

Curso Completo de Vim Moderno e VimScript

Personalização

Curso Completo de Vim Moderno e VimScript

Plugins são um meio de estender as funcionalidades do vim, há plugins para diversas tarefas, desde wikis para o Vim até ferramentas de auxílio a navegação em arquivos com é o caso do plugin NerdTree que divide uma janela que permite navegar pelos diretórios do sistema a fim de abrir arquivos a serem editados.

1. Instalar o **Vundle** <https://github.com/VundleVim/Vundle.vim>

É um plugin que instala plugins facilmente.

```
mkdir -p ~/.vim/bundle
git clone https://github.com/VundleVim/Vundle.vim.git ~/.vim/bundle/Vundle.vim
vim ~/.vimrc
```

Adicionar ao **vimrc**

```
set nocompatible
filetype off
set rtp+=~/.vim/bundle/Vundle.vim
call vundle#begin()
```

Plugin 'VundleVim/Vundle.vim'

```
call vundle#end()
filetype plugin indent on
```

Abrir o vim e rodar **:PluginInstall** e depois que houver a palavra **Done!** no rodapé do Vim, teste se está funcionando rodando: **:h vundle**, se aparecer a documentação está tudo certo.

Em alguns casos será necessário fechar e abrir de novo o Vim.

A partir de agora todos os plugins podem ser instalados adicionando as seguintes etapas:

Clonar o repositório e salvá-lo dentro do diretório **~/.vim/bundle/[REPOSITÓRIO_DO_PLUGIN]**

Adicionar uma linha abaixo a linha: **Plugin 'VundleVim/Vundle.vim'** com a seguinte regra:

Plugin '[NOME_DO_PERFIL_DO_GITHUB]/[NOME_DO_REPOSITÓRIO_DO_PLUGIN]'

Curso Completo de Vim Moderno e VimScript

2. Instalar o plugin Vim Airline e o conjunto de temas:

Customiza a barra de status do Vim

```
git clone https://github.com/vim-airline/vim-airline ~/.vim/bundle/vim-airline
git clone https://github.com/vim-airline/vim-airline-themes ~/.vim/bundle/vim-airline-themes
```

Abra o `~/.vimrc` e insira esse plugin para o Vundle. Ou seja, insira o conteúdo (**Plugin 'vim-airline/vim-airline'** e **Plugin 'vim-airline/vim-airline-themes'**) abaixo da linha `Plugin 'VundleVim/Vundle.vim'`, ficando assim:

```
Plugin 'VundleVim/Vundle.vim'
Plugin 'vim-airline/vim-airline'
Plugin 'vim-airline/vim-airline-themes'
```

NÃO TÃO NECESSÁRIO, MAS INTERESSANTE: Depois abra o Vim e rode o comando: **:PluginInstall** e aguarde a instalação dos plugins.

Após abrir o Vim, você notará que a barra já estará customizada. Porém sem os caracteres *Powerline*, pra resolver isso abra o `~/.vimrc` e cole esse conteúdo no final do arquivo: vim `~/.vimrc`:

```
"let g:airline_left_sep = '▣'
" powerline symbols
let g:airline_left_sep = '▣'
let g:airline_left_alt_sep = '▣'
let g:airline_right_sep = '▣'
let g:airline_right_alt_sep = '▣'
```

Curso Completo de Vim Moderno e VimScript

Pode ser que no seu caso os **símbolos** que fora de sincronia com o texto, para resolver isso mude a fonte do seu terminal (da mesma família do **Powerline**) e defina um tamanho relativo, por exemplo no meu caso eu alterei pra **Fantasque Sans Mono** e alterei o tamanho da fonte para *15 pixels*.

Lembrando que você pode obter diversas fontes e inclusive a **Fantasque** usando o endereço do **GitHub** do **Terminal Root**, acesse e veja o procedimento para instalar facilmente, isso serve até pra exibir os caracteres do **Powerline** caso pra você esteja com problema de visualização dos símbolos.

Para alterar as cores do tema do Airline, abra o Vim e digite **:AirlineTheme** depois espaço e navegue (pressionando **TAB**) pelos nomes que for aparecendo no Rodapé do Vim, para saber as cores de cada um deles acesse esse link: <https://github.com/vim-airline/vim-airline/wiki/Screenshots> , supondo que você escolheu o tema **violet**, só digitar **:AirlineTheme violet** e depois pressionar **ENTER**.

Se por algum motivo, ao fechar o Vim e reabrir e o tema não for salvo, adicione essa linha ao seu **~/.vimrc**: **let g:airline_theme='<theme>'** , no caso do **violet**: **let g:airline_theme='violet'**

Curso Completo de Vim Moderno e VimScript

3. Instalar plugin para auto completar palavras e sintaxes durante digitação.

Há duas opções disponíveis nesse tutorial: **YouCompleteMe** e **NeoComplete**.

