<https://medium.com/better-programming/linux-survival-guide-for-beginners-c18bfd982036>

[Sign in](https://medium.com/m/signin?operation=login&redirect=https%3A%2F%2Fmedium.com%2Fbetter-programming%2Flinux-survival-guide-for-beginners-c18bfd982036&source=post_page-----c18bfd982036---------------------nav_reg-)

Get started

[[](https://medium.com/better-programming?source=post_page-----c18bfd982036----------------------)](https://medium.com/better-programming?source=post_page-----c18bfd982036----------------------)

* [Archive](https://medium.com/better-programming/archive?source=post_page-----c18bfd982036----------------------)
* [Write For Us](https://medium.com/better-programming/write-for-us-5c4bcba59397?source=post_page-----c18bfd982036----------------------)
* [About](https://medium.com/better-programming/about?source=post_page-----c18bfd982036----------------------)

**Linux: A Survival Guide for Beginners**

**I survived the transition to Linux—and you can too**

[[](https://medium.com/@gokulnk?source=post_page-----c18bfd982036----------------------)](https://medium.com/@gokulnk?source=post_page-----c18bfd982036----------------------)

[Gokul N K](https://medium.com/@gokulnk?source=post_page-----c18bfd982036----------------------)

Follow

[Mar 29](https://medium.com/better-programming/linux-survival-guide-for-beginners-c18bfd982036?source=post_page-----c18bfd982036----------------------) · 10 min read

https://miro.medium.com/max/60/1*XXI-kg18liPn4XcfZmoqQQ.jpeg?q=20

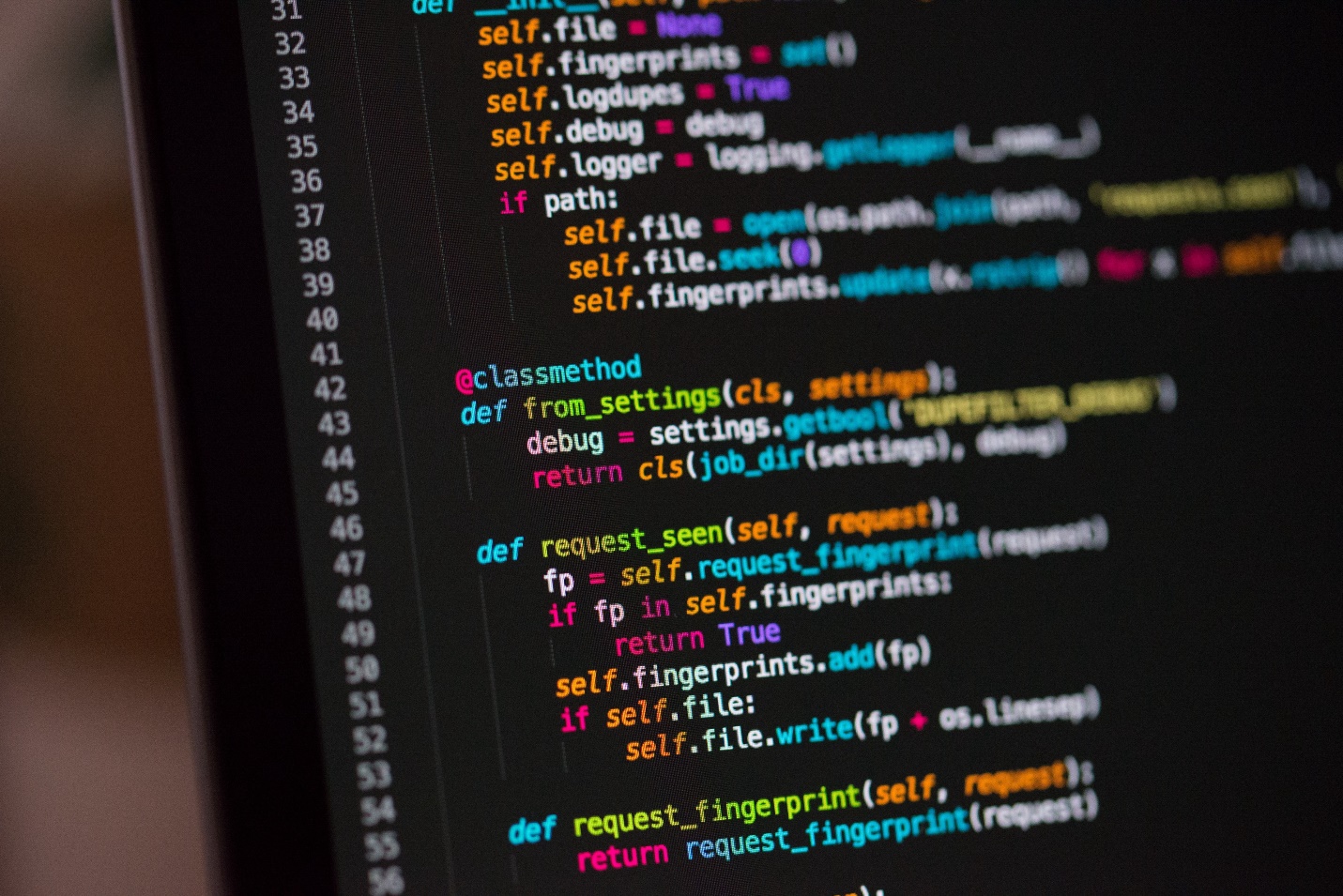
<img alt="" class="gl n o gk ab" src="https://miro.medium.com/max/12032/1\*XXI-kg18liPn4XcfZmoqQQ.jpeg" width="6016" height="4016" role="presentation"/>

