



# REUSE

S O F T W A R E



---

## NGI Zero PET and Discovery Legal To-Dos

---

REUSE Recommended Best Practices



Published by the Free Software Foundation Europe (FSFE)  
Berlin, September 2019  
EU Transparency register ID: 33882407107-76  
[www.fsfe.org](http://www.fsfe.org)

Responsible according to European press law:  
Matthias Kirschner / FSFE e.V.  
Schönhauser Allee 6/7  
10119 Berlin  
Germany

Content of this report may be quoted or reproduced, provided that the source of information is acknowledged. All contents are, as long as not noted otherwise, licensed as CC BY-SA 4.0.

License information:  
<https://creativecommons.org/licenses/by-sa/4.0>

The REUSE graphic profile is available in Git:  
<https://git.fsfe.org/reuse/reuse-ci/>



## Table of Contents

<b>REUSE Initiative</b> .....	<b>4</b>
Choosing and Providing Licenses.....	4
Adding Copyright and Licensing Information to Each File.....	5
Resulting Project Tree.....	6
Confirming REUSE Compliance.....	7
Getting More Help.....	7

## REUSE INITIATIVE

The REUSE Initiative (or simply, “REUSE”), is a project of the Free Software Foundation Europe to develop awareness of the best practices for expressing license and copyright information in free and open source projects. It intends to facilitate management of source code by making licensing and copyright information more consistent in how it is added to source code in ways which allow for automating many of the processes involved.

This document will show you the basic methods to make a software project REUSE compliance. By the end of this document, all of your files should clearly have their copyright and licensing marked, and you will be able to verify this using a linter tool.

Making your project REUSE compliant can be done in three simple steps:

- Choose and provide licenses
- Add copyright and license information to each file
- Confirm REUSE compliance

You can use the following tutorial to do so. For the purposes of this tutorial, we will assume that you have chosen to use the GNU General Public License v3.0 or any later version (the “GPL”), and that the directory of your project looks like this:

```
project/
├── img/
│   ├── cat.jpg
│   └── dog.jpg
├── src/
│   └── main.c
├── .gitignore
├── Makefile
└── README.md
```

---

### Choosing and Providing Licenses

After choosing a license, you will need to put the license in your project directory. You can find your license in the [SPDX License List](#). SPDX is an open standard for communicating license and copyright information. Each license is uniquely identified by a shortform SPDX License Identifier. The SPDX License Identifier for your chosen license is GPL-3.0-or-later.

You create a LICENSES directory in your project root which will contain all the licenses that you use in your project. You then download your license from the [license-list-data](#) repository and put it in the LICENSES directory.

---

## Adding Copyright and Licensing Information to Each File

Now that you have a license, you need to indicate in the relevant files that these files fall under that license. You edit the comment header of `src/main.c` as follows:

```
/*
 * SPDX-FileCopyrightText: 2019 Jane Doe <jane@example.com>
 *
 * SPDX-License-Identifier: GPL-3.0-or-later
 */
```

The `SPDX-FileCopyrightText` tag records the publication years and the copyright holder of the contents of the file.

The `SPDX-License-Identifier` tag is followed by a valid [SPDX License Expression](#), typically just the SPDX License Identifier of the license.

Each file must always contain these two tags in the header. You are allowed to use the tags multiple times if you have multiple copyright holders or licenses.

In the example project, you also edit `Makefile` and `README.md` using this header information, but of course with corresponding comment syntax.

### 1. What do I do with binary and uncommentable files?

You will also want to license your image files under GPL-3.0-or-later. Unfortunately, images and other binary files do not have comment headers that one can easily edit. Other examples include automatically generated files and certain data and configuration files for which comments are non-trivial.

There is a simple trick to circumvent this. Create the files `cat.jpg.license` and `dog.jpg.license`, each containing the same information about license and copyright holder as above.

