

# Code formatting a Python project in 2022

Isaac Bythewood · 2022-07-30 · 2 min read

coding

## Code formatting a Python project in 2022

For those who want a quick solution without reading all of PEP 8. The Black Python module has a fully automated solution for you.

It's not always worth hand formatting every line of code. You can dramatically increase the speed at which you write code by ignoring formatting entirely and making use of code formatters like [Black for Python](#). I've also found that using Black makes Git commits and diffs much cleaner by removing human inconsistency from the equation. To get straight to the point I install Black, Flake8, and isort on all of my projects. You can use your preferred Python package manager, I use pipenv and add these dependencies to my Pipfile under `[dev-packages]`.

```
# Pipfile

[[source]]
url = "https://pypi.org/simple"
verify_ssl = true
name = "pypi"

[packages]

[dev-packages]
black = "*"
flake8 = "*"
isort = "*"

[requires]
python_version = "3.10"
```

You'll then want to make a file named `.isort.cfg` and configure isort to use black's formatting style so that when you sort dependencies you aren't bouncing between the two's opinionated styles.

```
# .isort.cfg

[settings]
profile = black
```

I then configure Flake8 in the file `.flake8` to make sure it's not complaining about Black's opinionated styling.

```
# .flake8

[flake8]
max-line-length = 88
extend-ignore = E203
```

As a small bonus if you use Visual Studio Code you can install the [Python Extension](#) and get built in formatting support. I use three extra config lines to enable Flake8, add Black formatting, and auto format on save.

```
{
  "python.formatting.provider": "black",
  "python.linting.flake8Enabled": true,
  "editor.formatOnSave": true
}
```

If you are using these tools manually you can run Black, isort, or Flake8 directly on a file. With Pipenv you'd use something like `pipenv run black python_file.py` and have your code automatically formatted. These are all very popular tools though and I suggest taking a look at your IDE's documentation use them how your IDE suggests. I've also added `lint` and `format` commands with a `Makefile` too if you'd like an solution that isn't tied to an IDE.

That's all you need to do to have well formatted Python code in 2022. Let modern tooling do the work for you.