tokei

Count lines of code

CONCEPTS

Each language row splits lines into code, comments and blanks. Embedded code (CSS in HTML, say) is counted under its own language as a child row.

tokei honours .gitignore, .ignore and .tokeignore by default, so build output and vendored dirs listed there are not counted.

COUNTING

tokei
Table for the current directory

tokei *<dir> <dir>*
Count several paths together

tokei -f
Show every file

tokei -s *<key>*
Sort rows by files, lines, blanks, code or comments

tokei -s code -f
Biggest files by code lines

tokei --hidden
Include hidden files

tokei -l
List supported languages and extensions

tokei -V
Print the version

FILTERING

tokei -t=*<lang>*,*<lang>*
Only these languages

FILTERING (cont.)

tokei -t=Rust,Markdown
Example: Rust and Markdown only

tokei -e *<pattern>*
Exclude files or dirs, gitignore syntax

tokei -e '*.d' -e *<dir>*
Repeat -e for several patterns

tokei --no-ignore
Count everything, ignore all ignore files

tokei --no-ignore-vcs
Skip only .gitignore/.hgignore

Quote glob patterns so the shell does not expand them first.

OUTPUT AND CONFIG

tokei -o json
Print JSON instead of a table

tokei -o yaml
Print YAML

tokei -o json > stats.json
Save a run for later

tokei -i stats.json
Load a saved run instead of scanning

Some builds lack json, yaml or cbor. Reinstall with cargo install tokei --features all.

.tokeignore
Per-project exclude list, gitignore syntax

tokei.toml / .tokeirc
Default options; read from cwd, home, config dir