

命令行美化教程

更换镜像源

更换哈工大深圳校园镜像站，在校园网环境下链接，享受更快的下载速度。

```
nano /etc/apt/sources.list
```

将以下内容复制到 sources.list 文件中，替换原有内容：

```
# 默认注释了源码镜像以提高 apt update 速度，如有需要可自行取消注释
deb https://mirrors.osa.moe/ubuntu/ jammy main restricted universe multiverse
# deb-src https://mirrors.osa.moe/ubuntu/ jammy main restricted universe multiverse
deb https://mirrors.osa.moe/ubuntu/ jammy-updates main restricted universe multiverse
# deb-src https://mirrors.osa.moe/ubuntu/ jammy-updates main restricted universe multiv
deb https://mirrors.osa.moe/ubuntu/ jammy-backports main restricted universe multiverse
# deb-src https://mirrors.osa.moe/ubuntu/ jammy-backports main restricted universe mult

# deb https://mirrors.osa.moe/ubuntu/ jammy-security main restricted universe multivers
# # deb-src https://mirrors.osa.moe/ubuntu/ jammy-security main restricted universe mul

deb http://security.ubuntu.com/ubuntu/ jammy-security main restricted universe multiver
# deb-src http://security.ubuntu.com/ubuntu/ jammy-security main restricted universe mu

# 预发布软件源，不建议启用
# deb https://mirrors.osa.moe/ubuntu/ jammy-proposed main restricted universe multivers
# # deb-src https://mirrors.osa.moe/ubuntu/ jammy-proposed main restricted universe mul
```

terminal美化教程

无论你的桌面是kde还是gnome，终端都是你与系统交互的主要方式之一。美化终端不仅可以提升工作效率，还能让你的桌面环境更加个性化。下面介绍我常用的终端：wezterm，经过一定的配置后，它可以提供非常好的用户体验。

wezterm安装

- Ubuntu/Debian

```
sudo apt install wezterm
```

- Arch Linux

```
sudo pacman -S wezterm
```

美化

终端美化依赖NerdFont字体，shell 美化也会用到

```
sudo apt install font-jetbrains-mono-nerd-font
```

直接克隆配置文件

```
git clone https://github.com/KevinSilvester/wezterm-config.git ~/.config/wezterm
```

调整配置文件:

```
cd ~/.config/wezterm
```

- 你可以在backdrops目录下放置你喜欢的壁纸，或者直接使用默认的壁纸。
- 在bindings.lua中，你可以自定义快捷键。
- 在colors.lua中，你可以选择不同的配色方案。
- 在font.lua中，你可以将我们刚刚安装的字体放上去，同时调节字的大小(font_size)。

shell美化

终端美化的同时，shell 美化也很重要。对于新手，我推荐 fish shell，它比 bash 更加现代化，支持更好的自动补全和语法高亮。

安装 fish shell

```
sudo apt install fish
```

设置 fish 为默认 shell

```
chsh -s $(which fish)
```

fish shell 的配置文件位于 `~/.config/fish/config.fish`。你可以在这里添加自定义的环境变量、别名等。

配置环境变量

例如你想要将路径添加到 PATH 环境变量中：

```
# 方法1：使用 fish_add_path (推荐)
fish_add_path ~/bin
fish_add_path ~/.local/bin
```

```
# 方法2：使用 set -Ux
set -Ux PATH ~/bin $PATH
```

```
# 或者设置其他环境变量
set -Ux EDITOR vim
set -Ux GOPATH ~/go
```

安装插件

我们使用 Fisher 作为插件管理器：

```
curl -sL https://raw.githubusercontent.com/jorgebucaran/fisher/main/functions/fisher.fish
```

autopair.fish：自动补全括号和引号

fish-colored-man：为 man 命令添加颜色

```
fisher install jorgebucaran/autopair.fish
fisher install decorators/fish-colored-man
```

自定义技巧

- 添加别名：
例如将 ls-la 命令简化为 ll 和 clear 命令简化为 c:

```
alias ll='ls -la'
alias c='clear'
```

主题

我们使用 starship 主题，它提供了非常美观的提示符和颜色方案。

```
sudo apt install starship
nano ~/.config/fish/config.fish
```

然后在配置文件中添加以下内容：

```
if status is-interactive
    # Commands to run in interactive sessions can go here
    starship init fish | source &
    thefuck --alias | source &
    ~/.config/fish/tty.sh &
end
```

配置 starship 主题：

```
touch ~/.config/starship.toml
nano ~/.config/starship.toml
```

然后在配置文件中添加以下内容：

```

format = ""
[ ](#a3aed2)\
[ ](bg:#a3aed2 fg:#1793D1)\
[ ](bg:#769ff0 fg:#a3aed2)\
$directory\
[ ](fg:#769ff0 bg:#394260)\
$git_branch\
$git_status\
[ ](fg:#394260 bg:#212736)\
$nodejs\
$rust\
$golang\
$php\
$python\
$java\
$ruby\
$c\
$c++\
$haskell\
$ktlin\
[ ](fg:#212736 bg:#1d2230)\
$cmd_duration\
$time\
[ ](fg:#1d2230)\
\n$character""

