

direnv: manage dev environments

`direnv` (along with [nix-direnv](#)) allows one to persist ¹ nix **development shell** environments and share them seamlessly with text editors and IDEs. It obviates having to run `nix develop` manually every time you open a new terminal. The moment you `cd` into your project directory, the devshell is automatically activated, thanks to `direnv`.

Starship

It is recommended to use [starship](#) along with `nix-direnv`, because it gives a visual indication of the current environment. For example, if you are in a **Nix Shell**, your terminal prompt automatically changes to something like this:

```
srid on nixos haskell-template on  master [!] via λ 9.2.6 via ❄ impure  
>
```

Setup

direnv in 60 seconds



If you use [home-manager](#), both `nix-direnv` and `starship` can be installed using the following configuration:

```
# home.nix
programs.direnv = {
  enable = true;
  nix-direnv.enable = true;
};
programs.starship = {
  enable = true;
};
```

Newcomer Tip

If you have never used [home-manager](#) before, we recommend that you set it up by following the instructions at <https://github.com/juspay/nixos-unified-template> (which is based on [nixos-unified](#), thus works on macOS and Linux).

Text Editor configuration

VSCode

For VSCode, use [Martin Kühl's direnv extension](#).

Doom Emacs

Doom Emacs has the `:tools direnv module` to automatically load the devshell environment when you open the project directory.

Add a `.envrc`

To enable direnv on Flake-based projects, add the following to your `.envrc`:

```
use flake
```

Now run `direnv allow` to authorize the current `.envrc` file. You can now `cd` into the project directory in a terminal and the devshell will be automatically activated.

Reload automatically when some files change

Haskell – when `.cabal` files change

Since both `nixpkgs` and `haskell-flake`[↗] use Nix expressions that read the `.cabal` file to get dependency information, you will want the devshell be recreated every time a `.cabal` file changes. This can be achieved using the `watch_file` function. Modify your `.envrc` to contain:

```
watch_file *.cabal
use flake
```

As a result of this whenever you change a `.cabal` file, direnv will reload the environment. If you are using VSCode, you will see a notification that the environment has changed, prompting you to restart it (see [example](#)[↗]).

External Links

- [Effortless dev environments with Nix and direnv](#)[↗]

Footnotes

1.

`nix-direnv`[↗] prevents garbage collection of the devshell, so you do not have to re-download things again. direnv also enables activating the devshell in your current shell, without needing to use a customized bash.



Links to this page

VSCode

Setup direnv and install the direnv VSCode extension[↗].

If you use direnv, it is rather simple to get setup with VSCode:

Using home-manager to manage dotfiles, packages, services

Among the various examples, we will showcase how to configure Neovim declaratively in Nix, and have it work across platforms. In addition, we will show how to use direnv to develop projects uniformly across different machines.

Nix for Development

direnv: manage dev environments

