

# Vim配置

---

## 一、光标移动

- 1 普通模式
- 2 插入模式
- 3 命令行模式

## 二、复制

- 1 普通模式
- 2 插入模式
- 3 visual 模式

## 三、选中

- 1 普通模式
- 2 插入模式

## 四、删除

- 1 插入模式
- 2 normal

## 五、撤销

- 1 插入模式
- 2 normal

## 六、保存

- 1 插入模式
- 2 normal

## 七、窗口管理-normal

- 1 分屏
- 2 切换不同窗口
- 3 调整窗口大小
- 4 切换窗口分割模式
- 5 移动窗口
- 6 关闭窗口-normal & visual

## 八、其他自定义

## 九、插件以及快捷键

### 1 补全

插件

快捷键

### 2 快速检索

插件

快捷键

### 3 文件管理

插件

快捷键

### 4 代码调试

插件

快捷键

### 5 注释

插件

快捷键

### 6 代码折叠

插件

快捷键

### 7 添加包围符号

插件

快捷键

### 8 历史记录

插件

快捷键

### 9 Git

插件

快捷键

### 10 多光标

插件

快捷键

### 11 笔记

插件

快捷键

markdown

插件

依赖

12 其他

终端

插件

快捷键

格式化

插件

快捷键

表格

插件

快捷键

自定义代码片段

Markdown

9 书签

插件

快捷键

注：部分按键在 vim 中的表示方法

键位	表示
ctrl	<C>
alt	<M>
shfit	<S>
enter	<CR>
空格	<space>
Esc	<esc>
方向键	<up>、<down>、<left>、<right>

删除	<BackSpace>
----	-------------

### 定义 leader

`let mapleader=" "` 定义 leader 键为 空格

红色加粗快捷键，模仿的 pycharm

## 一、光标移动

### 1 普通模式

键位	行为	等于
J	将光标向下移动 5 行	5j
K	将光标向上移动 5 行	5k
ctrl y	将视角向上移动 5 行，而不移动光标	5<ctrl-y>
ctrl e	将视角向上移动 5 行，而不移动光标	5<ctrl-e>

### 2 插入模式

键位	行为	等于
ctrl h	向左移动一个字符	left
ctrl j	向下移动一个字符	down
ctrl k	向上移动一个字符	up
ctrl l	向右移动一个字符	right
alt h	移到行首	HOME
alt l	移到行尾	END
ctrl o	下方，另起一行	<esc>o
alt o	上方，另起一行	<esc>O
<b>shift up</b>	<b>将当前行代码上移</b>	<b>&lt;esc&gt;ddkPA</b>
<b>shift down</b>	<b>将当前行代码下移</b>	<b>&lt;esc&gt;ddpA</b>
alt e	将光标置于下一个 word 的尾字符上	<esc>gea
alt b	将光标置于上一个 word 的尾字符上	<esc>ea

### 3 命令行模式

键位	行为	等于
ctrl h	向左移动一个字符	left
ctrl l	向右移动一个字符	right
ctrl j	向下查找历史命令	down
ctrl k	向上查找历史命令	up
alt h	移到命令行行首，即：后面	HOME
alt l	移到命令行行尾	END
alt b	向左一个单词	shift left
alt e	向右一个单词	shift right
ctrl u	删除光标前的内容	

## 二、复制

### 1 普通模式

键位	行为	等于
Y	复制至行末	y\$

### 2 插入模式

键位	行为	等于
ctrl v	粘贴复制的内容	<esc>pa
ctrl d	复制并粘贴一行	<esc>yypA

### 3 visual 模式

键位	行为	等于
Y	复制到剪贴板	" + y

## 三、选中

### 1 普通模式

键位	行为	等于
ctrl a	选中全文，此时在 visual 模式下	ggVG

2 插入模式

键位	行为	等于
ctrl a	选中全文，此时在 visual 模式下	<esc>ggVG

四、删除

1 插入模式

键位	行为	等于
ctrl y	删除一行	<esc>d di
ctrl w	删除光标至word 首字符	
ctrl u	删除光标至 word 尾字符	<esc>l de i
ctrl e	删除光标至行尾	<esc>l d \$ a
ctrl b	删除光标至行首	<esc>l d 0 i

