

# Committing to Open Source

An overview about open source software and how *you* can get involved!

# Well, who am I? Parthiv Menon.

## Education

I am currently an MS CS graduate student at Northeastern University. And yes, I have taken PDP this semester :)

## Work wise

I am a full stack developer, experienced in working with various development environments and diverse teams.

## Open source stuff

- Worked as a Technical Fellow with the Wikimedia Foundation from Oct 2022 to Dec 2023
- Interacted with communities and organised triaging meetings for tech and non-tech tasks that helped build the community

## What about security?

- Deeply involved with Manipal Information Security Team in my undergraduate days
- Conducted security awareness events, workshops, played CTFs

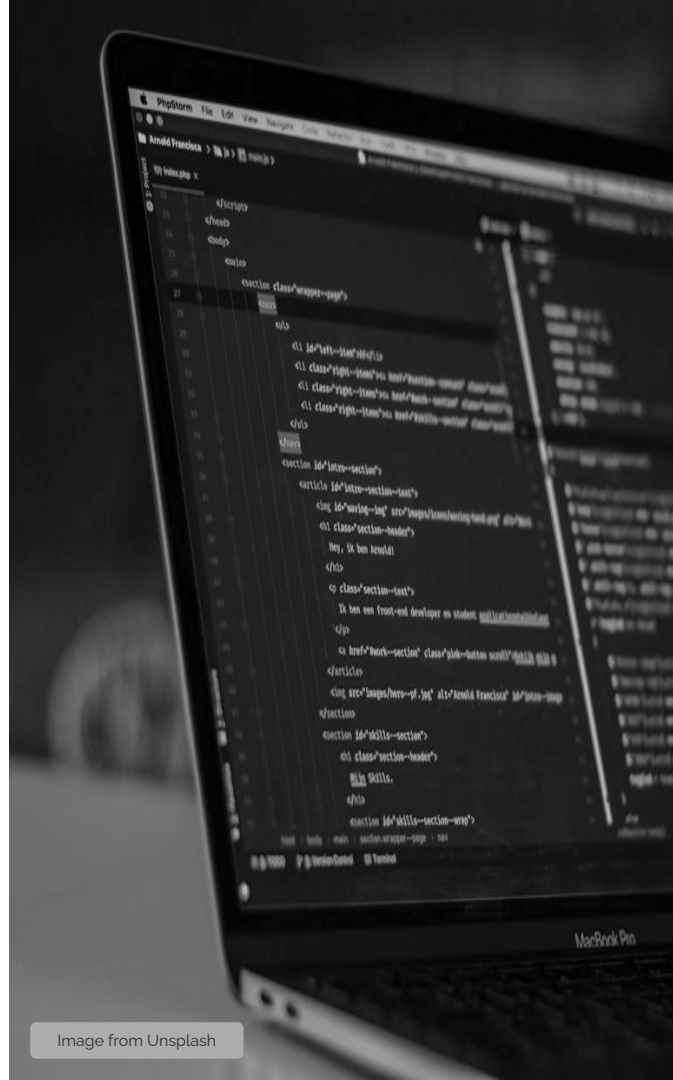
#opensource

#opensource

**what • why • who • how**

# ./opensource/what

- Social movement that **rejects secrecy and centralized control** and favours **decentralization, transparency, and unrestricted sharing** of information
- Good open source projects make **code, meta files, instructions/documentation, tasks related to the project**, and a whole lot of other data available to the public!
- Open source **does not mean** free of charge
- Free of charge **does not mean** open source



# ./opensource/why

## Control

- More control to users, developers, and other individuals interested
- Change, modify, improve parts that you do not like!
- Inspect what is going on behind the scenes

## Training

- Helps non-programmers learn to code in the best possible way
- Better coding practices are followed
- Mistakes and doubts during the learning process follow an open-to-all-resolvable-by-all policy
- Easy to onboard students and people with non-tech background onto projects/teams

# ./opensource/why

## Stability

- Since every release affects users (who are also contributors in various capacities), OSS is generally stable
- Possibly faster bug resolution, feature updates
- Procedure of operation of the project is predetermined (licenses, roadmaps, software standards)

## Community

- Most projects have vibrant communities supporting it in various ways
- Not just a fanbase—people taking time out to contribute to something they love collectively

## Security

- Most OSS have high privacy and security standards (as people are directly involved)
- Considerable control over telemetry, data sharing, software security, etc.

