

# Free/Libre and Open Source Software

# How to get started?

Qian Qian (@Cubik65536)

Wednesday, October 16<sup>th</sup>, 2024

Montreal, Quebec, Canada

Content licensed under Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.



## Why contributing to F.L.O.S.S?

- Help improving the software - that **you're using**.
- Help others.
- Gain experience:
  - Coding
  - Writing
  - Team Working
  - Designing
- Get Paid

“How do I get started in contributing?”

based on my *own* experience

**“What can I do?”**

## **What kind of contributions can you make?**

- Code

“Code is only one part of open source.”

By Pratham Patel, posted on *IT'S FOSS*.  
<https://itsfoss.com/hacktoberfest-guide/>

## **What kind of contributions can you make?**

- Code
- Test
- Translate
- Support
- Document
- Promote
- Design

Contribute - Participate - Mozilla Thunderbird Project Website  
<https://www.thunderbird.net/en-US/participate/>

**“How do I get started?”**



**1. Know the project.**

## Know the project

Ideally, you should be familiar with the project you want to contribute to before you start contributing. (You are using the software.)

What you can do:

- Read the **README** file.  
Usually, the README file contains information about the project, the purpose/goal of the project, and how to get started with the project.
- Join the project's **mailing list** or **chat room** (if there is one), and talk to the maintainers and other users.
- Take a look at the project's **LICENSE**.  
The license tells you what you can and cannot do with the project's code (and/or other works). Does the license align with your values? Do you want your work to be put under that license?

**2. Get to know how to contribute.**

## Get to know how to contribute

Is it very *important* to know the contribute guidelines of the project you want to contribute to.

What you can do:

- Read the **CONTRIBUTING** file.

Usually, the CONTRIBUTING file contains information about how to contribute to the project. e.g. how to report a bug (info you need to submit, format to follow, etc.), how to configure your development environment, the project code style, how to submit a patch, who to reach out if you need help, etc.

- Read the project's **code of conduct**.

It is sometimes included in the CONTRIBUTING file, or it may be a separate file.

The code of conduct contains the rules you need to follow when you are being a part of the community of the project. Not following the code of conduct can lead to some serious consequences.

- Peek at *past contributions* and source codes.

### **3. Contribute.**

- Get the beta (testing) version of the software, and test it. If you find an issue, report it.
- Translate the software or its documentation to your language.
- Find an issue (bug report, feature request, or generally a task to work on) and work on it.
  - This can be code related, or not code related (e.g. documentation, design, etc.).
  - Find a small-sized issue (or tagged with beginner, good-first-issue, etc.) to start with. This will help you get familiar with the project's existing works and the contribution process.
- Talk to the maintainers and other contributors.

They can help you get started, helping you choose something to work on.

**“Where do I get started?”**

Like I said, ideally you should contribute to a software/library that you are using.

However, if you don't have a specific project in mind, here are some places you can look at:

- [The awesome-for-beginners repository on GitHub](#), which has a list of projects that are beginner-friendly.
- Events like [Hacktoberfest](#), which encourages people to contribute to open source projects. There will be projects that accept hacktoberfest contributions, and by contributing to them, you can get some swags/merchandise.



# Good Luck, Have Fun!

