

# FORMAT | LINT | PRE-COMMIT



 [linkedin.com/in/arthurio](https://linkedin.com/in/arthurio)

 [notivize.com](https://notivize.com)

 [github.com/arthurio/pre-commit](https://github.com/arthurio/pre-commit)

# TOOLS



# FORMATTERS 🧼 ✨

**isort**: isort your imports, so you don't have to.

**black**: The Uncompromising Code Formatter.

**flynt**: String formatting converter.

# LINTERS

**bandit**: Find common security issues.

**flake8**: Your Tool For Style Guide Enforcement.

**flake8-docstrings**: Flake8 plugin to include checks provided by pep257.

**mypy**: Static type checker.

# AUTOMATION



**pre-commit**: A framework for managing and maintaining multi-language pre-commit hooks.

# CONFIGURATION



Keep it to a minimum

# BLACK

```
# pyproject.toml

[tool.black]
line-length = 120
```



# ISORT

```
# pyproject.toml

[tool.isort]
profile = "black"
line_length = 120
skip_gitignore = true
```

# FLYNT

Nothing to see here 🙄 ...

# BANDIT

```
# .bandit

[bandit]
targets: src,tests
skips: B101
```

- B101: assert used

## Speaker notes

- ⚠ Don't confuse the "ini" file **AND/OR** the "yaml" config file. ⚠
- assert is removed with compiling to optimized byte code.
- Bandit likes to complain about random, use the secrets module instead if you can or silent individual lines with #  
nosec: B311

# FLAKE8/FLAKE8-DOCSTRINGS

```
# .flake8

[flake8]
max_line_length = 120
ignore = D1
docstring-convention = google
```

- **W503**: Line break occurred before a binary operator
- **D1**: Missing docstring

## Speaker notes

- I don't care about missing docstrings as I try to keep them to a minimum. Instead I prefer to rely on explicit variable and function names, as well as type hints.
- W503 is not compatible with black and goes against PEP 8.

# MYPY

```
# mypy.ini
```

```
[mypy]
```

```
mypy_path = ./src:./tests
```

## Speaker notes

Mypy doesn't support pyproject.toml configuration, only mypy.ini and setup.cfg.



# PRE-COMMIT

```
repos:
  - repo: local
    hooks:
      - id: isort
        name: isort
        entry: isort
        types: [python]
        language: python
      - id: flynt
        name: flynt
        entry: flynt
        args: [--fail-on-change]
        types: [python]
        language: python
      - id: black
        name: black
        entry: black
        language: python
        types: [python]
        exclude: node_modules
      - id: flake8
        name: flake8
        entry: flake8
        language: python
        types: [python]
        exclude: .serverless,node_modules
      - id: bandit
        name: bandit
        entry: bandit
        language: python
        types: [python]
        args: [--ini, .bandit, --recursive]
      - id: mypy
        name: mypy
        entry: mypy
        types: [python]
        language: python
  - repo: https://github.com/pre-commit/pre-commit-hooks
    rev: v4.0.1
    hooks:
      - id: end-of-file-fixer
        exclude: .* (min.js|min.css|html|svg|css.map|js.map)
      - id: trailing-whitespace
        exclude: .* (md)
```

## Speaker notes

Notice that everything is setup as "local" because I keep track of the packages (black, flake8, etc.) in my dev

- Tools are installed for IDEs instead of separate virtual environment just for pre-commit.
- You can have dependabot keep your tools up to date.

requirements. Benefits are:

- Keeps things in sync between dev environment and pre-commit.

# DEV REQUIREMENTS

```
# dev-requirements.in  
  
bandit==1.7.0  
black==21.4b2  
flake8-docstrings==1.6.0  
flake8==3.9.1  
flynt==0.64  
isort==5.8.0  
mypy==0.812  
pre-commit==2.12.1
```

## Speaker notes

We use pip-tools to compile our requirements.txt files with pinned libraries for dependencies.