2 normal

键位	行为	等于
S	删除该行，并进入插入模式	
D	删除光标后的内容	

五、撤销

1 插入模式

键位	行为	等于
ctrl z	撤销上一步操作，不能反撤销	<esc>u i

2 normal

键位	行为	等于
u	撤销上一步操作	
U	反撤销	

六、保存

1 插入模式

--	--	--

键位	行为	等于
ctrl s	保存	<esc>:w<cr>

## 2 normal

键位	行为	等于
Q	保存并退出	:wq
<space><sapce>	保存	<esc>:silent! write<cr>

## 七、窗口管理-normal

### 1 分屏

快捷键	行为	等于
sk	在当前窗口上方新建一个窗口	:abo split<CR>
sj	在当前窗口下方新建一个窗口	:rightbelow split<CR>
sh	在当前窗口左方新建一个窗口	:abo split<CR>
sl	在当前窗口右方新建一个窗口	:rightbelow split<CR>

### 2 切换不同窗口

快捷键	行为	等于
ctrl h	切换到左边的窗口	<C-w>h
ctrl l	切换到右边的窗口	<C-w>l
ctrl k	切换到上边的窗口	<C-w>k
ctrl j	切换到下边的窗口	<C-w>j

### 3 调整窗口大小

快捷键	行为	等于
alt right	增大	:vertical resize +5<CR>
alt left	减小	:vertical resize -5<CR>

alt up	增高	:resize +5<CR>
alt down	降低	:resize -5<CR>
ctrl-w =	等大	

## 4 切换窗口分割模式

快捷键	行为	等于
<leader>v	水平分割切换为垂直分割	<C-W>t<C-w>H
<leader>h	垂直分割切换为水平分割	<C-W>t<C-w>K

## 5 移动窗口

快捷键	行为	等于
;h	将窗口左移	<c-w>H
;j	窗口下移	<c-w>j
;k	窗口上移	<c-w>k
;l	窗口右移	<c-w>l

## 6 关闭窗口-normal & visual

快捷键	行为	等于
q	关闭当前窗口	<esc>:close<cr>

## 八、其他自定义

快捷键	行为	等于
<BackSpace>	normal模式，关闭高亮搜索	:nohl<cr>
R	normal模式，重新加载 vimrc 配置	:source \$MYVIMRC
jk	进入 normal 模式	<esc>
<<、>>	normal模式缩进，添加_，速度更快	<<_、>>_
ctrl r	运行程序	
9	跳转到末尾	g_
<leader>tt	在当前窗口下方打开一个终端	:set split<cr>:split<cr>:term<c r>



<leader>tn	新建一个 tab	
<leader>tc	关闭一个tab	
alt-H/L	将当前的 tab 移动到左侧/右侧	
alt h/l	切换到上一个/下一个tab 如果存在浮动终端，则切换为上一个或一个浮动终端 如果不存在多个tab，那么切换上一个或下一个buffer	

## 九、插件以及快捷键

### 1 补全

#### 插件

```
Plug 'neoclide/coc.nvim', { 'branch': 'release' }
```

#### 快捷键

快捷键	描述
tab	1. 如果存在下拉框，那么选择下一个 2. 如果不存在下拉框，那么强制显示下拉框，前是有内容显示，否则输入tab
shfit-tab	如果存在下拉框的话，选择下拉框的上一个
enter	选中下拉框中的该条目，如果条目可以扩展，那么将会扩展
gd	跳转到定义
gr	跳转到引用
gi	跳转到实现
gy	跳转到类型定义
alt j	1. normal 模式下，跳转到下一处错误 2. insert 模式下，如果存在 snippet 跳转块，跳转到下一处补全处
alt k	同 alt j 行为相反
<space>k	查看当前光标下函数的帮助
<space>rn	重命名变量、函数名称等 (python 使用 coc-jedi)
<space>rf	重构