# ./opensource/who

Short answer? **Anyone.**

- Contributions can be technical or non-technical
  - ◆ Code/deployment (most)
  - ◆ Documentation (most projects require this!)
  - ◆ Design (open source design, icon libraries, etc.)
  - ◆ Events/workshops (awareness workshops by projects)
  - ◆ Conferences/summits (Community days, tech conferences)
  - ◆ Lately open source hardware projects have gotten a boost!
  
- Two broad rules
  - ◆ No one can force anyone else to include contribution in any project
  - ◆ Everyone can create their own version of anyone's project (with the correct licensing)

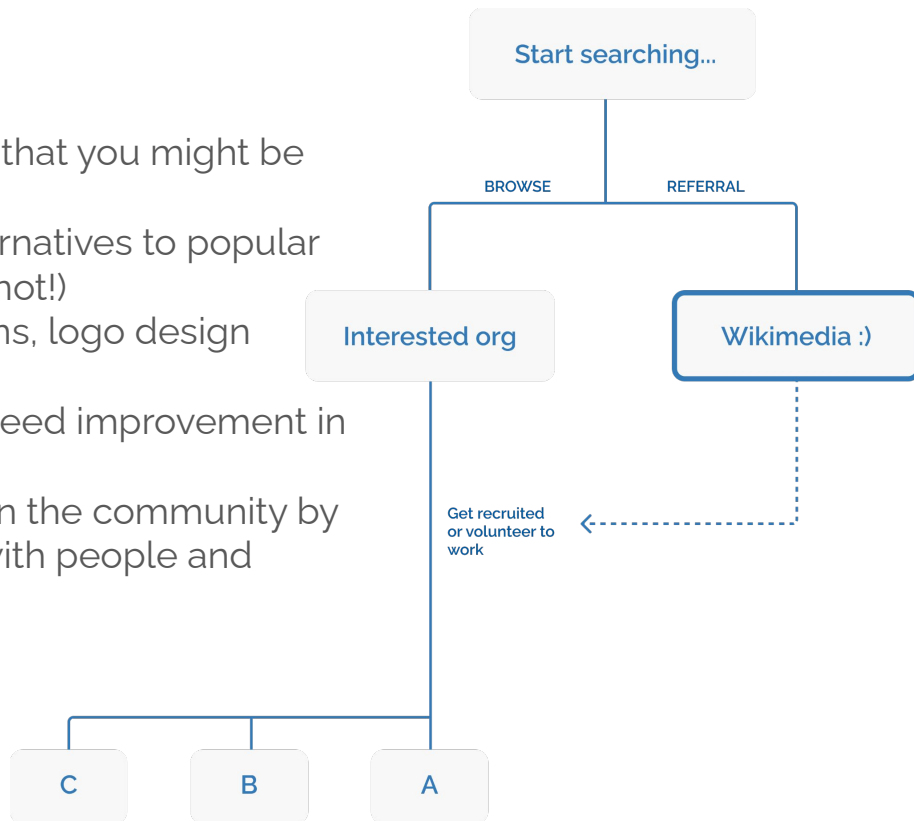


Image from wCommons

# ./opensource/how

**Shortlist.** Approach. Familiarise and contribute!

- Browse the web and make a list of projects that you might be interested in
- ◆ Developer? Look for open source alternatives to popular products (they exist, more often than not!)
  - ◆ Designer? Look for UI/UX contributions, logo design opportunities on projects.
  - ◆ Good with words? Look for OSS that need improvement in docs or other content.
  - ◆ Good with people? You can be a star in the community by helping to coordinate tasks, interact with people and organising events!