# USAGE

# SETUP YOUR IDE/EDITOR



# INSTALL PRE-COMMIT HOOK

```
pre-commit install  
git add .  
git commit -m "Some bad code"
```

## Speaker notes

- You can configure pre-commit to use any sort of hooks, pre/post push, pre/post message, etc.
- pre-commit runs only for the staged files, not the entire project. And only relevant linters/formatters are applied.



```

~/code/pre-commit  pysprings-pre-commit  main  git commit -m "Move tests and add presentation"
10120 03:12:34
[INFO] Installing environment for https://github.com/pre-commit/pre-commit-hooks.
[INFO] Once installed this environment will be reused.
[INFO] This may take a few minutes...
isort.....Passed
flynt.....Passed
black.....Passed
flake8.....Failed
- hook id: flake8
- exit code: 1

src/isort/test.py:2:1: F401 'sys' imported but unused
src/isort/test.py:8:11: F821 undefined name 'password'
src/isort/test.py:9:5: F841 local variable 'access_token' is assigned to but never used
src/isort/test.py:10:5: F841 local variable 'password' is assigned to but never used
src/isort/test.py:13:121: E501 line too long (121 > 120 characters)

bandit.....Passed
mypy.....Failed
- hook id: mypy
- exit code: 1

src/bandit/test.py:2: error: Missing return statement
src/isort/test.py:4: error: Cannot find implementation or library stub for module named 'utils'
src/isort/test.py:4: note: See https://mypy.readthedocs.io/en/latest/running\_mypy.html#missing-imports
src/isort/test.py:8: error: Cannot determine type of 'password'
src/isort/test.py:17: error: Incompatible return value type (got "None", expected "str")
src/isort/test.py:19: error: Incompatible return value type (got "int", expected "str")
src/isort/test.py:22: error: Argument 1 to "test" has incompatible type "str"; expected "int"
src/isort/test.py:23: error: Argument 1 to "test" has incompatible type "Optional[str]"; expected "int"
src/isort/test.py:27: error: Argument 1 to "int" has incompatible type "Optional[str]"; expected "Union[str, SupportsInt, _SupportsIndex, _SupportsTrunc]"
Found 8 errors in 2 files (checked 4 source files)

Fix End of Files.....Failed
- hook id: end-of-file-fixer
- exit code: 1
- files were modified by this hook

Fixing presentation/dist/theme/blood.css
Fixing presentation/dist/theme/sky.css
Fixing presentation/dist/theme/moon.css
Fixing presentation/dist/theme/black.css
Fixing presentation/dist/theme/beige.css
Fixing presentation/dist/theme/serif.css
Fixing presentation/dist/theme/solarized.css
Fixing presentation/dist/theme/night.css
Fixing presentation/dist/theme/white.css
Fixing presentation/dist/theme/simple.css
Fixing presentation/dist/theme/league.css
Fixing presentation/dist/reveal.css

Trim Trailing Whitespace.....Passed

```

## Speaker notes

- Linters only show errors.
- Formatters automatically fix the code so you might have unstaged changes.

```
~/code/ [REDACTED] [REDACTED] [REDACTED] git commit
isort.....Passed
flynt.....Passed
black.....Passed
flake8.....Passed
bandit.....Passed
mypy.....Passed
Fix End of Files.....Passed
Trim Trailing Whitespace.....Passed
```

```
~/code/pre-commit  pysprings-pre-commit  main  git commit -m "Add updated css files and usage"
git isort.....(no files to check)Skipped
flynt.....(no files to check)Skipped
black.....(no files to check)Skipped
flake8.....(no files to check)Skipped
bandit.....(no files to check)Skipped
mypy.....(no files to check)Skipped
Fix End of Files.....pPassed
Trim Trailing Whitespace.....usPassed
h[main 7226ce6] Add updated css files and usage
 14 files changed, 20 insertions(+), 13 deletions(-)
 create mode 100644 presentation/images/git_commit_fail.png
```

# CONTINUOUS INTEGRATION

```
pre-commit run --all-files
```

## Speaker notes

Devs can skip hooks with `git commit --no-verify`.

# Q&A



[linkedin.com/in/arthurio](https://www.linkedin.com/in/arthurio)



[notivize.com](https://notivize.com)



[github.com/arthurio/pre-commit](https://github.com/arthurio/pre-commit)