<space>f	按照 lsp 的建议对代码错误进行自动修复
----------	-----------------------

coc 插件的额外快捷键

快捷键	描述
<space> l	打开 coclist
<space> a	显示所有错误 coclist dialogistic
<space> c	打开 coclist command
<space> o	打开 coclis outline
<space> O	打开 coclist symbols
<space> e	打开 coclist extensions
<space> p	恢复最近的 coclist 列表

## 2 快速检索

### 插件

#### 1. 使用 fzf

```
Plug 'junegunn/fzf.vim'
```

```
Plug 'junegunn/fzf', { 'do': {-> fzf#install()} }
```

#### 2. 使用 LeaderF

```
Plug 'Yggdrroot/LeaderF'
```

#### 3. coc-lists

```
let g:coc_global_extensions = ['coc-lists']
```

### 快捷键

快捷键	描述	插件
alt r	最近打开的文件	fzf/leaderf/coc
alt b	显示当前打开的所有 buffer	fzf/leaderf/coc
alt f	1. 搜索工程目录下得文件 2. 搜索 wiki 笔记目录下的文件	fzf/leaderf/coc
alt F	搜索 HOME 下的所有文件	fzf/leaderf/coc
alt w	显示打开的窗口	fzf/leaderf/coc
alt m	显示所有的标记, 即 marks	fzf/leaderf/coc
alt M	显示所有映射	fzf/leaderf/coc
alt s	1. 使用 rg 搜索当前工程下的文件内容	fzf/leaderf/coc

	2. 如果正在编辑 wiki 目录下的笔记，那么使用 rg 会搜索 wiki 笔记目录下的内容 3. visual 模式下选中内容之后会自动搜索选中的内容	
alt c	显示所有命令	fzf/leaderf/coc
?	模糊搜索所有打开的 buffer 的内容	fzf/leaderf/coc
alt y	显示复制内容	fzf/leaderf/coc
alt J	显示跳转	fzf
alt x	使用系统默认应用打开检索到的文件	fzf
<F8>	quickfix	fzf
<F9>	locationlist	fzf

### 3 文件管理

#### 插件

```
let g:coc_global_extensions=['coc-explorer']
```

#### 快捷键

快捷键	描述
F2	打开 coc-explorer
j	向下移动
k	向上移动
h	收起目录或跳到上级目录
l	展开目录/打开文件
H	递归收起目录
L	递归打开目录
J	跳到下一个可以展开的地方
K	跳到上一个可以展开的地方
enter	进入目录并切换工作目录为进入的目录
backspace	跳到上一级目录并切换工作目录为切换的目录
r	刷新目录
v	取消/选中，并向下移动
V	选中/取消，并向上移动

*	选中/取消选中
ctrl s	水平打开
ctrl v	垂直打开
t/ctrl t	新 tab 中打开
dd	剪切文件
Y	复制文件
D	删除文件
P	粘贴文件
R	重命名文件
N	添加文件或者目录，如果最后有 / 则表示添加目录
yp	复制文件路径
yn	复制文件名称
.	显示/关闭隐藏文件
alt x	使用系统默认应用打开文件
f	搜索文件
F	递归搜索文件

## 4 代码调试

### 插件

```
Plug 'puremourning/vimspector', {'do': '.install_gadget.py --enable-c --enable-python'}
```

具体语言，需要具体设置

### 快捷键

快捷键	描述
F5	继续调试，如果不在调试中则开始进行调试
F3	中止调试
F4	用相同的配置重新开始调试
F6	暂停调试
F9	在当前行上切换行的断点
F8	在光标下为表达式添加函数断点