Photo by [Chris Ried](https://unsplash.com/@cdr6934?utm_source=unsplash&utm_medium=referral&utm_content=creditCopyText) on [Unsplash](https://unsplash.com/search/photos/code?utm_source=unsplash&utm_medium=referral&utm_content=creditCopyText)

Switching from Windows to [Linux](https://www.linux.org/) can be really scary. But if you can survive the first few months, the eventual returns are exponential. Here’s how I survived.

**TLDR**

1. Though I am just an amateur in Linux, I was able to survive the transition and indeed benefitted from it. So these are my notes for somebody facing a similar situation.
2. Pick [Ubuntu](https://ubuntu.com/) to start with. Choose other flavours once you know better and can decide for yourself.
3. Get comfortable with following commands ssh, pwd, ls, cd, mv, cp, scp, grep, find, rm. Tip: you can use <https://tldr.ostera.io/cp> to get the list of most frequently used options of these commands.
4. Learn to use the | symbol. Using this symbol, you can pass the output of one command as an input to the next command.

**The Long Version**

In my first company, we were using Windows extensively, whether it was desktop machines we used for development or the servers on which we deployed our code. But when I moved to my [second company](http://azrisolutions.com/?source=gokulnk), it was all in on OSS and hence using Linux was mandatory there. It became a herculean task for me. For the first month or so it was a nightmare.

Having gone through that nightmare and survived it, I am making this list to help others like me who are trying to make this transition.

**New Environments**

Transitions are hard in general. New environments can be scary. If you’re a Windows user who has never used the command line, then transition to Linux can get really scary. Don’t fret because that is generally the experience of many people who are making this transition. Knowing that others also find it difficult can be really consoling at times.

According to me, the two main reasons that make transitions difficult are: **lack of familiarity** and **fear of screwing up.**

**Lack of familiarity**

To address the issue of familiarity, I started using Linux on my office laptop as well as my personal laptop. I started reading blogs about Linux and followed some interesting Linux-related accounts on Twitter. I reached out to people who were good at Linux. I would walk up to their cubicles and ask them to show me their command history. I learned a lot more from this than from reading the blogs. Most of the times, since it’s just muscle memory, the programmers can’t explain it. But their history is a treasure trove.

I would recommend running the following command. You will get many insights into the commands your Linux heroes use frequently.

history | awk ‘{ $1=””; print $0 }’ | sort | uniq -c | sort -nr | head -20

Run the command on the terminals of the Linux gurus in your office. Ask them about the commands you are not familiar with and you should be able to learn a lot more than what a couple of books could teach you. Don’t forget that these are battle-tested commands and hence much more valuable than standard examples in blogs. If you cannot understand what the above command does, don’t worry—I’ll explain it later on in the piece.

