

database cleanup

In this article

- Synopsis
- Description
- Options

Compact a CodeQL database on disk.


GitHub CodeQL is licensed on a per-user basis upon installation. You can use CodeQL only for certain tasks under the license restrictions. For more information, see "[About the CodeQL CLI](#)." If you have a GitHub Advanced Security license, you can use CodeQL for automated analysis, continuous integration, and continuous delivery. For more information, see "[About GitHub Advanced Security](#)."

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 database cleanup [--mode=<mode>] <options>... -- <database>
```


Description

Compact a CodeQL database on disk.

Delete temporary data, and generally make a database as small as possible on disk without degrading its future usefulness.

Options

Primary Options

`<database>` 

[Mandatory] Path to the CodeQL database to clean up.

Low-level dataset cleanup options

--max-disk-cache=<MB> [🔗](#)

Set the maximum amount of space that the disk cache for intermediate query results can use.

If this size is not configured explicitly, the evaluator will try to use a "reasonable" amount of cache space, based on the size of the dataset and the complexity of the queries. Explicitly setting a higher limit than this default usage will enable additional caching which can speed up later queries.

--min-disk-free=<MB> [🔗](#)

[Advanced] Set target amount of free space on file system.

If `--max-disk-cache` is not given, the evaluator will try hard to curtail disk cache usage if the free space on the file system drops below this value.

--min-disk-free-pct=<pct> [🔗](#)

[Advanced] Set target fraction of free space on file system.

If `--max-disk-cache` is not given, the evaluator will try hard to curtail disk cache usage if the free space on the file system drops below this percentage.

-m, --mode=<mode> [🔗](#)

Select how aggressively to trim the cache. Choices include:

`clear` : Remove the entire cache, trimming down to the state of a freshly extracted dataset

`trim` (*default*): Trim everything except explicitly "cached" predicates.

`fit` : Simply make sure the defined size limits for the disk cache are observed, deleting as many intermediates as necessary.

--cleanup-upgrade-backups [🔗](#)

Delete any backup directories resulting from database upgrades.

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

© 2023 GitHub, Inc. [Terms](#) [Privacy](#) [Status](#) [Pricing](#) [Expert services](#) [Blog](#)