# Contributing to darktable

The darktable project welcomes contributions:

\* [Code](https://www.darktable.org/development/)

\* [Documentation](https://www.darktable.org/resources/)

\* Testing (and any backtraces if you happpen to crash darktable)

\* Translations

\* [Camera profiles](https://www.darktable.org/resources/camera-support/).

\* Tutorials, screencasts, etc.

See the [darktable development page](https://www.darktable.org/development/) for

more information.

## Code

Before you spend a lot of time working on a new feature, it's always best to

discuss your proposed changes with us first. The best place to do that is in

our IRC channel on \*\*irc.freenode.net\*\*, channel \*\*#darktable\*\* or the

development mailing list, [see here for more

information](https://www.darktable.org/contact/). This will dramatically

improve your chances of having your code merged, especially if we think you'll

hang around to maintain it.

### Coding style

We like our code to be properly formatted. We have a well-defined coding style,

and [.clang-format](.clang-format) style file represents it fully.

You can enforce your commits to follow it:

1. Install [clang-format](http://clang.llvm.org/docs/ClangFormat.html) clang tool. Probably, any version will be ok, but the newer the better.

2. You'll need to integrate `git` and `clang-format`.

\* For that, you will need to download `git-clang-format` from [here](https://github.com/llvm-mirror/clang/blob/master/tools/clang-format/git-clang-format) or [here](https://llvm.org/svn/llvm-project/cfe/trunk/tools/clang-format/git-clang-format).

\* Read it to check for nastiness.

\* Warning: apparently, it only works with Python2, and does not work with Python3!

\* Put it somewhere in your path (e.g. `~/bin` or `/usr/local/bin`) and ensure that it is executable (`chmod +x`).

3. Now, step into your local clone of repository:

\* `cd darktable/.git/hooks`

\* If you previously did not have a `pre-commit` hook:

\* `cp pre-commit.sample pre-commit && chmod +x pre-commit`

\* Open `pre-commit` with your favourite text editor, and append before the last block, here is how the end should look:

```

# format everything

res=$(git clang-format --diff | wc -l)

if [ $res -ne 1 ]

then

git clang-format

echo "Commit did not match clang-format"

exit 1;

fi

# If there are whitespace errors, print the offending file names and fail.

exec git diff-index --check --cached $against --

```

\* Also, there is a [Coding Style](https://redmine.darktable.org/projects/darktable/wiki/Coding\_Style) page on our redmine wiki.