**Fear of screwing up**

I have been using Linux for a couple of years but I still have this fear. This fear was multi-fold when I started. One thing that helped me a lot was, I spoke to Linux pros in my company and made a blacklist—a list of commands that I should never use or use with caution. sudo rm -rf was the top of the list. If you are anxious like me you can use <https://github.com/nivekuil/rip> on your local machine.

When I was going through the stage of being afraid of screwing up two Twitter users were of great help: [chakri\_iiith](https://twitter.com/chakri_iiith) and [tuxnani](https://twitter.com/tuxnani). I wish I had spent more time with them and learnt a couple more of their tricks. Find your angels and they will help you face your fears.

Now that your fears are addressed, let’s get started.

**Why You Should Learn Linux**

There are countless reasons why you should learn Linux. A google search will fetch you thousands of articles about why you should learn Linux, such as “[What are the benefits of learning Linux](https://www.quora.com/What-are-the-benefits-of-learning-Linux),” “[Why you should switch to Linux](https://fossbytes.com/10-reasons-switch-linux-os-right-now/)” and “[Is it worth my time to learn Linux while learning programming](https://www.reddit.com/r/learnprogramming/comments/38zytg/is_it_worth_my_time_to_learn_linux_while_learning/)?” Those three articles are worth a read, but here are my top two reasons why you should learn:

1. **Linux is ubiquitous**: Linux is everywhere. So with or without your knowledge there is a high probability that you are already using or benefitting from Linux. Understanding the basics of Linux can therefore come in handy in many situations. If you are a programmer then that chance is fairly high. A fair number of applications are deployed on Linux servers. So learning it can be a lifesaver.
2. **Linux is versatile**: [Both Linux and MAC are built on UNIX.](https://askubuntu.com/a/11396/217036) So if you are comfortable with the Linux terminal you should be able to use most of the commands in MAC terminal as well. [Android uses the Linux kernel.](https://unix.stackexchange.com/questions/25463/does-android-really-use-the-same-kernel-as-linux) [Raspberry Pi uses Linux.](https://www.raspberrypi.org/documentation/linux/kernel/building.md) [Many embedded devices use Linux.](https://en.wikipedia.org/wiki/Linux#Embedded_devices)

**Why Did You Start Learning Linux?**

As we’ve seen, there are many reasons for you to learn Linux. But if you are a programmer, there’s a fair chance that you fall into one of the two following categories:

1. You read up about the cool things that Linux can do or you heard from a friend who just can’t stop raving about Linux.
2. Your laptop or desktop has a non-Unix OS. But your application or website is deployed on a Linux server.

If you fall under the first category, you have all the time in the world. So take your sweet little time. If you fall under the second category, then there is a fair chance that you are running against a deadline. So finish the next parts and get your hands dirty.