Algumas diferenças entre ambos

- O YouCompleteMe é indicado somente para arquiteturas 64-bit , já o NeoComplete é indicado para todas as arquiteturas: 32-bit e 64-bit;
- Por padrão o YouCompleteMe não auto completa arquivos: .txt, .md ... mas sim .c, .cpp, ... Já o NeoComplete auto completa todos;
- Por padrão o NeoComplete após selecionar precisa pressionar ENTER ou Ctrl+b para usar o selecionado, já o YouCompleteMe ao mover o cursor com as teclas direcionais ele já auto completa.

Fica à sua escolha, eu ainda uso o NeoComplete porque ainda estou aprendendo mais sobre o YouCompleteMe , mas tenho os dois instalados e habilito às vezes o YouCompleteMe também.

Dica: Não deixe os dois habilitados, podem estar instalados, mas desabilitado via comentário [vimscript](#) no [~/.vimrc](#)

3.1 Instalar o plugin YouCompleteMe <https://github.com/Valloric/YouCompleteMe>

3.2 Instalar o plugin NeoComplete <https://github.com/Shougo/neocomplete.vim>

Curso Completo de Vim Moderno e VimScript

4. Instalar o Tema de cores Dracula: <https://draculatheme.com/vim/>

5. Instalar vim-instant-markdown: <https://github.com/suan/vim-instant-markdown>

6. Instalar NerdTREE: <https://github.com/scrooloose/nerdtree.git>

Alguma configurações para NerdTREE:

```
autocmd vimenter * NERDTree

map <C-n> :NERDTreeToggle<CR>

let g:NERDTreeDirArrowExpandable = '►'
let g:NERDTreeDirArrowCollapsible = '▼'

autocmd bufenter * if (winnr("$") == 1 && exists("b:NERDTree") && b:NERDTree.isTabTree()) | q | endif

let g:NERDTreeWinPos = "right"

autocmd vimenter * wincmd p

let NERDTreeQuitOnOpen = 1
```

Curso Completo de Vim Moderno e VimScript

7. Instalar vim-devicons e vim-webdevicons

Adicionar ambas linhas ao *Vundle* no seu *~/.vimrc*

```
Plugin 'ryanoasis/vim-devicons'  
Plugin 'cj/vim-webdevicons'
```

E também as linhas ao final do *~/.vimrc* :

```
set encoding=utf8  
let g:airline_powerline_fonts = 1
```

Abra o vim e rode o comando: *:PluginInstall* e espere a palavra *Done!* no rodapé do Vim.

Lembrando que você precisa ter alguma *fonte da família*, se quiser uma instalação rápida de *fonts*, use:

```
git clone https://github.com/terroo/fonts  
cd fonts/fonts  
mv *.ttf ~/.local/share/fonts  
mv *.otf ~/.local/share/fonts  
fc-cache -fv
```

GitHub dos plugins:

<https://github.com/ryanoasis/vim-devicons/wiki/Installation>
<https://github.com/cj/vim-webdevicons>

Curso Completo de Vim Moderno e VimScript

Outros Plugins

Conheça o site: <https://vimawesome.com/> para ter acesso a mais plugins interessantes e que facilitam escrever seus códigos mais rápidos , além da página: <https://www.vim.org/scripts/> para ainda mais scrips e plugins.

Alguns outros que você pode instalar e são interessantes:

- **Vim Startfy** <https://github.com/mhinz/vim-startify> - Cria uma tela inicial com arquivos mais recentes ... ;
- **Auto Pairs** <https://github.com/jiangmiao/auto-pairs> - Adiciona automaticamente um par de coolchetes, agudos,... ao digitar, alternativamente você ainda pode instalar o plugin Vim Surround que age de forma similar: <https://github.com/tpope/vim-surround>;
- **NERD Commenter** <https://github.com/scrooloose/nerdcommenter> - Deixa ainda mais fácil comentar múltiplas linhas;
- **Vim Multiple Cursors** <https://github.com/terryma/vim-multiple-cursors> - Edite a mesma combinação de caracteres em múltiplas linhas;
- **Ctrl P** <https://github.com/ctrlpvim/ctrlp.vim> - Abra diretórios facilmente pressionando Ctrl + P;
- **Ale** <https://github.com/w0rp/ale> - Identifica automaticamente as linhas que estão com sintaxes incorretas;
- **Emmet** <https://github.com/mattn/emmet-vim> - Altere várias linhas, bom para criar facilmente listas HTML, por exemplo.