很重要的功能。

## **resource\_manager：一键清除无用图片**

功能：资源管理，清除无用图片

使用方式：右键菜单 -> 少用插件 -> 资源管理

这个还挺好用的，里面什么都能看到，找到一个合适的文件夹，就可以这样高效清理。

不过，我发现了一点问题，它可以敏锐发现符号链接。

## **大纲**

toc，或者右侧点击，反正就我自己，我不是很喜欢。

## **commander：命令行环境**

功能：类似于 total commander 的命令行，一个快速执行命令的工具，并提供少量交互。

使用方式：右键菜单 -> 常用插件 -> 命令行环境

支持 shell：

- cmd/bash : Windows 或 Linux 的默认终端
- powershell : 微软的傻儿子 :D
- git bash : 需确保安装 Git Bash 并添加到环境变量
- wsl : 需确保安装 WSL2 并添加到环境变量

内置环境变量：

- \$f : 当前文件路径
- \$d : 当前文件的所属目录
- \$m : 当前挂载的根目录

蛮不错的，挺好玩的，就是我有更快捷的打开方式。



:sa:，还可以这样玩.....

## **scrollBookmarker：书签管理器**

使用方式：

1. 使用 alt+click 正文内容，打上书签。
2. 此时会自动调出书签管理器，点击上面的书签，即可跳转到书签。

显示/隐藏书签管理器：右键菜单 -> 常用插件 -> 二级插件 -> 书签管理器

## 图表

### echarts

既然支持echart和chart，那么我觉得就有必要好好学习，打开echart的官网吧。

另外，我也设置了一个快捷键，使用

echarts 就可以快速进入echarts编辑，然后再显示原文本，把对应的echarts代码放进来就可以了。

另外，搜狗输入法一些快捷操作我也应该记下来。

Typora 的快捷键我总是记不住，所以我想到了一个好方法，写一个文件，然后让 Typora 快捷打开，我真的是太聪明了。

## 光标跳转

我设置了，这也是一个快捷方式，使用 Alt + ← || → ，就可以跳转了，再也不怕文内跳转回不去了。