# GitHub Help

# About READMEs

You can add a README file to your repository to tell other people why your project is useful, what they can do with your project, and how they can use it.

**In this article**

Section links in README files and blob pages

Relative links and image paths in README files

Further reading

---

A README file, along with a repository license, contribution guidelines, and a code of conduct, helps you communicate expectations for and manage contributions to your project.

A README is often the first item a visitor will see when visiting your repository. README files typically include information on:

- What the project does
- Why the project is useful
- How users can get started with the project
- Where users can get help with your project
- Who maintains and contributes to the project

If you put your README file in your repository's root, `docs`, or hidden `.github` directory, GitHub will recognize and automatically surface your README to repository visitors.

## Section links in README files and blob pages

Many projects use a table of contents at the start of a README to direct users to different sections of the file. You can link directly to a section in a rendered file by hovering over the section heading to expose the link:

📖 README.md

# Scientist!

A Ruby library for carefully refactoring critical paths. `build passing`

## 🔗 How do I science?

Let's pretend you're changing the way you handle permissions in a large web app. Tests can help guide your refactoring, but you really want to compare the current and refactored behaviors under load.

```
require "scientist"
```

# Relative links and image paths in README files

You can define relative links and image paths in your rendered files to help readers navigate to other files in your repository.

A relative link is a link that is relative to the current file. For example, if you have a README file in root of your repository, and you have another file in *docs/CONTRIBUTING.md*, the relative link to *CONTRIBUTING.md* in your README might look like this:

```
[Contribution guidelines for this project](docs/CONTRIBUTING.md)
```

GitHub will automatically transform your relative link or image path based on whatever branch you're currently on, so that the link or path always works. You can use all relative link operands, such as `./` and `../`.

Relative links are easier for users who clone your repository. Absolute links may not work in clones of your repository - we recommend using relative links to refer to other files within your repository.
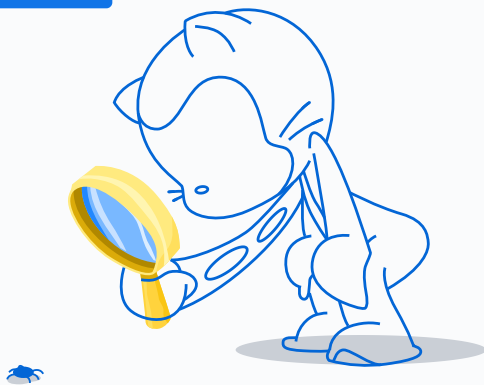
## Further reading

- "Adding a file to a repository"
- 18F's "Making READMEs readable"

Ask a human

## Can't find what you're looking for?

Contact us

# GitHub

| Product | Platform | Support | Company |
|---------|----------|---------|---------|
| Features | Developer API | Help | About |
| Security | Partners | Community Forum | Blog |
| Enterprise | Atom | Training | Careers |
| Case Studies | Electron | Status | Press |
| Pricing | GitHub Desktop | Contact GitHub | Shop |
| Resources | | | |