

database index-files

In this article

- Synopsis
- Description
- Options

[Plumbing] Index standalone files with a given CodeQL extractor.

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 index-files --language=<lang> [--threads=<num>] [--ram=<MB>] [--
extractor-option=<extractor-option-name=value>] <options>... -- <database>
```

Description [↗](#)

[Plumbing] Index standalone files with a given CodeQL extractor.


This command selects a set of files under the specified working directory, and applies the given extractor to them. By default, all files are selected. Typical invocations will specify options to restrict the set of included files.

The `--include`, `--exclude`, and `--prune` options all take glob patterns, which can use the following wildcard characters:


- A single "?" matches any character other than a forward/backward slash;
- A single "*" matches any number of characters other than a forward/backward slash;
- The pattern "**" matches zero or more complete directory components.

Options [↗](#)

Primary Options

<database> 

[Mandatory] Path to the CodeQL database under construction. This must have been prepared for extraction with [codeql database init](#).


-l, --language=<lang> 

[Mandatory] The extractor that should be used to index matching files.

-j, --threads=<num> 

Ask the extractor to use this many threads. This option is passed to the extractor as a suggestion. If the CODEQL_THREADS environment variable is set, the environment variable value takes precedence over this option.

You can pass 0 to use one thread per core on the machine, or *-N* to leave *N* cores unused (except still use at least one thread).

-M, --ram=<MB> 

Ask the extractor to use this much memory. This option is passed to the extractor as a suggestion. If the CODEQL_RAM environment variable is set, the environment variable value takes precedence over this option.

--working-dir=<dir> 

[Advanced] The directory in which the specified command should be executed. If this argument is not provided, the command is executed in the value of `--source-root` passed to [codeql database create](#), if one exists. If no `--source-root` argument is provided, the command is executed in the current working directory.

Options to control extractor behavior

-O, --extractor-option=<extractor-option-name=value> 

Set options for CodeQL extractors. `extractor-option-name` should be of the form `extractor_name.group1.group2.option_name` or `group1.group2.option_name`. If `extractor_option_name` starts with an extractor name, the indicated extractor must declare the option `group1.group2.option_name`. Otherwise, any extractor that declares the option `group1.group2.option_name` will have the option set. `value` can be any string that does not contain a newline.

You can use this command-line option repeatedly to set multiple extractor options. If you provide multiple values for the same extractor option, the behaviour depends on the type that the extractor option expects. String options will use the last value provided. Array options will use all the values provided, in order. Extractor options specified using this command-line option are processed after extractor options given via `--extractor-options-file`.

When passed to [codeql database init](#) or `codeql database begin-tracing`, the options will only be applied to the indirect tracing environment. If your workflow also makes calls to [codeql database trace-command](#) then the options also need to be passed there if desired.

See <https://codeql.github.com/docs/codeql-cli/extractor-options> for more information on CodeQL extractor options, including how to list the options declared by each extractor.

--extractor-options-file=<extractor-options-bundle-file> 

Specify extractor option bundle files. An extractor option bundle file is a JSON file (extension `.json`) or YAML file (extension `.yaml` or `.yml`) that sets extractor options. The file must have the top-level map key 'extractor' and, under it, extractor names as second-level map keys. Further levels of maps represent nested extractor groups, and string and array options are map entries with string and array values.

Extractor option bundle files are read in the order they are specified. If different extractor option bundle files specify the same extractor option, the behaviour depends on the type that the extractor option expects. String options will use the last value provided. Array options will use all the values provided, in order. Extractor options specified using this command-line option are processed before extractor options given via `--extractor-option`.

When passed to [codeql database init](#) or `codeql database begin-tracing`, the options will only be applied to the indirect tracing environment. If your workflow also makes calls to [codeql database trace-command](#) then the options also need to be passed there if desired.

See <https://codeql.github.com/docs/codeql-cli/extractor-options> for more information on CodeQL extractor options, including how to list the options declared by each extractor.

Options for limiting the set of indexed files

--include-extension=<.ext> 

Include all files in the search directory tree that have the given extension. Typically, you should include the dot before the extension. For example, passing `--include-extension .xml` will include all files with the ".xml" extension. This option is incompatible with negated `--include` options.


--include=<glob> 

Include all files and directories in the search directory tree that match the given glob, using each file and directory's relative path from the search directory. If the glob begins with a `!` character, the matching files and directories would instead be excluded.


`--include` options are processed in order, with later options overriding earlier ones. For example, `--include ** --include !sub/*.ts --include sub/main.*` would include `sub/main.ts` (because it is included by `sub/main.*`), exclude `sub/index.ts` (because it is excluded by `!sub/*.ts`), and include `sub/test.js` (because it is included by `**` without being subsequently excluded.)

--also-match=<glob> 

Require all results to also match the given glob, using each file and directory's relative path from the search directory. This option has the same structure and the same interpretation as `--include` but specifies a separate sequence of globs that are applied in conjunction with `--include`.

--exclude=<glob> 

Exclude all files and directories that match the given glob, using each file and directory's relative path from the search directory. This option overrides all include options. This option is incompatible with negated `--include` options.

--prune=<glob> 

Exclude all files and directories that match the given glob, using each file and directory's relative path from the search directory. This option overrides all include options. This option is incompatible with negated `--include` options.

`--size-limit=<bytes>` [↗](#)

Exclude all files whose size exceeds the given limit. The size limit is in bytes, or in kibibytes (KiB) with the "k" suffix, in mebibytes (MiB) with the "m" suffix, and in gibibytes (GiB) with the "g" suffix. This option overrides all include options.

`--total-size-limit=<bytes>` [↗](#)

Make the command exit with an error if the combined size of all resolved files would exceed the given limit. The size limit is in bytes, or in kibibytes (KiB) with the "k" suffix, in mebibytes (MiB) with the "m" suffix, and in gibibytes (GiB) with the "g" suffix.

`--[no-]follow-symlinks` [↗](#)

Follow any symbolic links to their targets.

`--[no-]find-any` [↗](#)

Find at most one match (as opposed to all matches).

Available since `v2.11.3`.

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