

# Docent

## *A Documentation Linter for Zig Projects*

v0.0.0, 2026-06-22: main (557b706)

# Table of Contents

Synopsis .....	2
Global options .....	3
-h, --help.....	3
-V, --version.....	3
Commands .....	4
Command reference .....	5
docent status.....	5
docent init.....	6
docent check.....	7
docent check doc.....	8
docent check style.....	10
docent check complexity.....	11
docent check all.....	13
docent docs .....	15
docent completion.....	15
docent help .....	16

*Documentation linter for Zig projects.*

# Synopsis

```
docent [OPTIONS] <COMMAND>
```

# Global options

**-h, --help**

Print help.

**Scope:** `built-in`

**-V, --version**

Print version.

**Scope:** `built-in`

# Commands

- `docent status` — Show project lint plan and effective rules
- `docent init` — Create a default Docent configuration file
- `docent check` — Run Docent lint checks
- `docent docs` — Generate AsciiDoc documentation for this CLI
- `docent completion` — Generate shell completion scripts
- `docent help` — Print this message or the help of the given subcommand(s)
- `docent help` — Print this message or the help of the given subcommand(s)

# Command reference

## docent status

Show project lint plan and effective rules

Print project metadata, lint scan roots, excluded dependencies, resolved targets, and effective rule severities. Always exits 0 after a successful report (use `docent` to lint and enforce severities).

Parent: `docent` ===== Synopsis

```
docent status [OPTIONS] <paths>...
```

### Arguments

`<paths>...`

Files or directories to summarize. If omitted, uses package paths from `build.zig.zon` when available.

### Options

`--config-path <PATH>`

Path to `docent.toml`

When omitted, Docent searches upward from the working directory for `.config/docent.toml`.

**Scope:** `local`

`--lib`

Lint library targets only (default)

**Default:** `false` **Scope:** `local`

`--bins`

Lint all binary targets

**Default:** `false` **Scope:** `local`

`--bin <STRING>...`

Lint specific binary by name (repeatable)

**Scope:** `local`

`--tests`

Lint all test targets

**Default:** `false` **Scope:** `local`

**--test <STRING>...**

Lint specific test by name (repeatable)

**Scope:** local

**--deps**

Also lint files under path dependencies from build.zig.zon

**Default:** false **Scope:** local

**--include-deps**

List build targets and module roots discovered in path dependencies

**Default:** false **Scope:** local

**--build-script**

Include the build script module and everything it depends on to be analyzed

**Default:** false **Scope:** local

**-h, --help**

Print help.

**Scope:** built-in

## docent init

Create a default Docent configuration file

Write `.config/docent.toml` using the bundled template and the published JSON Schema URL. Does not overwrite an existing file.

Parent: `docent` ===== Synopsis

```
docent init [OPTIONS]
```

### Options

**--force**

Overwrite an existing configuration file

**Default:** false **Scope:** local

**-h, --help**

Print help.

**Scope:** built-in

## docent check

Run Docent lint checks

Run documentation, style, or complexity checks. Use a category subcommand for full diagnostics, or run `docent check` alone for a compact summary across every category.

Parent: `docent` ===== Synopsis

```
docent check [OPTIONS] <COMMAND>
```

### Options

`--config-path <PATH>`

Path to `docent.toml`

When omitted, Docent searches upward from the working directory for `.config/docent.toml`.

**Scope:** local

`--lib`

Analyze library targets only (default)

**Default:** false **Scope:** local

`--bins`

Analyze all binary targets

**Default:** false **Scope:** local

`--bin <STRING>...`

Analyze specific binary by name (repeatable)

**Scope:** local

`--tests`

Analyze all test targets

**Default:** false **Scope:** local

`--test <STRING>...`

Analyze specific test by name (repeatable)

**Scope:** local

**--deps**

Also analyze files under path dependencies from build.zig.zon

**Default:** `false` **Scope:** `local`

**--build-script**

Include the build script module and everything it depends on to be analyzed

**Default:** `false` **Scope:** `local`

**-h, --help**

Print help.

