

Neovim setup steps

kitevnn

1. Screenshot Preview

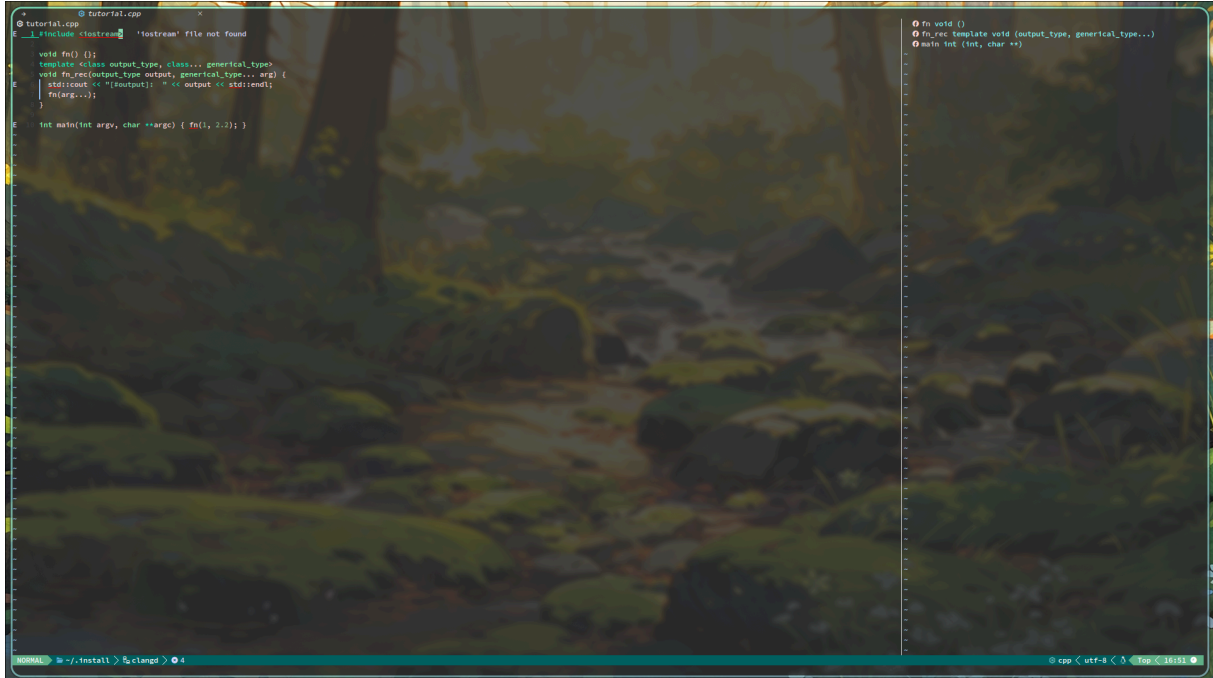


Figure 1: Screenshot Preview 1

2. How to Setup?

2.1 File Structure

```
.
├── init.lua
├── lazy-lock.json
├── lua
│   └── custom
│       ├── basic
│       │   ├── basic-filetype.lua
│       │   ├── basic-keybindings-conf.lua
│       │   ├── basic-macro-conf.lua
│       │   └── basic-settings-conf.lua
│       ├── lsp
│       │   ├── ls-clangd.lua
│       │   ├── ls-gopls.lua
│       │   ├── ls-rustanalyzer.lua
│       │   └── ls-texlab.lua
│       ├── neovide
│       │   └── neovide-conf.lua
│       ├── plugin
│       │   ├── plug-bufferline.lua
│       │   ├── plug-cmp.lua
│       │   ├── plug-cmp-window.lua
│       │   ├── plug-comment.lua
│       │   ├── plug-dashboard.lua
│       │   ├── plug-format.lua
│       │   ├── plug-gitsigns.lua
│       │   ├── plug-ibl.lua
│       │   ├── plug-icon.lua
│       │   ├── plug-line.lua
│       │   ├── plug-lsp-progress.lua
│       │   ├── plug-mini.lua
│       │   ├── plug-navigation-bar.lua
│       │   ├── plug-noice.lua
│       │   ├── plug-nvim-lspconfig.lua
│       │   ├── plug-outline.lua
│       │   ├── plug-preview-markdown.lua
│       │   ├── plug-preview-typst.lua
│       │   ├── plug-telescope.lua
│       │   ├── plug-tree.lua
│       │   ├── plug-TS.lua
│       │   ├── plug-ui.lua
│       │   ├── plug-virtualtext.lua
│       │   └── plug-workspaces.lua
│       ├── themes
│       │   └── theme-forest.lua
│       └── util
│           ├── util-keybindings.lua
│           └── util-lspconfig.lua
```

