SwiftLint Swift style linter

.swiftlint.yml rules + paths → *select* → **Swift files** included minus excluded → *lint* → **Violations** warning or error → *report* → **Reporter** xcode, json, ...

CONCEPTS

SwiftLint reads .swiftlint.yml from the directory you run it in; rules run on every .swift file below it.

Analyzer rules need a full compiler log and run only under swiftlint analyze; ordinary lint rules never run there.

--fix rewrites files on disk and skips linting. Commit or stash first, then run swiftlint again to see what is left.

RUN

swiftlint
Lint all Swift files below here

swiftlint lint *<path>*
Lint given files or folders

swiftlint --fix
Auto-correct what can be fixed

swiftlint lint --strict
Fail on warnings too

swiftlint lint --quiet
Print violations only

swiftlint lint --lenient
Downgrade errors to warnings

RUN (cont.)

swiftlint lint --force-exclude *<file>*
Skip excluded files even if named

swiftlint lint --only-rule *<rule>*
Run just this rule

swiftlint --version
Print the version

RULES

swiftlint rules
List rules and their status

swiftlint rules *<rule>*
Show one rule's description

swiftlint rules --enabled
Only rules enabled by config

swiftlint rules --disabled
Only rules that are off

swiftlint rules --correctable
Only rules --fix can fix

swiftlint docs
Open the rule reference

CONFIG

disabled_rules: [*<rule>***]**
Turn off default rules

CONFIG (cont.)

opt_in_rules: [*<rule>***]**
Turn on extra rules

included: [Sources]
Lint only these paths

excluded: [Pods, .build]
Never lint these paths

only_rules: [*<rule>***]**
Run exactly these rules

opt_in_rules: [all]
Enable every opt-in rule

strict: true
Warnings fail the run

reporter: "*<name>***"**
Default output format

***<rule>*: warning**
Set a rule's severity

line_length: 120
Warning threshold for a rule

file_length: {warning: 500, error: 1200}
Warning and error limits

swiftlint --config *<a>* **--config** *<b>*
Stack configs, last wins

INLINE CONTROL

// swiftlint:disable *<rule>*
Off until enabled or end of file

INLINE CONTROL (cont.)

// swiftlint:enable *<rule>*
Turn the rule back on

// swiftlint:disable:next *<rule>*
Off for the next line

// swiftlint:disable:this *<rule>*
Off for this line

// swiftlint:disable:previous *<r>*
Off for the previous line

// swiftlint:disable all
Silence every rule

// swiftlint:disable *<r1>* *<r2>*
Several rules at once

OUTPUT

swiftlint reporters
List available reporters

swiftlint lint --reporter json
Choose the output format

swiftlint lint --reporter github-actions-logging
Annotate GitHub PRs

BASELINE & ANALYZE

swiftlint lint --write-baseline *<file>*
Save current violations

swiftlint lint --baseline *<file>*
Report only new violations

swiftlint baseline report *<file>*
Show what a baseline hides

swiftlint analyze --compiler-log-path *<log>*
Run analyzer rules

INSTALL & CI

brew install swiftlint
Install with Homebrew

mint install realm/SwiftLint
Install with Mint

RECIPES

Minimal .swiftlint.yml

```
disabled_rules:
  - trailing_whitespace
opt_in_rules:
  - empty_count
  - closure_spacing
included:
  - Sources
excluded:
  - Pods
line_length: 120
type_name:
  min_length: 4
```