ESLint JavaScript linter

CONCEPTS

> ESLint 9+ reads only the flat config eslint.config.js (or .mjs/.cjs/.ts).
> Old .eslintrc files are ignored.

Files globs from CLI
↓ match
Flat config eslint.config.js
↓ apply rules
Problems error / warn
↓ format
Report -f, exit code

RUNNING

`npx eslint .`
Lint the current directory

`npx eslint <file> <dir>`
Lint specific files or folders

`npx eslint "src/**/*.{js,ts}"`
Lint a quoted glob

FIXING

`npx eslint --fix .`
Apply automatic fixes

WARNINGS AND EXIT CODES

`npx eslint --quiet .`
Hide warnings, show errors

`npx eslint --max-warnings 0 .`
Exit 1 if any warning appears

`echo $?`
0 clean, 1 lint failure, 2 config or crash

OUTPUT

`npx eslint -f json .`
Machine-readable JSON

CACHE AND SPEED

`npx eslint --cache .`
Only lint changed files

CONFIG

`npm init @eslint/config@latest`
Create eslint.config.js interactively

`npx eslint -c <file> .`
Use a specific config file

`npx eslint --print-config <file>`
Show the config a file gets

`npx eslint --debug <file>`
Print debug output

eslint.config.js

```
import js from "@eslint/js";
import globals from "globals";

export default [
  { ignores: ["dist/", "coverage/"] },
  js.configs.recommended,
  {
    files: ["**/*.js"],
    languageOptions: { globals:
globals.node },
    rules: { "no-unused-vars": "warn",
eqeqeq: "error" },
  },
];
```

defineConfig and extends

```
import { defineConfig } from "eslint/
config";
import js from "@eslint/js";

export default defineConfig([
  { files: ["src/**/*.js"], plugins:
{ js }, extends: ["js/recommended"] },
  { files: ["test/**"], rules: { "no-
console": "off" } },
]);
```

`{ ignores: ["dist/"] }`
Global ignore (only key in the object)

`{ files: ["**/*.ts"], ... }`
Limit a config object to matching files

INLINE COMMENTS

`// eslint-disable-next-line <rule>`
Silence a rule on the next line

`// eslint-disable-line <rule>`
Silence a rule on this line