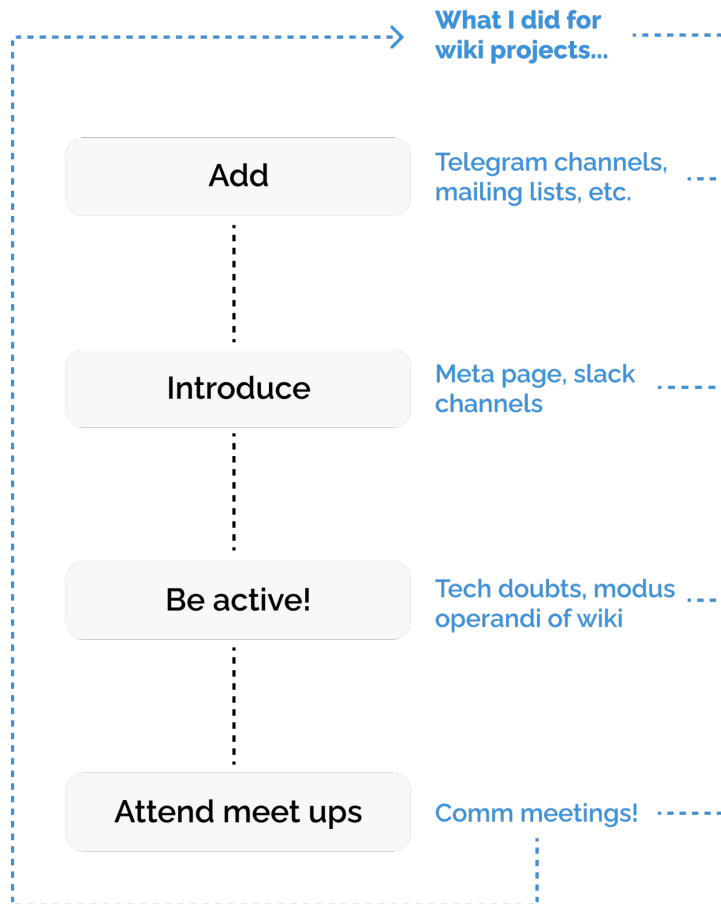


# ./opensource/how

Shortlist. **Approach.** Familiarise and contribute!

- All good open source projects have clear communication channels
- Take initiative and introduce yourself on the group/channel. People in communities love other people joining their communities!
- Join triaging discussions and/or welcome meetings—you might even get your first task there!

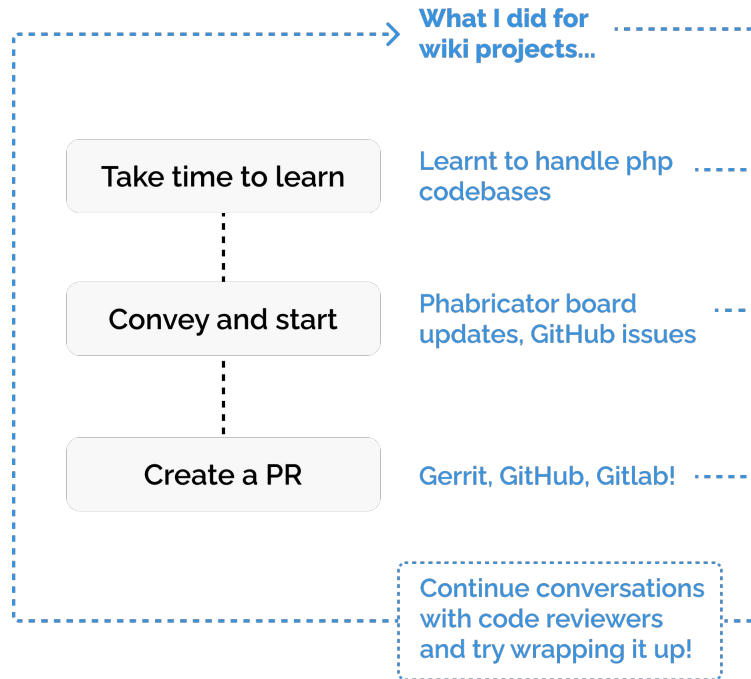
**Don't hesitate to make the first move!**



# ./opensource/how

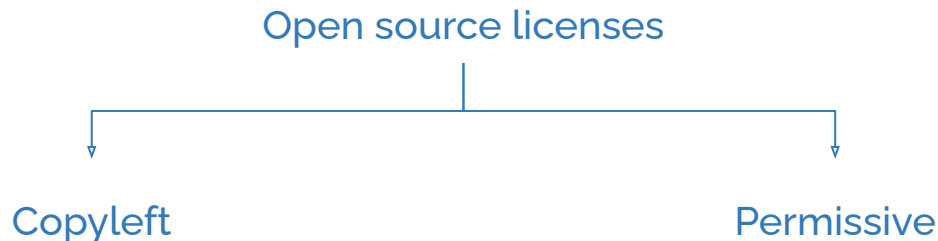
Shortlist. Approach. **Familiarise and contribute!**

- Get yourself comfortable with the part of the project you intend to work on: codebase, documentation, design
- Clearly state to the community what task you intend to work on
  - ◆ Prevents duplication of effort
  - ◆ Helps moderators and other maintainers prioritize tasks
- Create that first PR and you are in!