**Scope:** `built-in`

## Subcommands

- `docent check doc` — Check documentation comments
- `docent check style` — Check naming-case and style rules
- `docent check complexity` — Check function complexity and parameter counts
- `docent check all` — Run every check category

## docent check doc

Check documentation comments

Lint doc comments on the public API surface (or all declarations when `scan_mode` is "all" in config). Exits non-zero when a denied rule reports a finding.

Parent: `docent check` ===== Synopsis

```
docent check doc [OPTIONS] <paths>...
```

## Arguments

`<paths>...`

Files or directories to analyze. If omitted, uses package paths from build.zig.zon when available.

## Options

**--config-path** `<PATH>`

Path to docent.toml

When omitted, Docent searches upward from the working directory for `.config/docent.toml`.

**Scope:** `global`

**--lib**

Analyze library targets only (default)

**Default:** `false` **Scope:** `global`

**--bins**

Analyze all binary targets

**Default:** `false` **Scope:** `global`

**--bin <STRING>...**

Analyze specific binary by name (repeatable)

**Scope:** `global`

**--tests**

Analyze all test targets

**Default:** `false` **Scope:** `global`

**--test <STRING>...**

Analyze specific test by name (repeatable)

**Scope:** `global`

**--deps**

Also analyze files under path dependencies from build.zig.zon

**Default:** `false` **Scope:** `global`

**--build-script**

Include the build script module and everything it depends on to be analyzed

**Default:** `false` **Scope:** `global`

**-f, --format <FORMAT>**

Output format

**Default:** `pretty` **Possible values:** `pretty, minimal, json` **Scope:** `local`

**-F, --fail-fast <WHEN>**

Stop after the first matching severity

**Default:** `none` **Possible values:** `none, error, warn, any` **Scope:** `local`

**-h, --help**

Print help.

**Scope:** `built-in`

## docent check style

Check naming-case and style rules

Check identifiers in the import-closure reachable from the project's module roots. Severities are set in project config (`.config/docent.toml`). Exits non-zero when a denied rule reports a finding.

Parent: `docent check` ===== Synopsis

```
docent check style [OPTIONS] <paths>...
```

### Arguments

`<paths>...`

Files or directories to analyze. If omitted, uses package paths from `build.zig.zon` when available.

### Options

**--config-path** `<PATH>`

Path to `docent.toml`

When omitted, Docent searches upward from the working directory for `.config/docent.toml`.

**Scope:** `global`

**--lib**

Analyze library targets only (default)

**Default:** `false` **Scope:** `global`

**--bins**

Analyze all binary targets

**Default:** `false` **Scope:** `global`

**--bin** `<STRING>...`

Analyze specific binary by name (repeatable)

**Scope:** `global`

**--tests**

Analyze all test targets

**Default:** `false` **Scope:** `global`

**--test <STRING>...**

Analyze specific test by name (repeatable)

**Scope:** `global`

**--deps**

Also analyze files under path dependencies from `build.zig.zon`

**Default:** `false` **Scope:** `global`

**--build-script**

Include the build script module and everything it depends on to be analyzed

**Default:** `false` **Scope:** `global`

**-f, --format <FORMAT>**

Output format

**Default:** `pretty` **Possible values:** `pretty, minimal, json` **Scope:** `local`

**-F, --fail-fast <WHEN>**

Stop after the first matching severity

**Default:** `none` **Possible values:** `none, error, warn, any` **Scope:** `local`

**-h, --help**

Print help.

**Scope:** `built-in`

## docent check complexity

Check function complexity and parameter counts

Measure complexity and parameter counts for every function reachable from the project's module roots. Thresholds are set in project config (`.config/docent.toml`). Exits non-zero when a denied rule reports a finding.

Parent: `docent check` ===== Synopsis

```
docent check complexity [OPTIONS] <paths>...
```

## Arguments

`<paths>...`

Files or directories to analyze. If omitted, uses package paths from `build.zig.zon` when available.

