

强大的 vim 配置

作者: [zgzhang](#)

原文链接: <https://ld246.com/article/1495447437683>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)


```

call append(line("."), "# File Name: ".expand("%"))<br>
call append(line(".")+1, "# Author: ma6174")<br>
call append(line(".")+2, "# mail: ma6174@163.com")<br>
call append(line(".")+3, "# Created Time: ".strftime("%c"))<br>
call append(line(".")+4, "#####")<br>
call append(line(".")+5, "#!/bin/bash")<br>
call append(line(".")+6, "")<br>
else<br>
call setline(1, "/******")<br>
call append(line("."), " &gt; File Name: ".expand("%"))<br>
call append(line(".")+1, " &gt; Author: ma6174")<br>
call append(line(".")+2, " &gt; Mail: <a href="https://ld246.com/forward?goto=mailto%3Aa6174%40163.com" target="_blank" rel="nofollow ugc">ma6174@163.com</a> ")<br>
call append(line(".")+3, " &gt; Created Time: ".strftime("%c"))<br>
call append(line(".")+4, " *****")<br>
call append(line(".")+5, "")<br>
endif<br>
if &amp;filetype == 'cpp'<br>
call append(line(".")+6, "#include")<br>
call append(line(".")+7, "using namespace std;")<br>
call append(line(".")+8, "")<br>
endif<br>
if &amp;filetype == 'c'<br>
call append(line(".")+6, "#include")<br>
call append(line(".")+7, "")<br>
endif<br>
"新建文件后，自动定位到文件末尾<br>
autocmd BufNewFile * normal G<br>
endfunc<br>
.....

br>
"键盘命令<br>
.....

/p>
<p>nmap w :w!<br>
nmap f :find</p>
<p>" 映射全选 + 复制 ctrl+a<br>
map ggVGY<br>
map! ggVGY<br>
map gg=G<br>
" 选中状态下 Ctrl+c 复制<br>
vmap "+y<br>
"去空行<br>
nnoremap :g/^\s*$/d<br>
"比较文件<br>
nnoremap :vert difffsplit<br>
"新建标签<br>
map :tabnew<br>
"列出当前目录文件<br>
map :tabnew .<br>
"打开树状文件目录<br>
map \be<br>

```

```

"C, C++ 按 F5 编译运行<br>
map :call CompileRunGcc()<br>
func! CompileRunGcc()<br>
exec "w"<br>
if &filetype == 'c'<br>
exec "!g++ % -o %&lt;"<br>
exec "!./%&lt;"<br>
elseif &filetype == 'cpp'<br>
exec "!g++ % -o %&lt;"<br>
exec "!./%&lt;"<br>
elseif &filetype == 'java'<br>
exec "!javac %"<br>
exec "!java %&lt;"<br>
elseif &filetype == 'sh'<br>
:./%<br>
endif<br>
endfunc<br>
"C,C++ 的调试<br>
map :call Rungdb()<br>
func! Rungdb()<br>
exec "w"<br>
exec "!g++ % -g -o %&lt;"<br>
exec "!gdb ./%&lt;"<br>
endfunc<br>
.....

br>
""实用设置<br>
..... <

r>
" 设置当文件被改动时自动载入<br>
set autoread<br>
" quickfix 模式<br>
autocmd FileType c,cpp map :w:make<br>
"代码补全<br>
set completeopt=preview,menu<br>
"允许插件<br>
filetype plugin on<br>
"共享剪贴板<br>
set clipboard+=unnamed<br>
"从不备份<br>
set nobackup<br>
"make 运行<br>
:set makeprg=g++ \ -Wall \ %<br>
"自动保存<br>
set autowrite<br>
set ruler " 打开状态栏标尺<br>
set cursorline " 突出显示当前行<br>
set magic " 设置魔术<br>
set guioptions-=T " 隐藏工具栏<br>
set guioptions-=m " 隐藏菜单栏<br>
"set statusline=\ %&lt;%F[%1*M%*%nR%H]%=\ %y\ %0(%&fileformat)\ %&en
oding)\ %c:%l/%L%<br>
" 设置在状态行显示的信息<br>
set foldcolumn=0<br>

