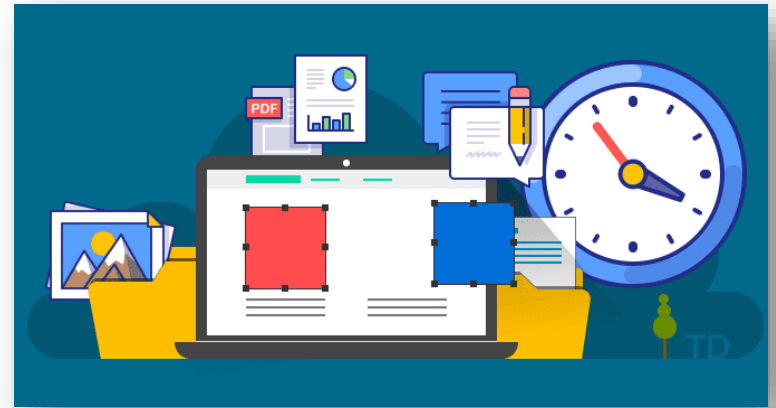




1. How long to develop a website?



2. How long to deploy the website?



3. Can I earn money from available software?



On giant shoulders



Open Source Software Development

Ung Văn Giàu
Email: giau.ung@eiu.edu.vn

If you have an **apple** &
I have an **apple** & we exchange **apples**
then you & I will still
each have one **apple**.



But if you have an **idea** &
I have an **idea** & we exchange these **ideas**
then each of us will have **2** ideas.

~ George Bernard Shaw



Introduction to Open Source Software Development



1

What is **free** software?

2

What is **open source** software?

3

What motivates developers to become involved in free / open source software projects?

4

How are their projects **coordinated**, and what is the software that they **produce like**?

Contents

01

Free software

02

Open source software

03

Types of software

04

Motivation

05

Advantages and Disadvantages

06

Licenses & Standards



What is free software?

100% FREE



The Free Software Definition

- “Free software” means software that respects users’ freedom and community
- **The users have the freedom to run, copy, distribute, study, change and improve the software**
- “Free software” is a matter of liberty, not price
It’s sometimes called “libre software”

Statue of Liberty





V/S



Developer

- Can control the program
- Can modify code to do expected works

User

- Uses the built-in features
- Cannot change the program

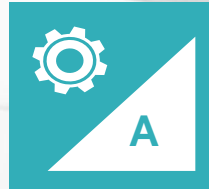
The Free Software Definition

- With these **freedoms**, the **users** (both individually and collectively) **control the program** and what it does for them
- When users don't control the program, we call it a “**nonfree**” or “**proprietary**” program
The nonfree program controls the users, and the developer controls the program

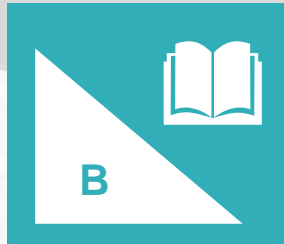
The Free Software Definition

The 4 essential freedoms

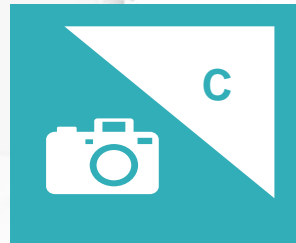
The freedom **to run**



The freedom **to study and change** it



The freedom **to redistribute** copies



The freedom **to distribute copies** of your modified versions



The Free Software Definition

The 4 essential freedoms:

- The freedom **to run** the program as you wish, for any purpose.
- The freedom **to study** how the program works and **change** it so it does your computing as you wish.

Access to the source code is a precondition for this.

- The freedom **to redistribute** copies so you can help others.
- The freedom **to distribute copies of your modified versions** to others. By doing this you can give the whole community a chance to benefit from your changes.

Access to the source code is a precondition for this.

The Free Software Definition

“Free software” does not mean “noncommercial”.