### 2. How do I change licensing information?

It may arise that the photos of the cat and the dog were not licensed under the GPL at all, but under the Creative Commons Attribution 4.0 International, and owned by John Doe. If this happens, then you should check the SPDX License Identifier of the license. In this case, it is `CC-BY-4.0`.

You should then create the file `LICENSES/CC-BY-4.0.txt`, following the same steps you used for GPL-3.0-or-later.

You can then edit `cat.jpg.license` and `dog.jpg.license` to say:

```
SPDX-FileCopyrightText: 2019 John Doe <john@doe.com>
```

```
SPDX-License-Identifier: CC-BY-4.0
```

### 3. Do I need to provide licensing information for build artifacts?

When you compile your program, you will generate some build artifacts, such as `src/main.o`. You do not need to provide any licensing information for those files. Just use your `.gitignore` file to ignore these build artifacts. The REUSE tool will respect the contents of `.gitignore`.

### 4. What do I do with insignificant files?

You probably will have files in your project that you do not find particularly copyrightable, for example configuration files such as `.gitignore`. Intuitively you may not want to license these files, but the fundamental idea of REUSE is that all your files will clearly have their copyright and licensing marked.

One way to indicate that you do not exercise any copyright over this file is by marking it as being in the public domain. The [CC0 license](#) is a good way to do this. Edit the file to contain:

```
# SPDX-FileCopyrightText: 2019 Jane Doe <jane@example.com>
#
# SPDX-License-Identifier: CC0-1.0
```

---

## Resulting Project Tree

The resulting project tree will now look like this:

```
project/
├── img/
│   ├── cat.jpg
│   ├── cat.jpg.license
│   ├── dog.jpg
│   └── dog.jpg.license
├── LICENSES/
│   ├── CC0-1.0.txt
│   ├── CC-BY-4.0.txt
│   └── GPL-3.0-or-later.txt
├── src/
│   └── main.c
├── .gitignore
├── Makefile
└── README.md
```

## Confirming REUSE Compliance

Now that you have marked all files with their copyright and licensing, it is time to check whether you did not miss anything. To do this, we provide a linter tool for you to use. You can read the [full documentation here](#), or read the quick steps below.

```
$ # Install the dependencies for the tool.
$ sudo apt install python3 python3-pip
$
$ # Install the tool
$ pip3 install --user fsfe-reuse
```

The executable is now in `$HOME/.local/bin/`. Make sure that this is in your `$PATH`. Now go to the project directory and run the linter.

```
$ cd path/to/project/
$ reuse lint
SUMMARY

Bad licenses: 0
Missing licenses: 0
Unused licenses: 0
Used licenses: CC-BY-4.0, CC0-1.0, GPL-3.0-or-later
Read errors: 0
Files with copyright information: 6 / 6
Files with license information: 6 / 6
```

### **Congratulations! Your project is REUSE compliant :-)**

As you can see in the last line, the tool confirm will confirm whether your project is compliant with REUSE standards. To learn what the different numbers mean, please have a look at the full documentation of the reuse tool.

---

## Getting More Help

We hope that this tutorial has helped you to understand how to adopt REUSE, and the three basic steps to properly license your software project. Although we have covered a few edge cases, you might run into more questions soon. Here are some more resources for your reference if you need more help:

- Our [Frequently Asked Questions](#) covers common questions as well as extraordinary cases and will constantly be updated.
- The full [REUSE specification](#) formally describes REUSE and the methods to become compliant.
- The [REUSE tool documentation](#) describes installation and usage of the REUSE tool.



If none of the links above were able to answer your question, please contact us by:

- opening an issue on [reuse-docs](#) for questions on the tutorial, FAQ or specification;
- opening an issue on [reuse-tool](#) for questions on the REUSE tool;
- or [sending an email to the FSFE](#). Please note that we would prefer issues because they are publicly searchable for other people.

Thank you for your valuable contribution towards making software reusable!

---