VIM的使用

2017年12月21日

18:48

Vim 是一个上古神器，本篇文章主要持续总结使用 Vim 的过程中不得不了解的一些指令和注意事项，以及持续分享一个前端工作者不得不安装的一些插件，而关于 Vim 的简介，主题的选择，以及为何使用 vim-plug 来管理插件等内容，有兴趣的同学下来可以自己了解和对比下

安装

sudo apt-get install vim  // Ubuntu

其他平台，可以自行谷歌

新手指南

vimtutor  // vim 教程

上面是史上最简单，最全面的 Vim 基础教程，至今无人超越

下面是作者基于上面的归纳

移动光标

# hjkl

# 2w 向前移动两个单词

# 3e 向前移动到第 3 个单词的末尾

# 0 移动到行首

# $ 当前行的末尾

# gg 文件第一行

# G 文件最后一行

# 行号+G 指定行

# <ctrl>+o 跳转回之前的位置

# <ctrl>+i 返回跳转之前的位置

退出

# <esc> 进入正常模式

# :q! 不保存退出

# :wq 保存后退出

删除

# x 删除当前字符

# dw 删除至当前单词末尾

# de 删除至当前单词末尾，包括当前字符

# d$ 删除至当前行尾

# dd 删除整行

# 2dd 删除两行

修改

# i 插入文本

# A 当前行末尾添加

# r 替换当前字符

# o 打开新的一行并进入插入模式

撤销

# u 撤销

# <ctrl>+r 取消撤销

复制粘贴剪切

# v 进入可视模式

# y 复制

# p 粘贴

# yy 复制当前行

# dd 剪切当前行

状态

# <ctrl>+g 显示当前行以及文件信息

查找

# / 正向查找（n：继续查找，N：相反方向继续查找）

# ？ 逆向查找

# % 查找配对的 {，[，(

# :set ic 忽略大小写

# :set noic 取消忽略大小写

# :set hls 匹配项高亮显示

# :set is 显示部分匹配

替换

# :s/old/new 替换该行第一个匹配串

# :s/old/new/g 替换全行的匹配串

# :%s/old/new/g 替换整个文件的匹配串

执行外部命令

# :!shell 执行外部命令

.vimrc

.vimrc 是 Vim 的配置文件，需要我们自己创建

cd Home               // 进入 Home 目录

touch .vimrc          // 配置文件

# Unix

# vim-plug

# Vim

curl -fLo ~/.vim/autoload/plug.vim --create-dirs

<https://raw.githubusercontent.com/junegunn/vim-plug/master/plug.vim>

# Neovim

curl -fLo ~/.local/share/nvim/site/autoload/plug.vim --create-dirs

<https://raw.githubusercontent.com/junegunn/vim-plug/master/plug.vim>

其他平台，可以查看 vim-plug

基本配置

取消备份

set nobackup

set noswapfile

文件编码

set encoding=utf-8

显示行号

set number

取消换行

set nowrap

显示光标当前位置

set ruler

设置缩进

set cindent

set tabstop=2

set shiftwidth=2

突出显示当前行

set cursorline

查找

set ic

set hls

set is

左下角显示当前 vim 模式

set showmode

代码折叠

# 启动 vim 时关闭折叠代码

set nofoldenable

主题

syntax enable

set background=dark

colorscheme solarized

插件配置

树形目录

Plug 'scrooloose/nerdtree'

Plug 'jistr/vim-nerdtree-tabs'

Plug 'Xuyuanp/nerdtree-git-plugin'

autocmd vimenter \* NERDTree

map <C-n> :NERDTreeToggle <CR>

let NERDTreeShowHidden=1

let g:NERDTreeShowIgnoredStatus = 1

let g:nerdtree\_tabs\_open\_on\_console\_startup=1

let g:NERDTreeIndicatorMapCustom = {

     "Modified"  : "✹",

     "Staged"    : "✚",

     "Untracked" : "✭",

     "Renamed"   : "➜",

     "Unmerged"  : "═",

     "Deleted"   : "✖",

     "Dirty"     : "✗",

     "Clean"     : "✔︎",

     'Ignored'   : '☒',

     "Unknown"   : "?"

     }

# o 打开关闭文件或目录

# e 以文件管理的方式打开选中的目录

# t 在标签页中打开

# T 在标签页中打开，但光标仍然留在 NERDTree

# r 刷新光标所在的目录

# R 刷新当前根路径

# X 收起所有目录

# p 小写，跳转到光标所在的上一级路径

# P 大写，跳转到当前根路径

# J 到第一个节点

# K 到最后一个节点

# I 显示隐藏文件