# Stylelint CSS linter

**START HERE**

**`npx stylelint "**/*.css"`**
Lint all CSS files

**`npx stylelint "src/**/*.scss" --fix`**
Lint and auto-fix

**`npx stylelint --print-config <file>`**
Show the config in effect

**`npx stylelint -f json <file>`**
Machine-readable output

**`npx stylelint --cache "**/*.css"`**
Skip unchanged files

**`npx stylelint --rd "**/*.css"`**
Find useless disable comments

**`echo "a{}" | npx stylelint --stdin-filename a.css`**
Lint piped input

**`npx stylelint --mw 0 "**/*.css"`**
Fail on any warning

## CONCEPTS

Quote globs so Stylelint expands them itself; node_modules is always ignored. Exit code 2 means lint problems, 1 a fatal error.

**Config** extends + rules
↓ applies to
**Files** CSS, SCSS, HTML…
↓ reports
**Problems** error or warning
↓ --fix
**Fixed source** safe rules only

## RUNNING

**`stylelint "<glob>"`**
Lint files matching a glob

**`stylelint <dir>/`**
Lint a directory

**`stylelint "<glob>" --allow-empty-input`**
No error if glob matches nothing
*Without it a glob matching no files is a fatal error (exit 1).*

**`stylelint --stdin`**
Read source from stdin

**`stylelint --stdin-filename <name>`**
Name the stdin input
*Picks the syntax and overrides by file name.*

**`stylelint --custom-syntax <name>`**
Use a custom PostCSS syntax

## RUNNING (cont.)

**`stylelint --ignore-path <file>`**
Read ignore patterns from a file

**`stylelint --ignore-pattern "<glob>"`**
Ignore extra paths

**`stylelint --disable-default-ignores`**
Also lint node_modules

## FIXING

**`stylelint --fix "<glob>"`**
Fix what is auto-fixable
*Only rules that support it are fixed; the rest are still reported.*

**`stylelint --suppress "<glob>"`**
Record current problems in a file
*Writes stylelint-suppressions.json so only new problems fail.*

## SEVERITY AND EXIT CODES

**`stylelint -q "<glob>"`**
Show errors only, hide warnings

**`stylelint --max-warnings <n>`**
Exit 2 above n warnings
*Useful with defaultSeverity: "warning" in CI.*

## SEVERITY AND EXIT CODES (cont.)

**Exit codes**

| Code | Meaning |
|---|---|
| 0 | No problems |
| 1 | Fatal error |
| 2 | Lint problems |
| 64 | Bad CLI usage |
| 78 | Invalid config file |

## OUTPUT

**`stylelint -f <name>`**
Choose a formatter
*string (default), compact, json, tap, unix, verbose.*

**`stylelint -f verbose "<glob>"`**
List files checked, with tally

**`stylelint -f json "<glob>"`**
JSON report for tools

**`stylelint -o <path>`**
Write the report to a file

**`stylelint --custom-formatter <path>`**
Use your own formatter module

## CACHE

**`stylelint --cache "<glob>"`**
Only re-lint changed files
*Cache lives in .stylelintcache by default; gitignore it.*

**`stylelint --cache-location <path>`**
Move the cache file or dir
*A directory needs a trailing slash.*

## CACHE (cont.)

**`stylelint --cache-strategy content`**
Detect changes by content
*Use in CI, where fresh clones change mtimes.*

## CONFIG

**`stylelint --config <file>`**
Use a specific config

**`stylelint --print-config <file>`**
Print the resolved config
*Takes a file path, not a glob.*

**Config files searched (in order)**

| Where |
|---|
| `"stylelint" key in package.json` |
| `.stylelintrc` |
| `.stylelintrc.{cjs,mjs,js,ts,json,yaml,yml}` |
| `stylelint.config.{cjs,mjs,js,ts}` |

**.stylelintrc.json**

```
{
  "extends": ["stylelint-config-standard"],
  "rules": {
    "color-named": "never",
    "declaration-no-important": [true, { "severity": "warning" }],
    "selector-class-pattern": null
  },
  "ignoreFiles": ["dist/**"]
}
```

## CONFIG (cont.)

**Per-file overrides**

```
"overrides": [{
  "files": ["**/*.scss"],
  "customSyntax": "postcss-scss"
}]
```

**`"defaultSeverity": "warning"`**
Make every rule a warning

**`"rules": { "<rule>": null }`**
Turn a rule off

## DISABLE COMMENTS

**`/* stylelint-disable */ ... /* stylelint-enable */`**
Turn off a region

**`/* stylelint-disable <rule> */`**
Turn off one rule below

**`/* stylelint-disable-next-line <rule> */`**
Skip the next line

**`/* stylelint-disable-line <rule> */`**
Skip this line

**`/* stylelint-disable <rule> -- <reason> */`**
Add a description
*Required by --report-descriptionless-disables.*

## DISABLE COMMENTS (cont.)

**`stylelint --report-needless-disables`**
Flag useless disable comments
*Exits 2 if any are found.*

**`stylelint --report-descriptionless-disables`**
Flag disables without a reason

**`stylelint --ignore-disables`**
Ignore all disable comments

## RECIPES

**CI check**

```
npx stylelint --cache --cache-strategy content \
  --max-warnings 0 "src/**/*.{css,scss}"
```

**Adopt in a legacy codebase**

```
npx stylelint --suppress "**/*.css"
git add stylelint-suppressions.json
# new problems fail; old ones are tolerated
```

**Quick rule audit**

```
npx stylelint --print-config src/app.css | less
npx stylelint -f json "**/*.css" > report.json
```

**Install with the standard config**

```
npm i -D stylelint stylelint-config-standard
echo '{"extends":["stylelint-config-standard"]}' \
  > .stylelintrc.json
```