F10	完成步骤
F11	下一步骤
F12	退出当前函数范围

## 5 注释

### 插件

```
Plug 'preservim/nerdcommenter'
```

### 快捷键

快捷键	描述
<leader>cc	注释
<leader>cu	取消注释
<leader>cs	用一种比较好看的方式注释当前行或选中的内容
<leader>ci	注释/取消注释
<leader>cm	用一种最简单的方式注释当前行或选中的内容
<leader>cl	按行的方式注释选中的内容
<leader>cb	注释选中的内容
<leader>ca	切换到另一种注释符号
<leader>cA	在当前行最后增加注释符号并进入插入模式
<leader>c\$	从当前光标处注释到当前行的最后

## 6 代码折叠

### 插件

```
Plug 'pseewald/vim-anyfold'
```

### 快捷键

快捷键	描述
zc	关闭光标下的折叠
zC	关闭光标下的折叠以及嵌套的折叠
zo	打开光标下的折叠
zO	打开光标下的折叠以及嵌套的折叠
za	光标下折叠若是关闭则打开若是打开则关闭

zA	光标下折叠若是关闭则打开包括嵌套若是打开则关闭
zm	一层一层的关闭折叠
zM	关闭所有折叠
zr	一层一层的打开折叠
zR	打开所有折叠
zn	禁用折叠
zN	启用折叠
zj	移动到下一个折叠
zk	移动到上一个折叠
<enter>	打开或关闭折叠

## 7 添加包围符号

### 插件

```
Plug 'tpope/vim-surround'
```

### 快捷键

快捷键	描述
ds"	删除双引号
ds(	删除括号
dst	删除 <> </>
cs"	双引号改为单引号
cs"<q>	双引号改为<q></q>
cs)]	圆括号改为中括号
cst"	将tag 改为双引号
ysiw]	iw是选择一个单词，在选中的单词处增加中括号
ysiw[	同上，但两侧增加空格
yss)	整行增加括号
ysiw<em>	在一个单词处增加<em></em>
vS"	在选中内容两边添加双引号
ysiwfprint<cr>	在当前单词前添加 print 函数
ysiWfprint<cr>	同上，这里的单词指 WORD，以空格为分界

yswFprint<cr>	同上，但会在括号两边添加空格
---------------	----------------

Old text	Command	New text ~
"Hello *world!"	ds"	Hello world!
(123+4*56)/2	ds)	123+456/2
<div>Yo!*</div>	dst	Yo!

Old text	Command	New text ~
"Hello *world!"	cs""	'Hello world!'
"Hello *world!"	cs"<q>	<q>Hello world!</q>
(123+4*56)/2	cs)]	[123+456]/2
(123+4*56)/2	cs)[	[ 123+456 ]/2
<div>Yo!*</div>	cst<p>	<p>Yo!</p>

Old text	Command	New text ~
Hello w*orld!	ysiw)	Hello (world)!

Old text	Command	New text ~
Hello w*orld!	yssB	{Hello world!}

Old text	Command	New text ~
"hello"	yswFprint<cr>	print("hello")
"hello"	yswFprint<cr>	print( "hello" )
"hello"	ysw<C-f>print<cr>	(print "hello")

## 8 历史记录

### 插件

Plug 'mbbill/undotree'

### 快捷键

快捷键	描述
L	打开撤回历史
u	更新的记录
e	更老的记录

## 9 Git

### 插件

```
Plug 'airblade/vim-gitgutter'
```

```
Plug 'vim-fugitive', {'on': ['Gwrite', 'Gcommit', 'Gread', 'Gdiff',  
'Gblame']}
```

## 快捷键

快捷键	描述
<leader>gk	跳转到上一个修改地方
<leader>gj	跳转到下一个修改地方
<leader>gp	浮动预览光标处的修改
<leader>gu	撤销光标处相对于上次提交的修改
<leader>gw	添加当前文件到 index
<leader>gc	提交
<leader>gf	查看差异
<leader>gr	恢复当前文件到上个版本
<leader>gb	blame