## Options

`--config-path <PATH>`

Path to `docent.toml`

When omitted, Docent searches upward from the working directory for `.config/docent.toml`.

**Scope:** `global`

`--lib`

Analyze library targets only (default)

**Default:** `false` **Scope:** `global`

`--bins`

Analyze all binary targets

**Default:** `false` **Scope:** `global`

`--bin <STRING>...`

Analyze specific binary by name (repeatable)

**Scope:** `global`

`--tests`

Analyze all test targets

**Default:** `false` **Scope:** `global`

`--test <STRING>...`

Analyze specific test by name (repeatable)

**Scope:** `global`

`--deps`

Also analyze files under path dependencies from `build.zig.zon`

**Default:** `false` **Scope:** `global`

`--build-script`

Include the build script module and everything it depends on to be analyzed

**Default:** `false` **Scope:** `global`

`-f, --format <FORMAT>`

Output format

**Default:** `pretty` **Possible values:** `pretty, minimal, json` **Scope:** `local`

`-F, --fail-fast <WHEN>`

Stop after the first matching severity

**Default:** `none` **Possible values:** `none, error, warn, any` **Scope:** `local`

`-h, --help`

Print help.

**Scope:** `built-in`

## docent check all

Run every check category

Run documentation, style, and complexity checks in one pass and print all diagnostics. Exits non-zero when a denied rule reports a finding.

Parent: `docent check` ===== Synopsis

```
docent check all [OPTIONS] <paths>...
```

### Arguments

`<paths>...`

Files or directories to analyze. If omitted, uses package paths from `build.zig.zon` when available.

### Options

`--config-path <PATH>`

Path to `docent.toml`

When omitted, Docent searches upward from the working directory for `.config/docent.toml`.

**Scope:** `global`

`--lib`

Analyze library targets only (default)

**Default:** `false` **Scope:** `global`

**--bins**

Analyze all binary targets

**Default:** `false` **Scope:** `global`

**--bin <STRING>...**

Analyze specific binary by name (repeatable)

**Scope:** `global`

**--tests**

Analyze all test targets

**Default:** `false` **Scope:** `global`

**--test <STRING>...**

Analyze specific test by name (repeatable)

**Scope:** `global`

**--deps**

Also analyze files under path dependencies from build.zig.zon

**Default:** `false` **Scope:** `global`

**--build-script**

Include the build script module and everything it depends on to be analyzed

**Default:** `false` **Scope:** `global`

**-f, --format <FORMAT>**

Output format

**Default:** `pretty` **Possible values:** `pretty, minimal, json` **Scope:** `local`

**-F, --fail-fast <WHEN>**

Stop after the first matching severity

**Default:** `none` **Possible values:** `none, error, warn, any` **Scope:** `local`

**-h, --help**

Print help.

**Scope:** `built-in`

## docent docs

Generate AsciiDoc documentation for this CLI

Parent: `docent` ===== Synopsis

```
docent docs [OPTIONS]
```

### Options

`-o, --output-dir <STRING>`

Output directory where the AsciiDoc documentation is written.

**Default:** `docs` **Scope:** `local`

`-f, --file <STRING>`

Output file name.

**Default:** `docent.adoc` **Scope:** `local`

`--template <STRING>`

Optional path to a custom Trama template file (AsciiDoc).

**Scope:** `local`

`-h, --help`

Print help.

**Scope:** `built-in`

## docent completion

Generate shell completion scripts

Parent: `docent` ===== Synopsis

```
docent completion [OPTIONS] <shell>
```

### Arguments

`<shell>`

Target shell.

**Possible values:** `bash, zsh, fish, pwsh, nu`

## Options

`-h, --help`

Print help.

**Scope:** `built-in`

## docent help

Print this message or the help of the given subcommand(s)

Parent: `docent` ==== Synopsis

```
docent help [COMMAND]...
```