History of the Open Source Initiative (OSI)

History of the OSI

- The Open Source Initiative (OSI) is a California public benefit corporation, founded in 1998
- The “open source” label was created at a strategy session held on February 3rd, 1998 in Palo Alto, California, shortly after the announcement of the release of the Netscape source code
- Adoption of the term was swift, with early support from figures in the community
Linus Torvalds, and many key individuals, including the founding figures of sendmail, Perl, Python, Apache, and representatives from the IETF and Internet Software Consortium

History of the OSI

- One of the first tasks undertaken by OSI was to draft the Open Source Definition (OSD), and use it to begin creating a list of OSI-approved licenses
- The Open Source Definition was originally derived from the Debian Free Software Guidelines (DFSG)
- By Oct. 1999, OSI had published its first formal list of approved licenses
- The OSI logo, which combines the “O” of open and a keyhole, for unlocking source code, was created by Colin Viebrock





What is Open Source software?

The Open Source Definition

Open source doesn't just mean access to the source code.

The distribution terms must comply with the following **criteria**:

- **Free Redistribution**

- not restrict any party from selling or giving away the software as a component

- **Source Code**

- must include source code
 - allow distribution in source code as well as compiled form

- **Derived Works**

- allow modifications and derived works
 - allow them to be distributed under the same terms

The Open Source Definition

The distribution terms must comply with the following **criteria**:

- **Integrity of The Author's Source Code**

may restrict source-code from being distributed in modified form only if the license allows the distribution of “patch files” with the source code

- **No Discrimination Against Persons or Groups**

- **No Discrimination Against Fields of Endeavor**

- **Distribution of License**

The rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties

The Open Source Definition

The distribution terms must comply with the following **criteria**:

- **License Must Not Be Specific to a Product**

The rights attached to the program must not depend on the program's being part of a particular software distribution

- **License Must Not Restrict Other Software**

The license must not place restrictions on other software that is distributed along with the licensed software

- **License Must Be Technology-Neutral**

No provision of the license may be predicated on any individual technology or style of interface



Types of Software

Classification based on the **availability** and **shareability**

Freeware



- **Gratis** programs
- Normally only distributed in binary format, and can be obtained free of charge
- Sometimes it is possible to obtain permission to redistribute, and sometimes not
- Frequently used to promote other programs (normally with more complete functionality)
- Examples: Skype, Adobe Reader, Team Viewer,...



Shareware



- Freely distributed to users on a trial basis
- Usually comes with a time limit and when the time limit expires, the user is asked to pay for the continued services
- Examples: Adobe Acrobat, Kaspersky Internet Security,...



kaspersky

Charityware, careware



- This is normally shareware that requires payment to be directed towards a sponsored charitable organization
- In many cases, instead of demanding payment, a voluntary contribution may be requested
- Examples: Vim, PeaZip,...



Public domain



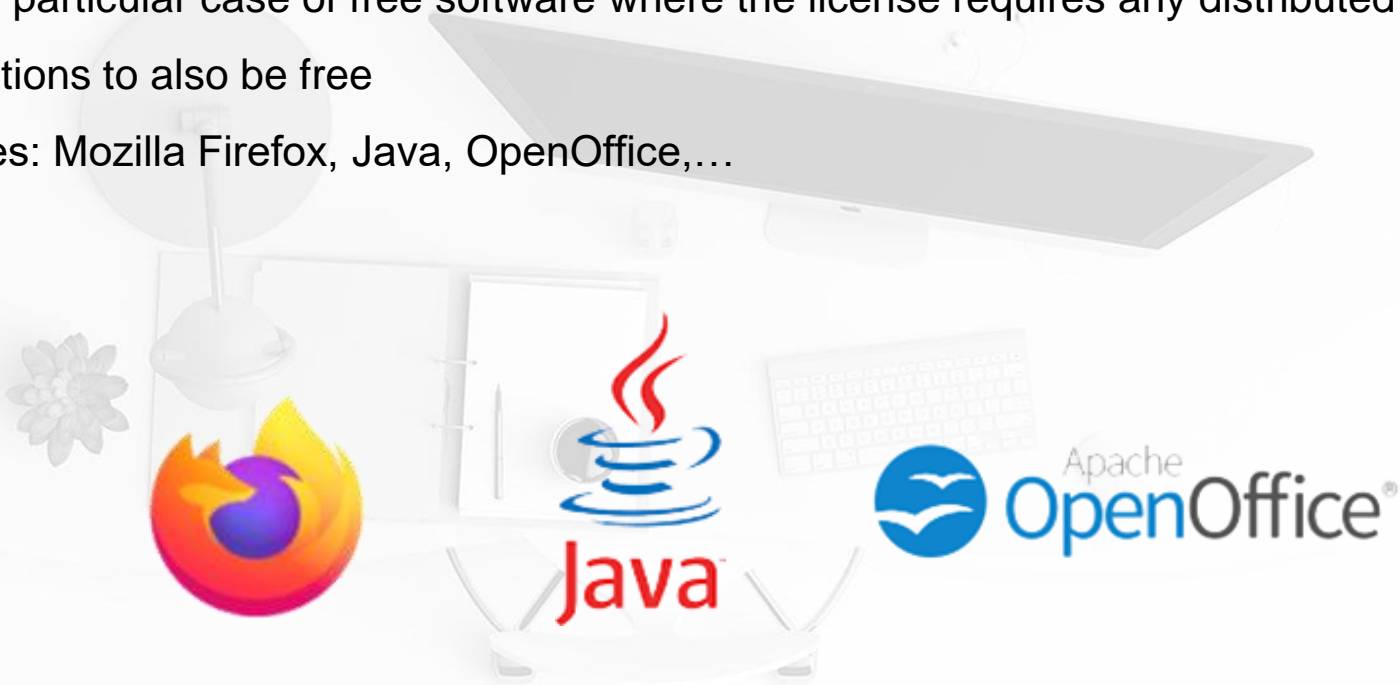
- The author totally renounces all his rights in favor of the public domain
- If the source code is provided → the program is **free**
- Examples: SQLite, 7-Zip,...