# Licenses in FOSS

- Allows content to be **used, modified, and shared**
- Important to ensure that software remains in the open domain, with proper attributions to creators and owners



- Obligated to provide source code
- People must release derivative works under a license offering these same freedoms
- Protects against being used in proprietary software
- Examples include GPL

- Academic license
- No obligation to provide source code
- Code may be used in proprietary software
- Examples include BSD (by UC Berkeley), MIT (by MIT), Apache (by Apache Foundation)

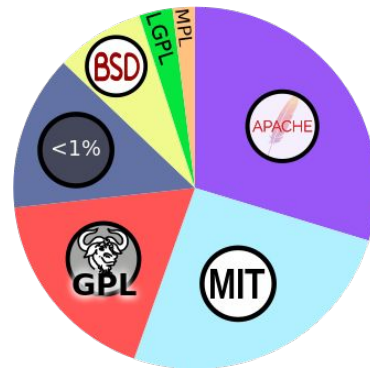


Image from Wikipedia

# Learn Git



**And look for the signs...**

# Oops, that repository **does not exist!**

Your first hint that the project needs help!

## Why version control?

- Makes progress visible to everyone
- Easier to collaborate with other people—too many cooks do not spoil the broth here!

## Why GitHub?

- 100,000,000 developers—largest community amongst VCS platforms
- Intuitive GUI for beginners, suite of CLI features for advanced users

**Pro tip** Get the Student Developer pack that gives you GitHub Pro as long as you are a student



# When you do find one, it **might** look like this

spf13 / cobra

Type to search

<> Code Issues 182 Pull requests 85 Actions Projects Wiki Security Insights

cobra Public

Watch 362 Fork 2.8k Star 35.4k

Tells you how interested the contributors/users of the project are

Rough estimate of how people are contributing to the project

Go to file Add file Code

Smidra Add Taikun CloudWorks to list of projects (#2098) ✓ bcfcf7 · 2 months ago 1,033 Commits

How actively updates and enhancements are made to the project

Code of conduct for the project

Instructions on how to contribute

Forks and stars tell you how highly the community rates the project!

Description of the project and other instructions (like setup/installation/usage)

The license under which the project is distributed

File	Commit Message	Time Ago
.github	build(deps): bump actions/labeler from 4 to 5 (#2016)	3 months ago
assets	Update main image to better handle d...	last year
doc	Fix linter errors (#2052)	5 months ago
site/content	Add Taikun CloudWorks to list of projects (#2098)	2 months ago
.gitignore	leverage makefile to run build tasks (#976)	4 years ago
.golangci.yml	golangci: enable 'unused' and disable deprecated replaced b...	8 months ago
.mailmap	Add .mailcap for a more tidy "git shortlog" output	9 years ago
CONDUCT.md	Cobra User Contract (#1292)	3 years ago
CONTRIBUTING.md	Add CONTRIBUTING.md (#1183)	4 years ago
LICENSE.txt	adding license	11 years ago
MAINTAINERS	Add MAINTAINERS file (alphabetical order) (#1545)	2 years ago
Makefile	Makefile: add target richtest (#1865)	2 years ago

About

A Command-Line Interface (CLI) for managing your projects.

cobra.dev

go cli golang commandline command command-line posix subcommands cli-app golang-library cobra golang-application cobra-library cobra-generator command-cobra

Readme

Apache-2.0 license

Activity

35.4k stars

362 watching

2.8k forks

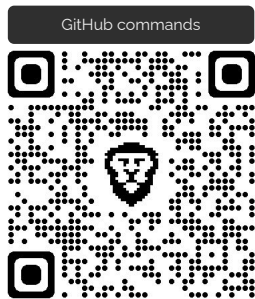
Report repository

Releases 21

# What else do you need to look for?

- Typically, testing in open source projects depends on the scale of the project
  - ◆ Huge projects like Apache and Firefox may have rigorous test suites and E2E testing
  - ◆ Smaller projects have comparatively relaxed testing—depends on contributors' bias
- Documentation is sometimes as overlooked as testing
  - ◆ Quite a lot of projects have really good documentation
  - ◆ Some lack *any* documentation because individual contributors do not feel the necessity to do it—this deters prospective contributors

**Look for (or create) good documentation and ensure you pass test suites!**



# Contributing **securely** with GitHub

- Setup 2FA on your account
  - ◆ Makes sure your account cannot be breached and used for malicious contributions
- Use SSH keys in your development environments
  - ◆ Ensures seamless integration with GitHub securely
- Setup Personal Access Tokens (PAT) with required scope
  - ◆ Github deprecated the use of password based auth for git operations in August 2021
  - ◆ Provides granular control over your repos
  - ◆ Useful in scripts or other CLI applications that use your GitHub account





