

Shell 和 CLI 工具美化

Ivan Chien

2023.7.18

目录

I. 前置知识	1
II. 我使用到的和你可能同样需要的东西	1
III. 使用到的软件	1
IV. 使用到的初始配置	1
IV.1. Vim	2
IV.2. Tmux	2
IV.3. Bash	2
V. 流程大纲	2
V.1. Bash/Zsh	3
V.1.a. Zsh 插件	3
V.2. Vim	3
V.3. Tmux	3
VI. 链接	3
VI.1. Shell	3
VI.2. Vim	3
VI.3. Tmux	4
VI.4. 我的相关配置	4
VII. 附录	4
VII.1. 本篇 PDF 的源码	4
VII.2. Tmux 成品配置	4

I. 前置知识

1. 你需要刚好够用的 Shell 知识
2. 你需要知道 git 和 GitHub 是什么、是干嘛用的
3. 最好完整学过 [MIT 6.NULL](#) 公开课，对 Linux 有一定了解

II. 我使用到的和你可能同样需要的东西

1. 一台电脑
2. Linux 系统。可以装在主机上，也可以使用虚拟机。如果你是 Windows 平台亦可以使用 [WSL 2](#)。我这边使用的是 SSH 连接真实 Linux 机器的方式。
3. Linux 最好是 Ubuntu。我演示使用的可能是 Fedora。

III. 使用到的软件

1. apt
2. bash / zsh
3. vim
4. tmux

IV. 使用到的初始配置

IV.1. Vim

```
~/.vimrc

"" Basic
set noswf
set nobk
set title
set vb
set noeb
set t_vb=

"" Editing
syntax on
set nu
set rnu
set et
set ts=4
set sw=4
set sts=4
set mouse=a

"" Keymaps
let mapleader=' '
imap kj <Esc>
nnoremap <leader>fs :w<CR>
nnoremap <leader>fr :source %<CR>
nnoremap <leader>q :q<CR>
nnoremap < <<
nnoremap > >>
vnoremap < <gv
vnoremap > >gv
nnoremap H ^
nnoremap L $
nnoremap U <C-u>
nnoremap D <C-d>
nmap J <nop>
nmap s <nop>

""
```

IV.2. Tmux

```
~/.tmux.conf

unbind C-b
set -g prefix C-x
bind C-x send-prefix

#

set-option -sa terminal-overrides ",xterm*:Tc"
set -g default-terminal "screen-256color"
```

IV.3. Bash

如果你用的是 Ubuntu 那就不用 Ubuntu 默认的 Bash 配置。

V. 流程大纲

V.1. Bash/Zsh

1. Bash、Zsh、Shell 分别都是什么? <https://www.zhihu.com/question/21418449/answer/2292448029>
2. Emacs
3. prompt 和 PS1
4. 安装和使用 zsh 和 oh-my-zsh
5. 如何选择?
6. fish

V.1.a. Zsh 插件

```
git clone https://github.com/zsh-users/zsh-syntax-highlighting.git $ZSH_CUSTOM/  
plugins/zsh-syntax-highlighting  
git clone https://github.com/zsh-users/zsh-autosuggestions.git $ZSH_CUSTOM/  
plugins/zsh-autosuggestions
```

V.2. Vim

1. 「无缝」:
 - 终端内
 - 全键盘
 - 主键区
2. 优势
 - 高度可定制
 - leader key
3. 插件管理
4. 主题和 status line
5. 想要深入了解? <https://www.bilibili.com/video/BV1ko4y1A7fQ/>

V.3. Tmux

1. 是什么
2. 有什么用/怎么用
3. 常用配置
4. 插件和 status line

VI. 链接

VI.1. Shell

- [Bash prompt generator](#)
- [Oh-my-zsh 官网](#)
- [spaceship prompt](#)
- [powerlevel10k](#)

VI.2. Vim

- [vim-plug](#)
- [Awesome Vim](#)
- [vim-airline](#)
- [Catppuccin for Vim](#)

VI.3. Tmux

- [Tmux Plugin Manager](#)
- <https://github.com/gpakosz/.tmux>

VI.4. 我的相关配置

- <https://github.com/Yestercafe/dotfiles>
- <https://github.com/Yestercafe/vim>
- <https://github.com/Yestercafe/nvim>

VII. 附录

VII.1. 本篇 PDF 的源码

<https://github.com/lab530/lab530.github.io/blob/main/storage/Shell和CLI工具美化.typ>

VII.2. Tmux 成品配置

~/tmux.conf

```
# 24-bit true color
set-option -sa terminal-overrides ",xterm*:Tc"
# set -g default-terminal "screen-256color"

# Operations
set -g mouse on
unbind C-b
set -g prefix C-x
bind C-x send-prefix

# Navigation
bind -n M-Left select-pane -L
bind -n M-Right select-pane -R
bind -n M-Up select-pane -U
bind -n M-Down select-pane -D
bind -n S-Left previous-window
bind -n S-Right next-window

# Index
set -g base-index 1
set -g pane-base-index 1
set-window-option -g pane-base-index 1
set -g renumber-windows on
setw -g allow-rename off
setw -g automatic-rename off

# TPM
set -g @plugin 'tmux-plugins/tpm'
set -g @plugin 'tmux-plugins/tmux-sensible'
set -g @plugin 'christoomey/vim-tmux-navigator'
set -g @catppuccin_flavour 'mocha'
set -g @plugin 'dreamsofcode-io/catppuccin-tmux'
set -g @plugin 'tmux-plugins/tmux-yank'
run '~/.tmux/plugins/tpm/tpm'

# Vi mode
## use C-[ to enter vi-mode
set-window-option -g mode-keys vi
bind-key -T copy-mode-vi v send-keys -X begin-selection
```

```
bind-key -T copy-mode-vi C-v send-keys -X rectangle-toggle
bind-key -T copy-mode-vi y send-keys -X copy-selection-and-cancel
```

```
# Rewrite split behavior
```

```
bind '"' split-window -v -c "#{pane_current_path}"
```

```
bind % split-window -h -c "#{pane_current_path}"
```