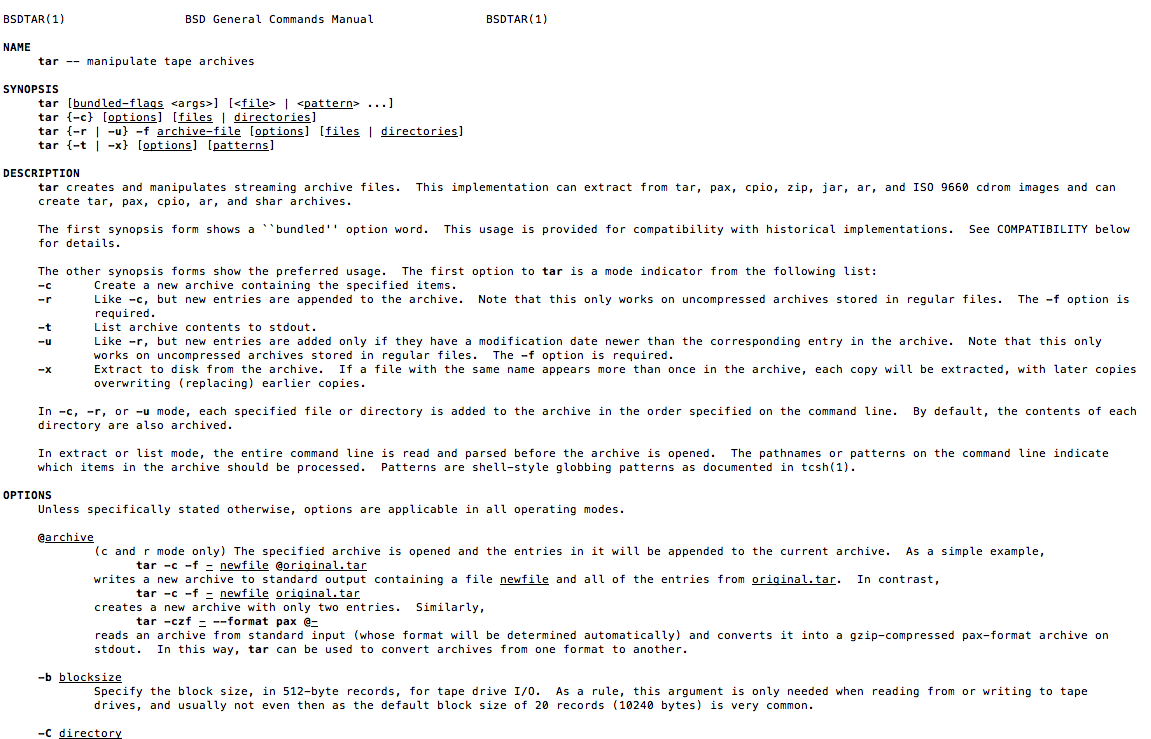
**Man Command Is Your Friend. Or Is It?**

One of the first tips you get when you want to learn Linux is “Use man command, it is your friend.” While there is a certain truth to it, it can be overwhelming for many first-time users. All you generally need are the options for the most frequently used scenarios of the command—and that is what is precisely missing from man pages. Luckily for you, there’s a project called [TLDR](https://tldr.sh/) which is trying to fix exactly that.

Just compare the outputs of these two commands to see what I mean.

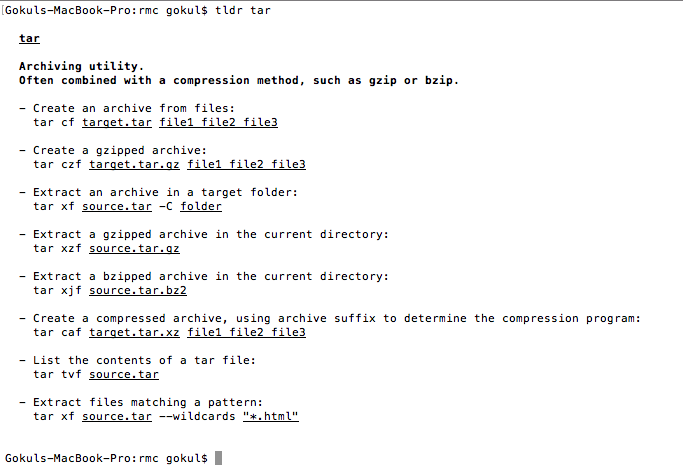
**First, output from man pages.**

https://miro.medium.com/max/60/1*0QXjvacgzf1OHvziiX9s2w.png?q=20

<img alt="" class="gl n o gk ab" src="https://miro.medium.com/max/2302/1\*0QXjvacgzf1OHvziiX9s2w.png" width="1151" height="739" role="presentation"/>

Now the output from TLDR project.

https://miro.medium.com/max/60/1*ZrSUDueSxi2uezlLqoq1iQ.png?q=20

<img alt="" class="gl n o gk ab" src="https://miro.medium.com/max/1366/1\*ZrSUDueSxi2uezlLqoq1iQ.png" width="683" height="472" role="presentation"/>

Do you see the difference?