2.2 Installation

Use the following command to install it.

```
# install neovim via zypper
#
# S | Name | Summary | Type
# ---+-----+-----+-----
# i+ | neovim | Vim-fork focused on extensibility and agility | Package
zypper in neovim # zypper install with sudo permission.

# git clone it.
git clone https://github.com/kitevnn/dotfiles

# setup configuration file
cd ~/.config
mkdir -p nvim && cd nvim

# and put the folder of nvim manually here

# then, type nvim in TERMINAL and waiting for installation.

# well, I recognize there is a missing step here,
# which is that there is a theme of lua-luacheck (aka statusline) that is not installed.

# Here is the steps of installing that lua-luacheck theme.
cd nvim/lua/custom/themes/
mv Forest.lua ~/.local/share/nvim/lazy/lua-luacheck.nvim/lua/lua-luacheck/themes/

# ok, done.
```

3. About Usage

3.1 Keybindings

TL;DR;

num	modes	Keybindings	Description
1	normal	S	Save file
2	normal	Q	Quit Neovim
3	normal	sj	Split screen down
4	normal	sk	Split screen up
5	normal	sh	Split screen left
6	normal	sl	Split screen right
7	normal	J	Move down 5 lines
8	normal	K	Move up 5 lines
9	normal	M	Open file-manager tree
10	normal	down	Resize screen down 5 units
11	normal	up	Resize screen up 5 units
12	normal	left	Resize screen left 5 units
13	normal	right	Resize screen right 5 units
14	normal	ESC ESC	Exit termin mode
15	normal	[w	Move one line of text up
16	normal]w	Move one line of text down
17	visual	[w	Move one line of text up
18	visual]w	Move one line of text down
19	normal	[h	Move the next git-hunk
20	normal]h	Move the prev git-hunk
21	insert	Ctrl e	Input binary-characters prefix key
22	visual	leader S	Search for this word under visual mode
23	visual	leader C	Copy content to system clipboard
24	normal	leader P	Paste content from system clipboard
25	normal	leader il	Disable indent blankline
26	normal	leader b	Toggle breadcrumb
27	normal	leader wl	List workspace
28	normal	leader wa	Add workspace
29	normal	leader wo	Open workspace
30	normal	leader wm	Remove workspace
31	normal	leader wr	Rename workspace
32	normal	leader pl	Start texlab LS
33	normal	leader pc	Start clangd LS
34	normal	leader pa	Start lua_ls LS
35	normal	leader pg	Start gopls LS

36	normal	leader pt	Start typst-lsp LS
37	normal	leader s	Open outline (It needs open LS first)
38	normal	leader l	Stop LS
39	normal	leader ot	Build LaTeX file
40	normal	leader leader I	Sync all plugins with lazy.vim
41	normal	leader yp	Preview markdown file with charmbracelet/glow
42	normal	leader yx	Toggle markdown file with charmbracelet/glow
43	normal	leader yt	Preview markdown file with chomosuke/typst-preview.nvim
44	normal	leader gh	Gitsigns setloclist
45	normal	leader gv	Gitsigns select_hunk
46	normal	leader ff	Telescope find_files
47	normal	leader fg	Telescope live_grep
48	normal	leader fb	Telescope buffers
49	normal	leader ft	Telescope help_tags
50	normal	leader 1	Move Numbered 1 tab
51	normal	leader 2	Move Numbered 2 tab
52	normal	leader 3	Move Numbered 3 tab
53	normal	leader 4	Move Numbered 4 tab
54	normal	leader 5	Move Numbered 5 tab
55	normal	leader 6	Move Numbered 6 tab
56	normal	leader 7	Move Numbered 7 tab
57	normal	leader 8	Move Numbered 8 tab
58	normal	leader 9	Move Numbered 9 tab
59	normal	leader xx	Pick to close a tab with the starting character
60	normal	leader xh	Close ALL tabs on the left side of the current tab page
61	normal	leader xl	Close ALL tabs on the right side of the current tab page
62	insert	Ctrl f	About cmp: the preview box for cmp defaults drop by 4 units
63	insert	Ctrl b	About cmp: the preview box for cmp defaults rise by 4 units
64	insert	Ctrl j	About cmp: select next cmp item
65	insert	Ctrl k	About cmp: select prev cmp item
66	insert	Ctrl l	About cmp: show cmp item
67	insert	Ctrl h	About cmp: abort cmp
68	insert	Ctrl i	About cmp: confirm current cmp item
69	normal	gD	About cmp-lsp: go to declaration
70	normal	gd	About cmp-lsp: go to definition
71	normal	gh	About cmp-lsp: show hover info
72	normal	gi	About cmp-lsp: go to implementation
73	normal	gk	About cmp-lsp: show signature_help
74	normal	gr	About cmp-lsp: go to references
75	normal	[d	About cmp-lsp: goto prev diagnostic

76	normal]d	About cmp-lsp: goto next diagnostic
77	normal	leader e	About cmp-lsp: open_float
78	normal	leader pq	About cmp-lsp: show all diagnostic setloclist
79	normal	leader ca	About codeaction: show codeaction with ui (needs LS)
80	visual	leader ca	About codeaction: show codeaction with ui (needs LS)
81	normal	leader ft	About formatting: formatting all codes (needs LS)

3.2 About Language-Server

3.2.1 LS that are now supported

We could make a further configurations via `~/.config/nvim/lua/custom/lsp/ls-*` here.

