dataset check

In this article

[Plumbing] Check a particular dataset for internal consistency.

This content describes the most recent release of the CodeQL CLI. For more information about this release, see https://github.com/github/codeql-cli-binaries/releases.

To see details of the options available for this command in an earlier release, run the command with the `--help` option in your terminal.

Synopsis ⊘

```shell
codeql dataset check <options>... -- <dataset>
```

Description ⊘

[Plumbing] Check a particular dataset for internal consistency.

This command is most commonly useful to developers of CodeQL extractors, as it validates the data produced by the extractor. It may also be useful if queries against a database are giving inconsistent results, to rule out issues in the underlying data as the cause.

Options ⊘

Primary Options ⊘

`<dataset>` ⊘

[Mandatory] Path to the raw QL dataset to check.

`--failing-exitcode=<code>` 🔗

[Advanced] Set the exit code to produce if any failures are encountered. Usually 1, but tooling that parses the output may find it useful to set it to 0.

`--format=<fmt>` 🔗

Select output format. Possible choices:

`text` *(default)*: A human-readable textual rendering.

`json` : A streamed JSON array of objects.

`jsonz` : A stream of zero-terminated JSON objects.

`--[no-]precise-locations` 🔗

[Advanced] Expend extra effort to compute precise locations for inconsistencies. This will take more time, but may make it easier to debug extractor behaviour.

`--max-resolve-depth=<n>` 🔗

[Advanced] The maximum depth to which IDs should be resolved to explain inconsistencies. (Default: 3)

`--max-errors-per-checker=<n>` 🔗

The maximum number of inconsistency errors of each kind that should be reported explicitly. (Default: 5)

Common options 🔗

`-h, --help` 🔗

Show this help text.

`-J=<opt>` 🔗

[Advanced] Give option to the JVM running the command.

(Beware that options containing spaces will not be handled correctly.)

`-v, --verbose` 🔗

Incrementally increase the number of progress messages printed.

`-q, --quiet` 🔗

Incrementally decrease the number of progress messages printed.

`--verbosity=<level>` 🔗

[Advanced] Explicitly set the verbosity level to one of errors, warnings, progress, progress+, progress++, progress+++. Overrides `-v` and `-q` .

`--logdir=<dir>` 🔗

[Advanced] Write detailed logs to one or more files in the given directory, with generated names that include timestamps and the name of the running subcommand.

(To write a log file with a name you have full control over, instead give `--log-to-stderr` and redirect stderr as desired.)

Legal

Terms   Privacy   Status   Pricing   Expert services   Blog