TLDR is like the notes about commands I would have written for myself. I find it very handy. I installed the TLDR using [nodejs](https://nodejs.org/) command sudo npm i -g tldr. If you have not installed nodejs I suggest you do it, as there are many node packages that are very handy. You can install nodejs using [this installation manual by Digitial Ocean.](https://www.digitalocean.com/community/tutorials/how-to-install-node-js-on-ubuntu-18-04)

I thought of sharing my notes on all the commands in this post, but then I came across a post by Andrew where he covers 101 bash commands:

**[101 Bash Commands and Tips for Beginners to Experts](https://dev.to/awwsmm/101-bash-commands-and-tips-for-beginners-to-experts-30je?source=post_page-----c18bfd982036----------------------" \l "whereis-which-whatis)**

**[Andrew Jan 13 ・39 min readThe commands below are laid out in a more-or-less narrative style, so if you're just getting…](https://dev.to/awwsmm/101-bash-commands-and-tips-for-beginners-to-experts-30je?source=post_page-----c18bfd982036----------------------" \l "whereis-which-whatis)**

**[dev.to](https://dev.to/awwsmm/101-bash-commands-and-tips-for-beginners-to-experts-30je?source=post_page-----c18bfd982036----------------------" \l "whereis-which-whatis)**

He has categorised all the commands and has good examples as well, and I can’t do a better job than that.

**Learn About Bash Profiles**

I found bash config files or bash profiles to be handy, so it helps to [learn the differences and how they work](https://stackoverflow.com/a/415444/493742).

One rule of thumb I follow is to add all my configs to .bash\_profile and also make sure to load .bashrc within the .bash\_profile file. I add my favourite aliases to this file. I keep a basic version of my .bash\_profile in my private gist and I download the raw version of that on the servers where I need these.

**Learn to Use Emacs**

One thing that I look for these days is commonality. This has helped me leverage what I know in multiple scenarios. For example, we are pushing a lot of Javascript in our team as we can use it in multiple scenarios, like our website front end, in browser console to scrape things quickly, debug our front end or learn from other websites, for joining collections in mongodb, and in nodejs for server side.

Stressing commonality helps us “Learn Once, Benefit Multiple Times”—a much greater ROI.

Coming back to Linux, I wanted to decide on a command line editor. I had the options of choosing Nano, Vim, or Emacs. I chose [Emacs](https://www.gnu.org/software/emacs/emacs.html).

Most of the commands used in Emacs can also be used in Linux shell. For example, you can use CTRL/CMD+A to go to the beginning of the line on both shell and Emacs. There are many such commands which work in both shell and Emacs. I think this is a huge advantage.

Since it is a command line editor, you can install it easily on any server. On every server where I am root I generally install Emacs. I am not sure if this is a good practice, but I generally find it very convenient. Yes I have decided not to learn Nano or Vim. Roast me for it if you want to.

**PIPE It**

Pipe command in Linux lets you use the output of one command as the input of the next command. This can be really helpful once you get the hang of a few Linux commands like grep, sort, awk, uniq, head, and tail. Piping along with these commands is immensely powerful. For example, I never remember what the options are in ls for showing only text files (and I don’t think you should either). I just run the following command.

ls -l | grep txt

I know this is quick and dirty but it works in most scenarios.

For example, if we look at the history processing command we used in the first section:

history | awk ‘{ $1=””; print $0 }’ | sort | uniq -c | sort -nr | head -20

We are taking the output of the history command, and we are passing it to awk to remove the line numbers at the beginning of each line from the output. Then we are passing the output to the sort command so that we can sort it. Then we are passing the output uniq command to retain only unique lines along with the number of occurrences. Then we are passing it to sort command to sort it in reverse order. Then we are passing it to head command to list only the top 20 most frequently used commands that are present in our history.

How cool is that?

**GREP it**

If you are used to SDKs and GUI editors, GREP might seem little limited. But most of the time, the differentiator is that you can chain the output of the grep command. That is very handy. Most of the time I am not really worried about the performance of the grep queries. The difference is insignificant. Only when the performance of the grep query matters I spend time on checking the options and fine-tuning it. In all other cases, I find it better to beat the heck out of grep and pipe to get what I want.

So, for example, when I wanted to check the list of services that were loaded I quickly looked at the pattern and used the following command.

systemctl list-units --all | **grep** service | **grep** loaded

What I have done here is, I have reduced the result set to those lines that contain the word service, and from that result set I am again filtering for the lines that also contain the wordloaded. Let’s say in addition to that you want to remove the lines that contain exited, then you can modify the query to:

systemctl list-units --all | grep service | grep loaded | grep -v exited

Piping grep with grep gets you what you want most of the time without remembering any other options.