## 10 多光标

### 插件

```
Plug 'mg979/vim-visual-multi'
```

### 快捷键

快捷键	描述
tab	光标和扩展模式直接切换
shift-左右方向键	选中区域并进入 visual-multi-mode
ctrl c	在当前位置插入一个光标（然后使用上下左右键移动到指定位置，添加下一个光标，hjkl键会移动添加的光标）
ctrl up/down	垂直方向插入多光标
ctrl s	1. 选中当前光标下的所有单词 2. visual 模式下选中所有的当前选中的内容
ctrl n/N	在visual模式下选中内容之后，按下ctrl-n会在下一处与当前选中内一样的地方插入光标。非visual模式下，与ctrl-s表现差不多，不过是依次选择当前光标下的单词。
q	取消当前光标或者选中的区域



]/[	在多个选中的光标之间进行跳转
n/N	跳到下一个/上一个匹配的模式，可以代替 ]/[ 进行使用
S	多光标模式下，将多个选中的内容以某种符号进行包围
o	交换光标位置
V	以正则匹配的方式设置光标

## 11 笔记

### 插件

```
Plug 'vimwiki/vimwiki'
```

### 快捷键

快捷键	描述
alt s	搜索 vimwiki 中的内容
<leader>ww	打开 vimwiki
<leader>wt	在新的tab打开vimwiki
<leader>wd	删除当前page
<leader>wr	重命名当前的page
<leader>wi	打开日记列表
<leader>w<leader>i	更新当前的日记列表
<leader>w<leader>w	打开今天的日记
<leader>w<leader>t	在新的tab中打开今天的日记
ctrl up/down	打开昨天/明天的日记
enter	创建链接
back	回到上一级目录
tab	跳转到下一处链接
shift tab	跳转到上一处链接
=	增加标题
-	减少标题
+	创建链接
glm	增加list item的缩进

gll	减少list item的缩进
gl* 或 gl8	将该行变为*条目
gl# 或 gl3	将该行变为#
gl-	将该行变为-
gl1	将该行变为1.
cr	插入模式
tab	插入模式，跳到下一个输入框
gqq 或 gww	格式化表格

## markdown

### 插件

```
Plug 'instant-markdown/vim-instant-markdown', { 'for' : 'markdown' }
```

### 依赖

- pip install --user smdv 或 [sudo] npm -g install instant-markdown-d
- xdg-utils
- curl
- nodejs
- 指定的浏览器

## 12 其他

### 终端

#### 插件

```
Plug 'voldikss/vim-floaterm'
```

#### 快捷键

快捷键	描述
alt =	打开/关闭当前终端
alt +	新建一个终端
alt h	打开终端后，切换到上一个终端
alt l	打开终端后，切换到下一个终端

## 格式化

### 插件

```
Plug 'junegunn/vim-easy-align'
```

## 快捷键

快捷键	描述
ga+符号	根据符号对齐文本

## 表格

### 插件

```
Plug 'dhruvasagar/vim-table-mode'
```

## 快捷键

快捷键	描述
SPACE t m	开关表格模式
SPACE t r	重组表格

## 自定义代码片段

## Markdown

快捷键	描述
,n	---
,b	<b>粗体文字</b>
,s	<del>被划去的文字</del>
,i	<i>斜体文字</i>
,d	<code>代码块</code>
,c	大的 <code>代码块</code>
,m	- [ ] 清单
,p	图片
,a	<a href="#">链接</a>
,1	# H1
,2	## H2
,3	### H3
,4	#### H4

## 9 书签

### 插件

```
Plug 'MattesGroeger/vim-bookmarks'
```

### 快捷键

快捷键	描述
ma	增加一个带有备注的书签
ml	显示所有书签
mj	下一个书签
mk	上一个书签
mc	清除一个书签