Copyleft



- This is a particular case of free software where the license requires any distributed modifications to also be free
- Examples: Mozilla Firefox, Java, OpenOffice,...














Proprietary, locked-in, non-free

- Refer to software that is neither free nor open source
- Examples: Microsoft Windows, Adobe Photoshop, WinRAR,...



Compare free and open-source software, public domain and freeware

	 Free software	 Open-source software	 Freeware	 Public-domain software
Definition	"FREE" is a matter of liberty, not price	"OPEN" doesn't just mean access to the source code	"FREE" refers to price, while freedom of the use is restricted by creator	"PUBLIC DOMAIN" belongs to the public as a whole
Ground philosophy	Social movement	Development methodology	Marketing goals	Copyright disclamation
Ground rules	Four Freedoms https://www.gnu.org/philosophy/free-sw.html	Open Software initiative https://opensource.org/osd		Creative Common Organization https://creativecommons.org
Free of charge	Not necessary	Not necessary	✓ YES	✓ YES
Covered by copyright law	✓ YES	✓ YES	✓ YES	✗ NO
Examples	   		 	



Motivations

Motivations of individuals

Motivation in Free/Open Source Software Development

Four distinct clusters:

- Express **enjoyment** and **learning** as primary motivators
- Need the code to **satisfy non-work-related user needs**
- Have **work-related needs** and **career** concerns
- **Feel an obligation** to the **community** and believe that software should be free/open



The Advantages and Disadvantages

Advantages

- **Cheaper** than commercially marketed products
 - Be accessible to anyone
 - Work with almost any type of computer (old hardware)
- Created by skillful and talented people
- Highly reliable
 - developed chiefly by skillful and talented experts
 - worked on by tens or hundreds of people
- Help you become **more flexible**
don't need to abide by a specific IT architecture

Disadvantages



- Vulnerable to malicious users



- Might not be as user-friendly as commercial versions



- Don't come with extensive support
low cost and easy accessibility come with a trade-off



Licenses & Standards

About Open Source Licenses

- Open source licenses must comply with the Open Source Definition (OSD)
In brief, they allow software to be freely used, modified, and shared
- To be approved by OSI, a license must go through the license review process

Popular Licenses

The following OSI-approved licenses are popular, widely used, or have strong communities:

