



(<https://www.hashicorp.com>)

Command: `fmt`

The `terraform fmt` command is used to rewrite Terraform configuration files to a canonical format and style. This command applies a subset of the Terraform language style conventions (</docs/language/syntax/style.html>), along with other minor adjustments for readability.

Other Terraform commands that generate Terraform configuration will produce configuration files that conform to the style imposed by `terraform fmt`, so using this style in your own files will ensure consistency.

The canonical format may change in minor ways between Terraform versions, so after upgrading Terraform we recommend to proactively run `terraform fmt` on your modules along with any other changes you are making to adopt the new version.

We don't consider new formatting rules in `terraform fmt` to be a breaking change in new versions of Terraform, but we do aim to minimize changes for configurations that are already following the style examples shown in the Terraform documentation. When adding new formatting rules, they will usually aim to apply more of the rules already shown in the configuration examples in the documentation, and so we recommend following the documented style even for decisions that `terraform fmt` doesn't yet apply automatically.

Formatting decisions are always subjective and so you might disagree with the decisions that `terraform fmt` makes. This command is intentionally opinionated and has no customization options because its primary goal is to encourage consistency of style between different Terraform codebases, even though the chosen style can never be everyone's favorite.

We recommend that you follow the style conventions applied by `terraform fmt` when writing Terraform modules, but if you find the results particularly objectionable then you may choose not to use this command, and possibly choose to use a third-party

formatting tool instead. If you choose to use a third-party tool then you should also run it on files that are generated automatically by Terraform, to get consistency between your hand-written files and the generated files.

Usage

Usage: terraform fmt [options] [DIR]

By default, `fmt` scans the current directory for configuration files. If the `dir` argument is provided then it will scan that given directory instead. If `dir` is a single dash (`-`) then `fmt` will read from standard input (STDIN).

The command-line flags are all optional. The list of available flags are:

- `-list=false` - Don't list the files containing formatting inconsistencies.
- `-write=false` - Don't overwrite the input files. (This is implied by `-check` or when the input is STDIN.)
- `-diff` - Display diffs of formatting changes
- `-check` - Check if the input is formatted. Exit status will be 0 if all input is properly formatted and non-zero otherwise.
- `-recursive` - Also process files in subdirectories. By default, only the given directory (or current directory) is processed.