

# Kickstart docs

*Thomas Wells*

## 1. Purpose

Although the `kickstart.nvim` documentation will not be as useful outside of its original context, I am moving it here in the interest of maintaining a cleaner `init.lua`.

## 2. Lazy

Use `opts = {}` to automatically pass options to a plugin's `setup()` function, forcing the plugin to be loaded. Alternatively, use `config = function() ... end` for full control over the configuration. Therefore, if you prefer to call `setup` explicitly, use:

```
{
  'lewis6991/gitsigns.nvim',
  config = function()
    require('gitsigns').setup({
      -- Your gitsigns configuration here
    })
  end,
}
```

Plugins can specify dependencies. The dependencies are proper plugin specifications as well - anything you do for a plugin at the top level, you can do for a dependency. Use the `dependencies` key to specify the dependencies of a particular plugin

## 3. Telescope

Telescope is a fuzzy finder that comes with a lot of different things that it can fuzzy find! It's more than just a "file finder", it can search many different aspects of Neovim, your workspace, LSP, and more!

The easiest way to learn Telescope is: `:Telescope help_tags`. After running this command, a window will open up and you're able to type in the prompt window. You'll see a list of `help_tags` options and a corresponding preview of the help.

Two important keymaps to use while in Telescope are:

- Insert mode: `<C- />`
- Normal mode: `?`

More setup info can be found in `:help telescope.setup()`

## 4. LSP

Brief aside: **What is LSP?**

LSP is an initialism you've probably heard, but might not understand. LSP stands for Language Server Protocol. It's a protocol that helps editors and language tooling communicate in a standardized fashion.

In general, you have a "server" which is some tool built to understand a particular language (such as `gopls`, `lua_ls`, `rust_analyzer`, etc.). These Language Servers (sometimes called LSP servers, but that's kind of like ATM Machine) are standalone processes that communicate with some "client" - in this case, Neovim!

LSP provides Neovim with features like:

- Go to definition

- Find references
- Autocompletion
- Symbol Search
- and more!

Thus, Language Servers are external tools that must be installed separately from Neovim. This is where `mason` and related plugins come into play. If you're wondering about `lsp` vs `treesitter`, you can check out the wonderfully and elegantly composed help section, `:help lsp-vs-treesitter`