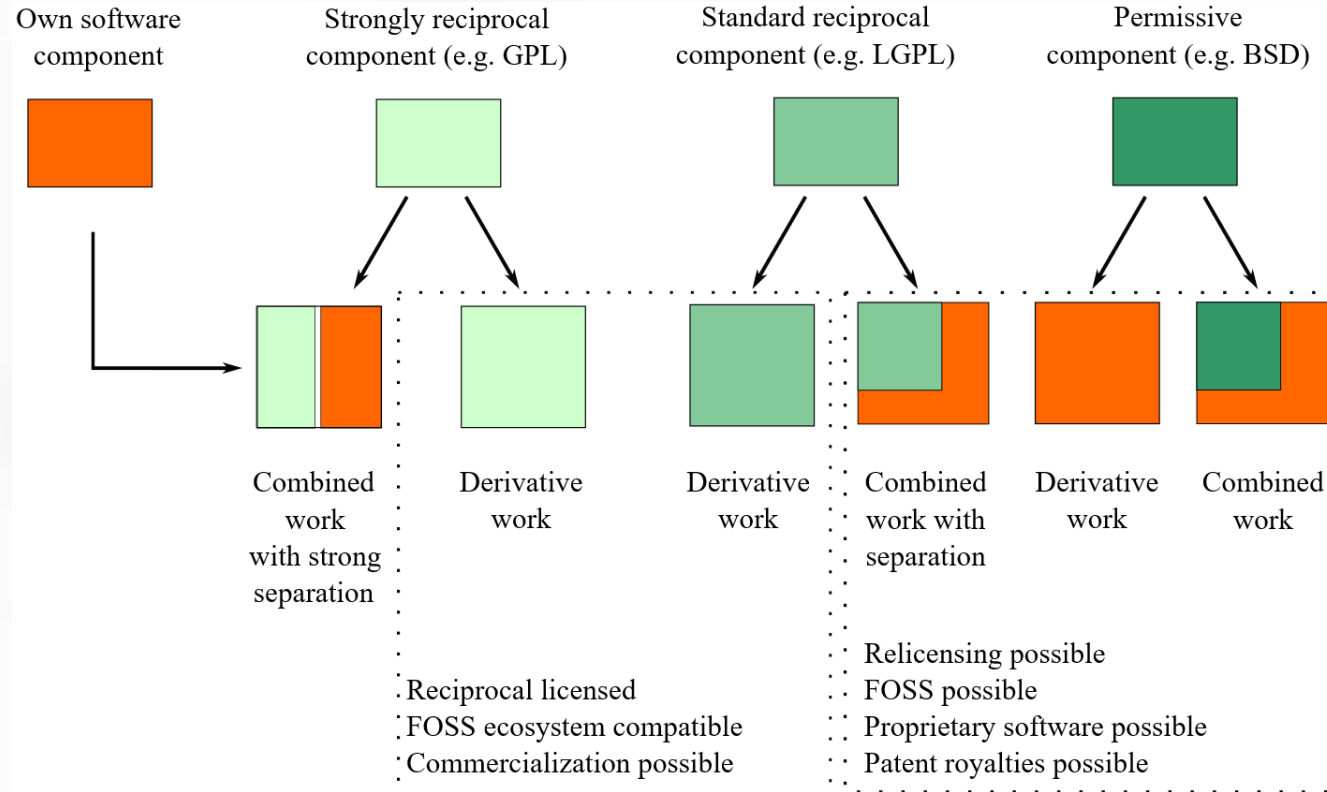
- Apache License 2.0
- BSD 3-Clause “New” or “Revised” license
- BSD 2-Clause “Simplified” or “FreeBSD” license
- GNU General Public License (GPL)
- GNU Library or “Lesser” General Public License (LGPL)
- MIT license
- Mozilla Public License 2.0
- Common Development and Distribution License
- Eclipse Public License version 2.0

The most popular open-source software licenses comparison

							
Type	Permissive	Permissive	Permissive	Permissive	Copyleft	Copyleft	Copyleft
Provides copyright protection	✓ TRUE	✓ TRUE	✓ TRUE	✓ TRUE	✓ TRUE	✓ TRUE	✓ TRUE
Can be used in commercial applications	✓ TRUE	✓ TRUE	✓ TRUE	✓ TRUE	✓ TRUE	✓ TRUE	✓ TRUE
Provides an explicit patent license	✓ TRUE	✗ FALSE	✗ FALSE	✗ FALSE	✗ FALSE	✗ FALSE	✗ FALSE
Can be used in proprietary (closed source) projects	✓ TRUE	✓ TRUE	✓ TRUE	✓ TRUE	✗ FALSE	✗ FALSE partially	✗ FALSE for web
Popular open-source and free projects	Kubernetes Swift Firebase	Django React Flutter	Angular.js JQuery, .NET Core Laravel	Joomla Notepad++ MySQL	Qt SharpDevelop	SugarCRM Launchpad	

