

1. 一步到位的 Alt + 点击

我们经常要在命令行中输入非常类似的命令，仅做一点点修改。如果要修改的位置是开头或末尾还好，可以通过快捷键快速定位。但如果要修改的位置位于一个很长的命令的中间位置就比较讨厌了，使用 **Alt + 左右箭头** 按单词跳跃也要按好多下，然后再按单个字符精确定位。

解决方法：**Alt + 鼠标点击**，一步到位！

2. pbcopy 和 pbpaste：连接命令行管道与剪切板

```
echo 'hello world' | pbcopy  
echo `pbpaste`
```

pbcopy 接收命令行中的标准输出作为剪切板的内容。

pbpaste 则将剪切版中的内容输出到标准输出。

3. 其它有用的快捷键

Ctrl + A：光标移动到行首

Ctrl + E：光标移动到行尾

Ctrl + U：删除光标所在位置之前的所有字符（不含当前位置字符）

Ctrl + K：删除光标所在位置之后的所有字符（含当前位置字符）

Ctrl + W：删除光标所在位置之前的一个单词

Ctrl + R：根据输入搜索以往使用过的命令

4. dot files

什么是 **dot files**？类似 Unix 的系统中，以点(.)开头的文件名默认是不显示的，如 **.bash_profile**，这些文件被称为 **dot files**。它们一般是各种工具的配置文件，如 **bash**、**vim** 等。比如我们可以在 **bash** 中将各种常用的长命令通过别名 (**alias**)设置为自己使用方便的短命令。

```
# git
alias g="git"
alias s="g status -s"

# applications
alias st='open -a "Sublime Text"'
alias atom="open -a '/Applications/Atom.app'"
alias preview="open -a '$PREVIEW'"
alias xcode="open -a '/Developer/Applications/'"
alias filemerge="open -a '/Developer/Applications/'"
alias safari="open -a safari"
alias firefox="open -a firefox"
```

Github 上有很多人共享了他们的 dot files ，这里收录了很多 <https://dotfiles.github.io/> 。在 git 上保存自己 dot files 有这些好处：方便备份、恢复和同步，学习别人的，共享你自己的。

注意，别人的东西不能随便用，适合自己的才是最好的。本来 dot files 就是用于个性化设置的。

5. say 命令

纯属娱乐，shell 中 say 命令用于朗读后面的字符串参数(只支持英文)，如 say 'hello world' 。

类似的快捷键和工具还很多，另外不同 shell 和 terminal 程序支持的功能也有所差异。

作者：韩国恺。本系列文章以 Addy & Matt 的系列视频 [Totally Tooling Tips](#) 的内容为基础重新整理而成，主要介绍一些(前端)开发者喜欢的工具和技巧。

📊 Post Views: 151

除非特别声明，此文章内容采用[知识共享署名 3.0](#)许可，代码示例采用[Apache 2.0](#)许可。更多细节请查看我们的[服务条款](#)。