# DEMO

**Setting up SSH keys and PATs**

# Prospects of **earning with FOSS**



Google Summer of Code

12 weeks  
USD 3000  
(mid-sized project)



Google Season of Docs

Each project receives \$5,000 -  
\$15,000 USD depending on  
requirements



12 weeks  
Flat USD 7000



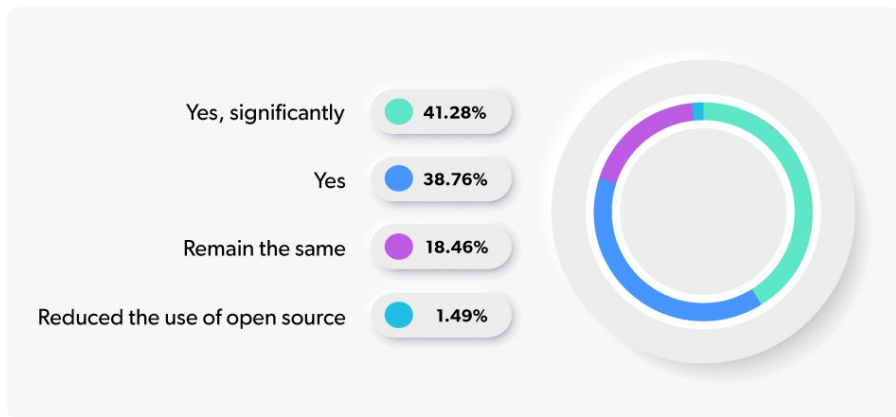
Funding depends on the  
project, not always a certainty

- GNOME and Linux Kernel mentorship programs
- Work with the Free Software Foundation
- Major League Hacking has an open source track
- [Start your own project!](#)

# Careers in FOSS

As of 2021, Linux sysadmins earn *more* than their Windows counterparts => better jobs, better pay, better job security! (as per [payscale.com](#))

## Why consider at all?



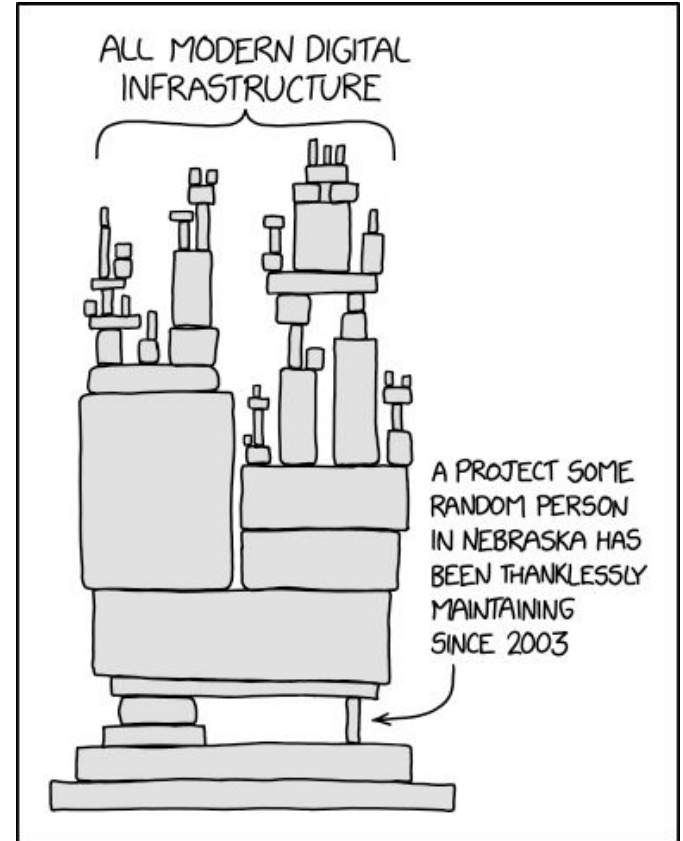
Increase in open source usage in companies according to State of Open Source report 2023 by the Open Source Initiative

- Increasing adoption of cloud technologies automatically fuels the open source software movement
- Google, AWS, NASA, Spotify (and more) are allocating specific budgets for open source projects



*The world is opening up to open source.*

**What about you?**



*Thank you!*