Ruff Python lint and format

Config pyproject.toml / ruff.toml → `select rules` → **ruff check** violations → `--fix` → **ruff format** layout → `--check in CI` → **Clean tree** commit

CONCEPTS

Two tools in one binary: check lints (and fixes), format rewrites layout. Run both; they do not overlap.

Since 0.17 the default is a broad curated set (F, E4/E7/E9, B, UP, SIM, RUF, PL and more). See it with ruff check --show-settings --isolated; select replaces it, extend-select adds.

CHECK

`ruff` **`check`**
Lint current directory

`ruff` **`check`** *`<path>...`*
Lint given files or dirs

`ruff` **`check --fix`**
Apply safe fixes

`ruff` **`check --diff`**
Preview fixes as a diff

`ruff` **`check --unsafe-fixes --fix`**
Also apply unsafe fixes

`ruff` **`check --show-fixes`**
List what was fixed

`ruff` **`check --fix-only`**
Fix, do not report leftovers

`ruff` **`check --watch`**
Re-run on file changes

FORMAT

`ruff` **`format`**
Format current directory

`ruff` **`format`** *`<path>...`*
Format given files or dirs

`ruff` **`format --check`**
Fail if anything would change

`ruff` **`format --diff`**
Show changes, write nothing

`ruff` **`format --line-length`** *`<n>`*
Override line length

Format suppression comments

Comment	Effect
`# fmt: off / # fmt: on`	skip a region
`# fmt: skip`	skip one statement

RULE SELECTION

`--select` *`<codes>`*
Enable only these rules

`--extend-select` *`<codes>`*
Add rules to the current set

`--ignore` *`<codes>`*
Disable rules

`--per-file-ignores` *`<glob>:<code>`*
Ignore rules per path

RULE SELECTION (cont.)

`--target-version` *`<py3xx>`*
Minimum Python to assume

`--statistics`
Count violations per rule

`ruff` **`check --select I --fix`**
Sort imports (isort rules)

`ruff` **`check --select UP --fix`**
Modernise syntax (pyupgrade)

Popular rule prefixes

Prefix	From
`E / W`	pycodestyle
`F`	Pyflakes
`I`	isort
`B`	bugbear
`UP`	pyupgrade
`SIM`	simplify
`N`	pep8-naming
`C4`	comprehensions

SUPPRESSING

`# noqa:` *`<code>`*
Ignore a rule on this line

`# noqa`
Ignore all rules on this line

`# ruff: noqa:` *`<code>`*
Ignore a rule for the file

`ruff` **`check --add-noqa`**
Insert noqa for all hits

EXPLORE

`ruff` **`rule`** *`<code>`*
Explain one rule

`ruff` **`config`**
List all settings

`ruff` **`config`** *`<option>`*
Describe one setting

OUTPUT AND FILES

`--output-format` *`<fmt>`*
Choose report format

`--exclude` *`<pattern>`*
Replace excluded paths

`--extend-exclude` *`<pattern>`*
Exclude more paths

`--config` *`<file|key=value>`*
Use a file or override a setting

CONFIG

pyproject.toml basics

```
[tool.ruff]
line-length = 100
target-version = "py312"
extend-exclude =
["migrations"]

[tool.ruff.lint]
select = ["E4", "E7", "E9",
"F", "I", "B"]
ignore = ["B008"]
```

CONFIG (cont.)

Per-file ignores

```
[tool.ruff.lint.per-file-
ignores]
"tests/**" = ["S101"]
"__init__.py" = ["F401"]
```

Formatter options

```
[tool.ruff.format]
quote-style = "double"
indent-style = "space"
docstring-code-format = true
```

ruff.toml (no tool. prefix)

```
line-length = 100

[lint]
extend-select = ["I"]

[format]
quote-style = "single"
```

EDITORS AND CI

`ruff` **`server`**
Run the language server

pre-commit hook

```
repos:
- repo: https://github.com/
astral-sh/ruff-pre-commit
  rev: <tag>
  hooks:
  - id: ruff-check
    args: [--fix]
  - id: ruff-format
```

GitHub Actions step

```
- run: pip install ruff
- run: ruff check --output-
format=github
- run: ruff format --check
```