License compatibility

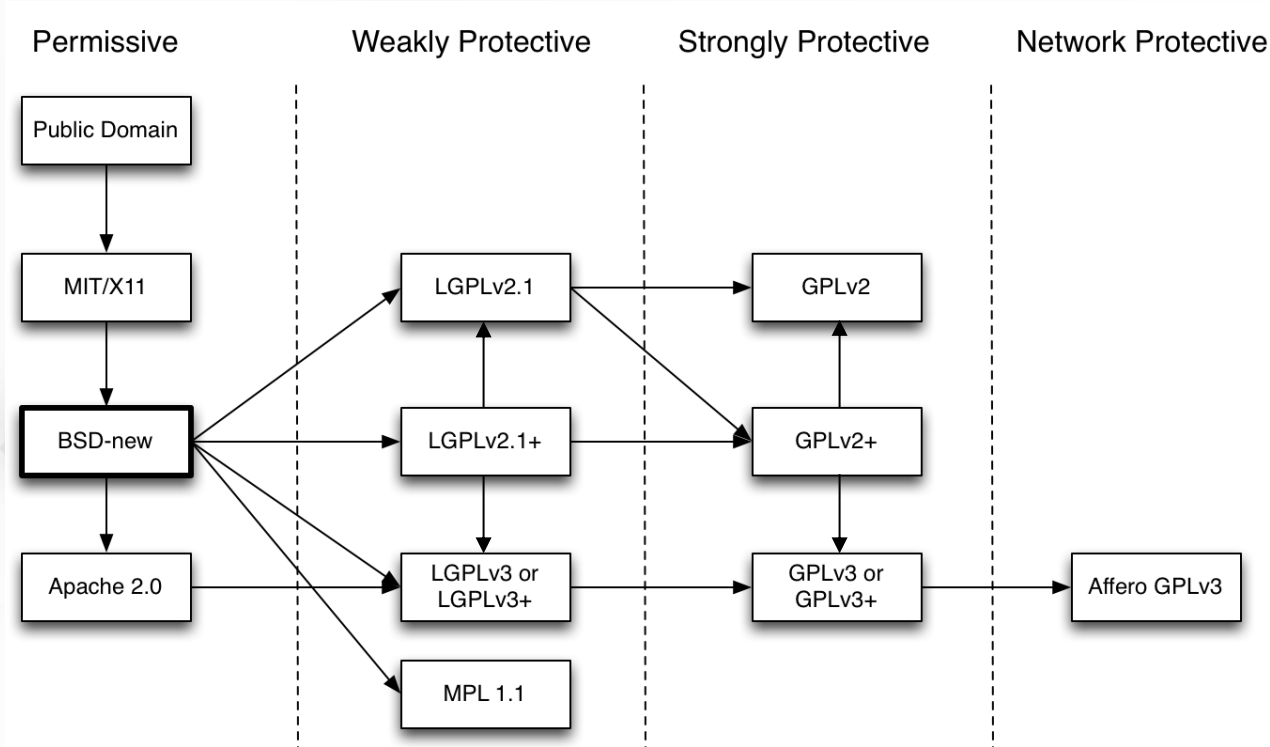
for derived works and combined works



Comparison of Open Source Licenses

License	Release changes under a different license	Link with code using a different license	GPL Compatible
Apache License 2.0	Yes	Yes	Yes
GNU GPL	No	No	Yes
GNU LGL	No	Yes	Yes
Prior BSD Licenses	Yes	Yes	No
New BSD License	Yes	Yes	Yes
MIT License	Yes	Yes	Yes
Ms-PL	No	No	No

License compatibility between common FOSS software licenses



By David A. Wheeler (2007)

License compatibility between common FOSS software licenses

- Box A → box B:
 - You can combine software with these licenses
 - The combined result effectively has the license of B, possibly with additions from A
- To see if software can be combined, just start at their respective licenses, and find a common box you can reach following the arrows
- I.e., Apache 2.0-licensed software and GPLv2+-licensed software can both reach “GPLv3 or GPLv3+” → so they can be combined using GPLv3 or GPLv3+