There are many other tricks like these. But I feel these should be sufficient to get you hooked on Linux and motivate you to start exploring it further.

**What Can You Do in Four Lines of Linux Code?**

Short answer: a lot. My friend [Chakri](https://medium.com/u/ad308d423076?source=post_page-----c18bfd982036----------------------) scraped 1500 web pages, took the relevant information from those pages, and created a handy table to do his own analysis. The code is:

#!/bin/sh  
for i in {10000..11500}  
do  
 wget -O — [http://](http://racetime.in/2019nebwdr/?bibNo=)[resultsite.com](http://racetime.in/2019nebwdr/?bibNo=11019\&submit=SUBMIT)[/2019nebwdr/?bibNo=](http://racetime.in/2019nebwdr/?bibNo=)"$i"\&submit=SUBMIT | grep “10 KM” | sed -E ‘s/<tbody>|<\/tb\  
ody>|<\/table>//g’ >> output.html  
done

If you want to read how the above script works, [check out his post.](https://medium.com/@chakri_iiith/scraping-the-results-of-a-10k-run-for-fun-28144bb2dc93)

**Before We Close**

For me, the primary goal while working on Linux was not to get overwhelmed by the command line and the learning curve. I designed my learning path to suit my requirements. Do let me know if this works out for you. If you are also a user who has successfully transitioned from another OS to Linux, do share your notes.

**Understanding Series**

This article is a part of my learning series [**Understanding X**](https://medium.com/@gokulnk/declaration-of-intent-2018-f222ce76b6c8).

**Understanding Linux**

**[Linux Survival Guide for Beginners](https://hackernoon.com/linux-survival-guide-for-beginners-c18bfd982036?source=post_page-----c18bfd982036----------------------)**

**[Survival guide for those who are starting with Linux](https://hackernoon.com/linux-survival-guide-for-beginners-c18bfd982036?source=post_page-----c18bfd982036----------------------)**

**[hackernoon.com](https://hackernoon.com/linux-survival-guide-for-beginners-c18bfd982036?source=post_page-----c18bfd982036----------------------)**

**[Understanding Zabbix](https://hackernoon.com/understanding-zabbix-f2a83eeb1221?source=post_page-----c18bfd982036----------------------)**

**[Monitor all your server infrastructure](https://hackernoon.com/understanding-zabbix-f2a83eeb1221?source=post_page-----c18bfd982036----------------------)**

**[hackernoon.com](https://hackernoon.com/understanding-zabbix-f2a83eeb1221?source=post_page-----c18bfd982036----------------------)**

**[Understanding UFW](https://hackernoon.com/understanding-ufw-8d70d5d8f9d2?source=post_page-----c18bfd982036----------------------)**

**[Ubuntu Firewall for humans](https://hackernoon.com/understanding-ufw-8d70d5d8f9d2?source=post_page-----c18bfd982036----------------------)**

**[hackernoon.com](https://hackernoon.com/understanding-ufw-8d70d5d8f9d2?source=post_page-----c18bfd982036----------------------)**

**[Linux Systemctl](https://hackernoon.com/linux-systemctl-46bd0a11e27b?source=post_page-----c18bfd982036----------------------)**

**[Monitoring services and starting them on boot](https://hackernoon.com/linux-systemctl-46bd0a11e27b?source=post_page-----c18bfd982036----------------------)**

**[hackernoon.com](https://hackernoon.com/linux-systemctl-46bd0a11e27b?source=post_page-----c18bfd982036----------------------)**

**[Better Programming](https://medium.com/better-programming?source=post_sidebar--------------------------post_sidebar-)**

**Advice for programmers.**

Follow

**747**

* [Programming](https://medium.com/tag/programming)
* [Linux](https://medium.com/tag/linux)
* [Ubuntu](https://medium.com/tag/ubuntu)
* [Shell](https://medium.com/tag/shell)
* [Command Line](https://medium.com/tag/command-line)

**747 claps**

[[](https://medium.com/@gokulnk?source=follow_footer--------------------------follow_footer-)](https://medium.com/@gokulnk?source=follow_footer--------------------------follow_footer-)

