



Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'bundle-gem.1' command

\$ man bundle-gem.1

BUNDLE-GEM(1) BUNDLE-GEM(1)

NAME

bundle-gem - Generate a project skeleton for creating a rubygem

SYNOPSIS

bundle gem GEM_NAME OPTIONS

DESCRIPTION

Generates a directory named GEM_NAME with a Rakefile, GEM_NAME.gemspec, and other supporting files and directories that can be used to develop a rubygem with that name.

Run `rake -T` in the resulting project for a list of Rake tasks that can be used to test and publish the gem to rubygems.org.

The generated project skeleton can be customized with OPTIONS, as explained below. Note that these options can also be specified via Bundler's global configuration file using the following names:

? gem.coc

? gem.mit

? gem.test

OPTIONS

--exe or -b or --bin

Specify that Bundler should create a binary executable (as `exe/GEM_NAME`) in the generated rubygem project. This binary will also be added to the `GEM_NAME.gemspec` manifest. This behavior is disabled by default.

--no-exe

Do not create a binary (overrides --exe specified in the global config).

--coc Add a CODE_OF_CONDUCT.md file to the root of the generated project. If this option is unspecified, an interactive prompt will be displayed and the answer will be saved in Bundler's global config for future bundle gem use.

--no-coc

Do not create a CODE_OF_CONDUCT.md (overrides --coc specified in the global config).

--ext Add boilerplate for C extension code to the generated project.

This behavior is disabled by default.

--no-ext

Do not add C extension code (overrides --ext specified in the global config).

--mit Add an MIT license to a LICENSE.txt file in the root of the generated project. Your name from the global git config is used for the copyright statement. If this option is unspecified, an interactive prompt will be displayed and the answer will be saved in Bundler's global config for future bundle gem use.

--no-mit

Do not create a LICENSE.txt (overrides --mit specified in the global config).

-t, --test=minitest, --test=rspec, --test=test-unit

Specify the test framework that Bundler should use when generating the project. Acceptable values are minitest, rspec and test-unit. The GEM_NAME.gemspec will be configured and a skeleton test/spec directory will be created based on this option.

Given no option is specified:

When Bundler is configured to generate tests, this defaults to Bundler's global config setting gem.test.

When Bundler is configured to not generate tests, an interactive prompt will be displayed and the answer will be used for the

current rubygem project.

When Bundler is unconfigured, an interactive prompt will be displayed and the answer will be saved in Bundler's global config for future bundle gem use.

`--ci, --ci=github, --ci=travis, --ci=gitlab, --ci=circle`

Specify the continuous integration service that Bundler should use when generating the project. Acceptable values are github, travis, gitlab and circle. A configuration file will be generated in the project directory. Given no option is specified:

When Bundler is configured to generate CI files, this defaults to Bundler's global config setting `gem.ci`.

When Bundler is configured to not generate CI files, an interactive prompt will be displayed and the answer will be used for the current rubygem project.

When Bundler is unconfigured, an interactive prompt will be displayed and the answer will be saved in Bundler's global config for future bundle gem use.

`--linter, --linter=rubocop, --linter=standard`

Specify the linter and code formatter that Bundler should add to the project's development dependencies. Acceptable values are rubocop and standard. A configuration file will be generated in the project directory. Given no option is specified:

When Bundler is configured to add a linter, this defaults to Bundler's global config setting `gem.linter`.

When Bundler is configured not to add a linter, an interactive prompt will be displayed and the answer will be used for the current rubygem project.

When Bundler is unconfigured, an interactive prompt will be displayed and the answer will be saved in Bundler's global config for future bundle gem use.

`-e, --edit[=EDITOR]`

Open the resulting `GEM_NAME.gemspec` in EDITOR, or the default editor if not specified. The default is `$BUNDLER_EDITOR, $VIS`

UAL, or \$EDITOR.

SEE ALSO

? [bundle config\(1\) bundle-config.1.html](#)

December 2021

BUNDLE-GEM(1)