The License Review Process

The goal of the OSI License Review Process

- To ensure that licenses and software labeled as “open source” conform to existing community norms and expectations
- For that reason, all licenses must go through a public review process

Purpose of the Process

- Ensure approved licenses conform to the Open Source Definition and provide software freedom
- Identify appropriate License Proliferation Category
- Discourage vanity and duplicative Licenses
- Ensure a thorough, transparent and timely review (e.g., within 60 days)

How to Submit a Request

1. Read the OSD and ensure that your license complies with it
2. Identify the type of submission (Retirement, Legacy Approval or Approval)
3. Ensure you have appropriate standing to submit the type of submission that you have identified
4. Subscribe to license-review (if you aren't already)
5. Submit a formal request to license-review.

How to Submit a Request

The request email must include:

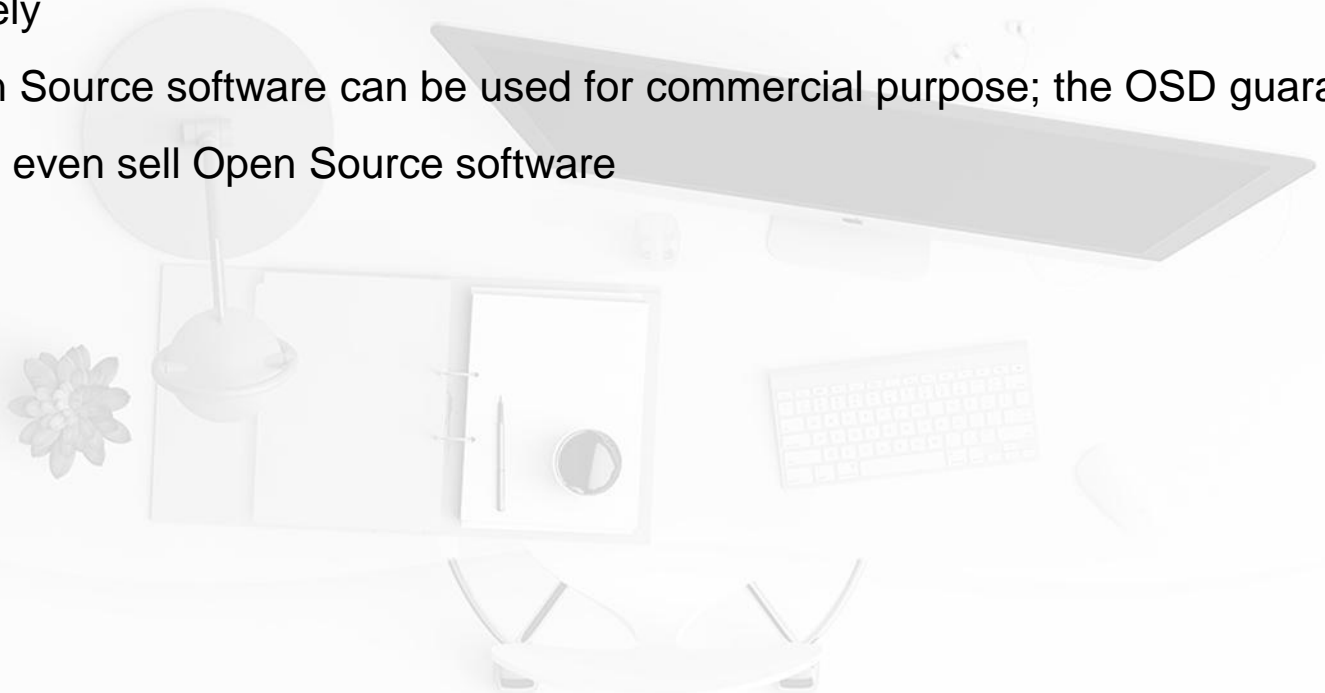
- the submission type and license name in subject field (to ensure proper tracking)
- a plaintext copy of the license
- the supporting data (as appropriate for the type of submission)
- a link to earlier public discussions (if any)



Frequently Answered Questions

Can Open Source software be used for commercial purposes?

- Absolutely
- All Open Source software can be used for commercial purpose; the OSD guarantees this
- You can even sell Open Source software



Can I restrict how people use an Open Source licensed program?