- texlab
- gopls(WIP)
- clangd
- rust-analyzer

3.2.2 How I could add a new LS with completion plugin(for example, nvim-cmp)?

Well, There is a **four-step** process here.

- First of all, please **make sure** that the servers you are using is in the lists of `nvim-lspconfig` LS and `nvim-cmp` now and see `:h lspconfig-all` for the entire LS lists.
- Then, It's best to go through the package manager given by your linux distro and install the LS you want to use. [mason](#) is also a great choice to manage LS. And here is the [language-server lists official website](#)
- Next, We need to touch a new file named "ls-LANGUAGESERVERYOUUSE.lua" `~/.config/nvim/lua/lsp/` here (Of course, the naming convention is my own, and I'm just doing it here for the sake of uniform naming)

1. for example, Let's assume that we need to install the rust LS named `texlab`
2. ok, the next step is that `zypper` in `texlab` on SuSE Linux Tumbleweed. (To be honest, I don't know if other distributions have this package for example, because I haven't tested it either, but in general it should be, after all, Archlinux packages are relatively the fastest and most up-to-date)
3. well, we touch to create a new file named `ls-texlab.lua` and put it `~/.config/nvim/lua/custom/lsp/` there.
4. then, we should take a glance with `:h lspconfig-all` for further help, which provided the simplest installation case. However, it does not provide as many configuration items as it does in the repository source code. So if we want to do a full configuration, we'd better go to [neovim/nvim-lspconfig](#) to see the full configuration.

5. then, we need to add a simple code in `~/.config/nvim/init.lua`

```
-- IMPORTANT: Added a simple code after "_M.setup()"

-- _M.setup()
require('custom.lsp.ls-texlab')
```

- Next, We need to create a `cmp` instance that forward the INFO produced by the LS to the completion box `~/.config/lua/custom/plugin/plugin-cmp.lua` there.

So, there is a **reproducible code**.

```
-- the LANGUAGESEVERYOUUSE could be found https://github.com/neovim/nvim-
lspconfig/blob/master/doc/server_configurations.md
require("lspconfig")['LANGUAGESEVERYOUUSE'].setup {
  capabilities = capabilities,
  vim.keymap.set('n', '<leader>e', vim.diagnostic.open_float),
  vim.keymap.set('n', '[d', vim.diagnostic.goto_prev),
  vim.keymap.set('n', ']d', vim.diagnostic.goto_next),
  vim.keymap.set('n', '<leader>pq', vim.diagnostic.setloclist),
  vim.api.nvim_create_autocmd('LspAttach', {
    group = vim.api.nvim_create_augroup('UserLspConfig', {}),
    callback = function(ev)
      vim.bo[ev.buf].omnifunc = 'v:lua.vim.lsp.omnifunc'
      local opts = { buffer = ev.buf }

      -- new
      vim.keymap.set('n', 'gh', vim.lsp.buf.hover, opts)
      vim.keymap.set('n', 'gd', vim.lsp.buf.definition, opts)
      vim.keymap.set('n', 'gi', vim.lsp.buf.implementation, opts)
      vim.keymap.set('n', 'gk', vim.lsp.buf.signature_help, opts)
      vim.keymap.set('n', 'gr', vim.lsp.buf.references, opts)
      vim.keymap.set('n', 'gD', vim.lsp.buf.declaration, opts)
      vim.keymap.set({ 'n', 'v' }, '<leader>ca', vim.lsp.buf.code_action, opts)
      vim.keymap.set('n', '<leader>ft', function()
        vim.lsp.buf.format { async = true }
      end, opts)
    end,
  })
}
```

- OK, done

1. If there are no other errors, you should be able to start the LS in the corresponding code file. Sometimes, due to the quality of the LS, services such as highlighting, outlines, jumps, etc. may not be available, but it is important to note that you can change to a new LS.

4. Some Restriction

Some Restriction

- **Only support Neovim TUI mode**, sorry :(
 1. Because I don't turn on 256 color here (`:set notermguicolors<CR>`), it's because maybe my theme has some vimscript code that isn't fully set up. And if I turn on it, it will cause my personal colorscheme to be messed up and temporarily unadjustable.
- **The theme file must be set in the list of plugins**

1. Like the one mentioned above.

```
# well, I recognize there is a missing step here,  
# which is that there is a theme of lua-line(aka statusline) that is not  
installed.
```

```
# Here is the steps of installing that lua-line theme.
```

```
cd nvim/lua/custom/themes/
```

```
mv Forest.lua ~/.local/share/nvim/lazy/lua-line.nvim/lua/lua-line/themes/
```

```
# ok, done.
```


5. PostScript

PostScript

- This is my first repository submission on github, and I hope you could understand properly if there is anything unfriendly to you. I'm sorry.

Happy Use ! _(:3」 ∠)_