Written by

[**Gokul N K**](https://medium.com/@gokulnk?source=follow_footer--------------------------follow_footer-)

Follow

**Inquisitive, Student, Teacher and a wanna be Story teller. Working on https://learningpaths.io/**

Follow

[](https://medium.com/better-programming?source=follow_footer--------------------------follow_footer-)

[**Better Programming**](https://medium.com/better-programming?source=follow_footer--------------------------follow_footer-)

Follow

**Advice for programmers.**

Follow

[See responses (5)](https://medium.com/p/c18bfd982036/responses/show?source=follow_footer--------------------------follow_footer-)

**More From Medium**

**More from Better Programming**

**More from Better Programming**

**[The End of PHP?](https://medium.com/better-programming/the-end-of-php-18ba209dbc6d?source=post_recirc---------0------------------)**

[Simon Tony](https://medium.com/@inhereat?source=post_recirc---------0------------------)

[Simon Tony](https://medium.com/@inhereat?source=post_recirc---------0------------------) in [Better Programming](https://medium.com/better-programming?source=post_recirc---------0------------------)

[Aug 27](https://medium.com/better-programming/the-end-of-php-18ba209dbc6d?source=post_recirc---------0------------------) · 4 min read

**1.1K**

**More from Better Programming**

**More from Better Programming**

**[Simplify your JavaScript with these 6 Array Methods](https://medium.com/better-programming/simplify-your-javascript-with-these-6-array-methods-db4c278f08c9?source=post_recirc---------1------------------)**

[[Aphinya Dechalert](https://medium.com/@PurpleGreenLemon?source=post_recirc---------1------------------)](https://medium.com/@PurpleGreenLemon?source=post_recirc---------1------------------)

[Aphinya Dechalert](https://medium.com/@PurpleGreenLemon?source=post_recirc---------1------------------) in [Better Programming](https://medium.com/better-programming?source=post_recirc---------1------------------)

[Aug 30](https://medium.com/better-programming/simplify-your-javascript-with-these-6-array-methods-db4c278f08c9?source=post_recirc---------1------------------) · 5 min read

**1K**

**More from Better Programming**

**More from Better Programming**

**[How To Learn a New Programming Language Fast](https://medium.com/better-programming/how-to-learn-a-new-programming-language-fast-5fe269370210?source=post_recirc---------2------------------)**

[[Jun Wu](https://medium.com/@junwu_46652?source=post_recirc---------2------------------)](https://medium.com/@junwu_46652?source=post_recirc---------2------------------)

[Jun Wu](https://medium.com/@junwu_46652?source=post_recirc---------2------------------) in [Better Programming](https://medium.com/better-programming?source=post_recirc---------2------------------)

[Aug 27](https://medium.com/better-programming/how-to-learn-a-new-programming-language-fast-5fe269370210?source=post_recirc---------2------------------) · 6 min read

**1K**

**[Discover Medium](https://medium.com/about?autoplay=1&source=post_page-----c18bfd982036----------------------)**

Welcome to a place where words matter. On Medium, smart voices and original ideas take center stage - with no ads in sight. [Watch](https://medium.com/about?autoplay=1&source=post_page-----c18bfd982036----------------------)

**[Make Medium yours](https://medium.com/topics?source=post_page-----c18bfd982036----------------------)**

Follow all the topics you care about, and we’ll deliver the best stories for you to your homepage and inbox. [Explore](https://medium.com/topics?source=post_page-----c18bfd982036----------------------)

**[Become a member](https://medium.com/membership?source=post_page-----c18bfd982036----------------------)**

Get unlimited access to the best stories on Medium — and support writers while you’re at it. Just $5/month. [Upgrade](https://medium.com/membership?source=post_page-----c18bfd982036----------------------)

[About](https://medium.com/about?autoplay=1&source=post_page-----c18bfd982036----------------------)[Help](https://help.medium.com/?source=post_page-----c18bfd982036----------------------)[Legal](https://medium.com/policy/9db0094a1e0f?source=post_page-----c18bfd982036----------------------)

To make Medium work, we log user data. By using Medium, you agree to our [Privacy Policy](https://medium.com/policy/f03bf92035c9), including cookie policy.