```

```
set foldmethod=indent<br>
set foldlevel=3<br>
set foldenable          " 开始折叠<br>
" 不要使用 vi 的键盘模式，而是 vim 自己的<br>
set nocompatible<br>
" 语法高亮<br>
set syntax=on<br>
" 去掉输入错误的提示声音<br>
set noeb<br>
" 在处理未保存或只读文件的时候，弹出确认<br>
set confirm<br>
" 自动缩进<br>
set autoindent<br>
set cindent<br>
" Tab 键的宽度<br>
set tabstop=4<br>
" 统一缩进为 4<br>
set softtabstop=4<br>
set shiftwidth=4<br>
" 不要用空格代替制表符<br>
set noexpandtab<br>
" 在行和段开始处使用制表符<br>
set smarttab<br>
" 显示行号<br>
set number<br>
" 历史记录数<br>
set history=1000<br>
"禁止生成临时文件<br>
set nobackup<br>
set noswapfile<br>
"搜索忽略大小写<br>
set ignorecase<br>
"搜索逐字符高亮<br>
set hlsearch<br>
set incsearch<br>
"行内替换<br>
set gdefault<br>
"编码设置<br>
set enc=utf-8<br>
set fencs=utf-8,ucs-bom,shift-jis,gb18030,gbk,gb2312,cp936<br>
"语言设置<br>
set langmenu=zh_CN.UTF-8<br>
set helplang=cn<br>
" 我的状态行显示的内容（包括文件类型和解码）<br>
"set statusline=%F%m%r%h%w\ [FORMAT=%{&#amp;ff}]\ [TYPE=%Y]\ [POS=%l,%v][%p%%]\
%{strftime("%d/%m/%y\ -\ %H:%M")}<br>
"set statusline=[%F]%y%r%m%*%=[Line:%l/%L,Column:%c][%p%%]<br>
" 总是显示状态行<br>
set laststatus=2<br>
" 命令行（在状态行下）的高度，默认为 1，这里是 2<br>
set cmdheight=2<br>
" 侦测文件类型<br>
filetype on<br>
" 载入文件类型插件<br>
```

```

filetype plugin on<br>
" 为特定文件类型载入相关缩进文件<br>
filetype indent on<br>
" 保存全局变量<br>
set viminfo+=!<br>
" 带有如下符号的单词不要被换行分割<br>
set iskeyword+=_,$,@,%,#,-<br>
" 字符间插入的像素行数目<br>
set linespace=0<br>
" 增强模式中的命令行自动完成操作<br>
set wildmenu<br>
" 使回格键 (backspace) 正常处理 indent, eol, start 等<br>
set backspace=2<br>
" 允许 backspace 和光标键跨越行边界<br>
set whichwrap+=&lt;,&gt;,h,l<br>
" 可以在 buffer 的任何地方使用鼠标 (类似 office 中在工作区双击鼠标定位) <br>
set mouse=a<br>
set selection=exclusive<br>
set selectmode=mouse,key<br>
" 通过使用: commands 命令, 告诉我们文件的哪一行被改变过<br>
set report=0<br>
" 在被分割的窗口间显示空白, 便于阅读<br>
set fillchars=vert:\ ,stl:\ ,stlnc:<br>
" 高亮显示匹配的括号<br>
set showmatch<br>
" 匹配括号高亮的时间 (单位是十分之一秒) <br>
set matchtime=1<br>
" 光标移动到 buffer 的顶部和底部时保持 3 行距离<br>
set scrolloff=3<br>
" 为 C 程序提供自动缩进<br>
set smartindent<br>
" 高亮显示普通 txt 文件 (需要 txt.vim 脚本) <br>
au BufRead,BufNewFile * setfiletype txt<br>
"自动补全<br>
:inoremap ( ()i<br>
:inoremap ) =ClosePair('')<br>
:inoremap { {}O<br>
:inoremap } =ClosePair('{}')<br>
:inoremap [ []i<br>
:inoremap ] =ClosePair(']')<br>
:inoremap " ""i<br>
:inoremap ' ''i<br>
function! ClosePair(char)<br>
if getline('.')[col('.') - 1] == a:char<br>
return ""<br>
else<br>
return a:char<br>
endif<br>
endfunction<br>
filetype plugin indent on<br>
"打开文件类型检测, 加了这句才可以用智能补全<br>
set completeopt=longest,menu<br>
" CTags 的设定<br>

```

```

....." <br>
let Tlist_Sort_Type = "name" " 按照名称排序<br>
let Tlist_Use_Right_Window = 1 " 在右侧显示窗口<br>
let Tlist_Compact_Format = 1 " 压缩方式<br>
let Tlist_Exist_OnlyWindow = 1 " 如果只有一个 buffer, kill 窗口也 kill 掉 buffer<br>
let Tlist_File_Fold_Auto_Close = 0 " 不要关闭其他文件的 tags<br>
let Tlist_Enable_Fold_Column = 0 " 不要显示折叠树<br>
autocmd FileType java set tags+=D:\tools\java\tags<br>
"autocmd FileType h,cpp,cc,c set tags+=D:\tools\cpp\tags<br>
"let Tlist_Show_One_File=1 "不同时显示多个文件的 tag, 只显示当前文件的<br>
"设置 tags<br>
set tags=tags<br>
"set autochdir</p>
<p>.....
"....." <br>
"其他东东<br>
.....
"....." <br>
"默认打开 Taglist<br>
let Tlist_Auto_Open=1<br>
"....." <br>
" Tag list (ctags)<br>
"....." <br>
let Tlist_Ctags_Cmd = '/usr/bin/ctags'<br>
let Tlist_Show_One_File = 1 "不同时显示多个文件的 tag, 只显示当前文件的<br>
let Tlist_Exit_OnlyWindow = 1 "如果 taglist 窗口是最后一个窗口, 则退出 vim<br>
let Tlist_Use_Right_Window = 1 "在右侧窗口中显示 taglist 窗口<br>
" minibufexpl 插件的一般设置<br>
let g:miniBufExplMapWindowNavVim = 1<br>
let g:miniBufExplMapWindowNavArrows = 1<br>
let g:miniBufExplMapCTabSwitchBufs = 1</p>
<p>let g:miniBufExplModSelTarget = 1</p>
<p>nnd, 忘记原作地址了, 如有问题请找我删除。 </p>

```