```

```

[cmd_duration]
format = '[[ 🕒 $duration ](fg:#a0a9cb bg:#1d2230)]($style)'
min_time = 60_000 # Minimum time in milliseconds to display duration
style = "bg:#1d2230"

```

```

[directory]
format = "[ $path ]($style)"
style = "fg:#e3e5e5 bg:#769ff0"
truncation_length = 3
truncation_symbol = ".../"

```

```

[directory.substitutions]
"Documents" = "[ ] "
"Downloads" = "[ ] "
"Music" = "[ ] "
"Pictures" = "[ ] "

```

```
[git_branch]
format = '[[ $symbol $branch ](fg:#769ff0 bg:#394260)]($style)'
style = "bg:#394260"
symbol = "□"
```

```
[git_status]
format = '[[($all_status$ahead_behind $count )](fg:#769ff0 bg:#394260)]($style)'
style = "bg:#394260"
modified = "~${count}"
staged = "+${count}"
untracked = "?${count}"
deleted = "-${count}"
conflicted = "!${count}"
```

```
[nodejs]
format = '[[ $symbol ($version) ](fg:#769ff0 bg:#212736)]($style)'
style = "bg:#212736"
symbol = "□"
```

```
[rust]
format = '[[ $symbol ($version) ](fg:#769ff0 bg:#212736)]($style)'
style = "bg:#212736"
symbol = "□"
```

```
[golang]
format = '[[ $symbol ($version) ](fg:#769ff0 bg:#212736)]($style)'
style = "bg:#212736"
symbol = "□"
```

```
[php]
format = '[[ $symbol ($version) ](fg:#769ff0 bg:#212736)]($style)'
style = "bg:#212736"
symbol = "□"
```

```
[java]
format = '[[ $symbol( $version) ](fg:crust bg:green)]($style)'
style = "bg:green"
symbol = "□ "
```

```
[kotlin]
format = '[[ $symbol( $version) ](fg:crust bg:green)]($style)'
style = "bg:green"
```

```

symbol = ""

[haskell]
format = '[[ $symbol( $version) ](fg:crust bg:green)]($style)'
style = "bg:green"
symbol = ""

[python]
format = '[[ $symbol( $version)(\($virtualenv\)) ](fg:crust bg:green)]($style)'
style = "bg:green"
symbol = ""

[c]
format = '[[ $symbol( $version) ](fg:crust bg:green)]($style)'
style = "bg:green"
symbol = " "

[cpp]
format = '[[ $symbol( $version) ](fg:color_fg0 bg:color_blue)]($style)'
style = "bg:color_blue"
symbol = " "

[time]
disabled = false
format = '[[  $time ](fg:#a0a9cb bg:#1d2230)]($style)'
style = "bg:#1d2230"
time_format = "%R" # Hour:Minute Format

```

这样你就可以得到一个 tokyonight 风格的提示符了。

如果你还想更换其他风格的主题，可以在 starship 的配置文件中进行修改。例如，你可以将背景色和前景色更改为你喜欢的颜色，或者选择其他的主题风格。

主题网站：[Starship Themes](#)

其他命令行工具

bat

bat 是一个增强版的 cat 命令，支持语法高亮和分页显示。

```
sudo apt install bat
```

配置主题

```
mkdir -p ~/.config/bat
nano ~/.config/bat/config
```

然后添加以下内容：

```
--theme="Dracula"
```

zoxide

zoxide 是一个智能的目录跳转工具，可以根据你的使用习惯快速跳转

```
sudo apt install zoxide
```

例如你想要访问的文件路径为 `/home/user/projects/myproject`，你可以使用以下命令快速跳转：

你在第一次访问该目录时，zoxide 会自动记录该路径。下次你只需输入以下命令即可快速跳转：

```
z myproject
echo "zoxide init fish | source" >> ~/.config/fish/config.fish
```

btop

btop 是一个资源监视工具，类似于 htop，但界面更加美观和现代化。

```
sudo apt install btop
```

然后在界面中点击 ESC，选中 options 回车，在里面可以自由选择主题

yazi

yazi 是一个命令行下的图片查看器，支持多种图片格式。

```
sudo apt install yazi
```

使用方法非常简单，只需在终端中输入以下命令即可查看图片：

yazi 设置主题的方式可以访问 [yazi主题仓库](#)

- 例如安装 Tokyo Night 主题:

```
git clone https://github.com/Benny0e/tokyo-night.yazi.git ~/.config/yazi/flavors/tokyo-
```

- 或者 Dracula 主题:

```
ya pkg add yazi-rs/flavors:dracula
```

结语

本教程已经同步到我的个人博客，欢迎访问: [Linux101 - 命令行美化教程](#)