# Ignoring files

You can configure Git to ignore files you don't want to check in to GitHub.

## [In this article](https://docs.github.com/en/github/getting-started-with-github/getting-started-with-git/ignoring-files" \l "in-this-article)

* [Configuring ignored files for a single repository](https://docs.github.com/en/github/getting-started-with-github/getting-started-with-git/ignoring-files" \l "configuring-ignored-files-for-a-single-repository)
* [Configuring ignored files for all repositories on your computer](https://docs.github.com/en/github/getting-started-with-github/getting-started-with-git/ignoring-files" \l "configuring-ignored-files-for-all-repositories-on-your-computer)
* [Excluding local files without creating a](https://docs.github.com/en/github/getting-started-with-github/getting-started-with-git/ignoring-files" \l "excluding-local-files-without-creating-a-gitignore-file) *.gitignore* file
* [Further Reading](https://docs.github.com/en/github/getting-started-with-github/getting-started-with-git/ignoring-files" \l "further-reading)

### [Configuring ignored files for a single repository](https://docs.github.com/en/github/getting-started-with-github/getting-started-with-git/ignoring-files" \l "configuring-ignored-files-for-a-single-repository)

You can create a .gitignore file in your repository's root directory to tell Git which files and directories to ignore when you make a commit. To share the ignore rules with other users who clone the repository, commit the .gitignore file in to your repository.

GitHub maintains an official list of recommended .gitignore files for many popular operating systems, environments, and languages in the github/gitignore public repository. You can also use gitignore.io to create a .gitignore file for your operating system, programming language, or IDE. For more information, see "[github/gitignore](https://github.com/github/gitignore)" and the "[gitignore.io](https://www.gitignore.io/)" site.

1. Open Terminal.
2. Navigate to the location of your Git repository.
3. Create a .gitignore file for your repository.

$ touch .gitignore

For an example .gitignore file, see "[Some common .gitignore configurations](https://gist.github.com/octocat/9257657)" in the Octocat repository.

If you want to ignore a file that is already checked in, you must untrack the file before you add a rule to ignore it. From your terminal, untrack the file.

$ git rm --cached FILENAME

### [Configuring ignored files for all repositories on your computer](https://docs.github.com/en/github/getting-started-with-github/getting-started-with-git/ignoring-files" \l "configuring-ignored-files-for-all-repositories-on-your-computer)

You can also create a global .gitignore file to define a list of rules for ignoring files in every Git repository on your computer. For example, you might create the file at ~/.gitignore\_global and add some rules to it.

1. Open Terminal.
2. Configure Git to use the exclude file ~/.gitignore\_global for all Git repositories.

$ git config --global core.excludesfile ~/.gitignore\_global

### [Excluding local files without creating a](https://docs.github.com/en/github/getting-started-with-github/getting-started-with-git/ignoring-files" \l "excluding-local-files-without-creating-a-gitignore-file) *.gitignore* file

If you don't want to create a .gitignore file to share with others, you can create rules that are not committed with the repository. You can use this technique for locally-generated files that you don't expect other users to generate, such as files created by your editor.

Use your favorite text editor to open the file called .git/info/exclude within the root of your Git repository. Any rule you add here will not be checked in, and will only ignore files for your local repository.

1. Open Terminal.
2. Navigate to the location of your Git repository.
3. Using your favorite text editor, open the file .git/info/exclude.

### [Further Reading](https://docs.github.com/en/github/getting-started-with-github/getting-started-with-git/ignoring-files" \l "further-reading)

* [Ignoring files](https://git-scm.com/book/en/v2/Git-Basics-Recording-Changes-to-the-Repository" \l "_ignoring) in the Pro Git book
* [.gitignore](https://git-scm.com/docs/gitignore) in the man pages for Git
* [A collection of useful *.gitignore* templates](https://github.com/github/gitignore) in the github/gitignore repository
* [gitignore.io](https://www.gitignore.io/) site

https://github.com/github/gitignore

https://www.toptal.com/developers/gitignore