Guía paso a paso para configurar Neovim con init.vim

Este documento detalla paso a paso cómo configurar Neovim en Ubuntu con un entorno moderno, incluyendo navegación de archivos, autocompletado inteligente, búsqueda avanzada y buena indentación.

# 1. Instalar dependencias del sistema

- ripgrep: sudo apt install ripgrep  
- fd: sudo apt install fd-find  
- Crear alias si es necesario:  
 mkdir -p ~/.local/bin  
 ln -s $(which fdfind) ~/.local/bin/fd  
 Agregar ~/.local/bin al PATH en tu .bashrc o .zshrc

# 2. Instalar plugin manager (vim-plug)

curl -fLo ~/.local/share/nvim/site/autoload/plug.vim --create-dirs \  
 https://raw.githubusercontent.com/junegunn/vim-plug/master/plug.vim

# 3. Editar init.vim

Ubicación del archivo: ~/.config/nvim/init.vim

## Contenido completo del init.vim

call plug#begin('~/.vim/plugged')  
  
Plug 'nvim-tree/nvim-tree.lua'  
Plug 'nvim-tree/nvim-web-devicons'  
Plug 'neoclide/coc.nvim', {'branch': 'release'}  
Plug 'morhetz/gruvbox'  
Plug 'folke/tokyonight.nvim'  
Plug 'nvim-lua/plenary.nvim'  
Plug 'nvim-telescope/telescope.nvim'  
  
call plug#end()  
  
syntax on  
set number  
set relativenumber  
set termguicolors  
colorscheme tokyonight  
  
if has("autocmd")  
 au BufReadPost \*  
 \ if line("'\"") > 0 && line("'\"") <= line("$") |  
 \ exe "normal! g'\"" |  
 \ endif  
endif  
  
set autoindent  
set smartindent  
set tabstop=4  
set shiftwidth=4  
set softtabstop=4  
set expandtab  
set whichwrap+=<,>,h,l  
  
lua << EOF  
require("nvim-tree").setup()  
EOF  
nnoremap <C-n> :NvimTreeToggle<CR>  
  
lua << EOF  
require('telescope').setup{}  
EOF  
  
let mapleader = " "  
nnoremap <leader>ff :Telescope find\_files<CR>  
nnoremap <leader>fg :Telescope live\_grep<CR>  
nnoremap <leader>fb :Telescope buffers<CR>  
nnoremap <leader>fh :Telescope help\_tags<CR>  
  
inoremap <silent><expr> <TAB>  
 \ coc#pum#visible() ? coc#pum#next(1) :  
 \ CheckBackspace() ? "\<Tab>" :  
 \ coc#refresh()  
inoremap <silent><expr> <S-TAB> coc#pum#visible() ? coc#pum#prev(1) : "\<C-h>"  
  
function! CheckBackspace() abort  
 let col = col('.') - 1  
 return !col || getline('.')[col - 1] =~# '\s'  
endfunction  
  
nmap <silent> gd <Plug>(coc-definition)  
nmap <silent> K :call CocActionAsync('doHover')<CR>  
nmap <leader>rn <Plug>(coc-rename)