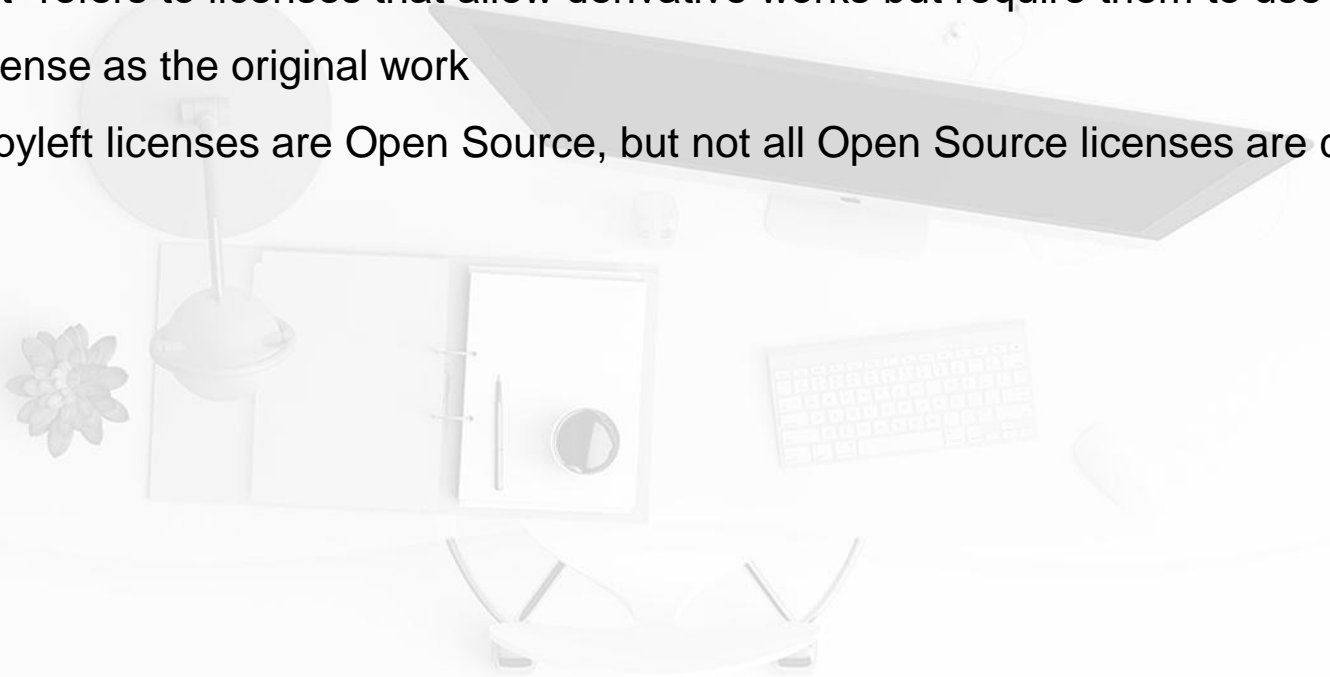
- No. The freedom to use the program for any purpose is part of the Open Source Definition.
- Open source licenses do not discriminate against fields of endeavor.



What is “copyleft”?

Is it the same as “open source”?

- “Copyleft” refers to licenses that allow derivative works but require them to use the same license as the original work
- Most copyleft licenses are Open Source, but not all Open Source licenses are copyleft



Reference

- GNU Operating System, **What is free software?**, <https://www.gnu.org/philosophy/free-sw.en.html>
- Open Source Initiative, **The Open Source Definition**, <https://opensource.org/docs/osd>
- Squareboat, **Different Types of Software with Examples**, <https://squareboat.com/blog/different-types-of-software-with-examples>

References

- **How to Contribute to Open Source**, <https://opensource.guide/how-to-contribute/>
- **What is open source?**, <https://opensource.com/resources/what-open-source>
- **Frequently Answered Questions**, <https://opensource.org/faq#approved-licenses-only>
- **Categories of free and nonfree software**,
<https://www.gnu.org/philosophy/categories.html>
- **Which Open Source License?**, <http://www.bitsandbuzz.com/article/which-open-source-license/>
- **Licenses**, <https://choosealicense.com/licenses/>



Q&A