

[LTCWM \(https://learntocodewith.me/\)](https://learntocodewith.me/) > [Blog \(https://learntocodewith.me/blog/\)](https://learntocodewith.me/blog/) > [What to learn \(https://learntoco...](https://learntocodewith.me/blog/what-is-command-line-interface-cli/)



What Is Command Line Interface (CLI)? Command Line for Beginners

Updated on October 28th, 2021 | Sign up for [learn to code tips](https://learntocodewith.me/newsletter/) (<https://learntocodewith.me/newsletter/>)

What is a CLI, aka command line interface? In short, the command line interface is a way of interacting with your computer. And if you ever want to be a serious programmer, or even half-serious, you need to learn command line skills.

Admittedly, the CLI looks intimidating to a lot of first-timers. (It was for me.) *But it's not that hard to learn command line basics.*

In this article, my friend Michael Gattozzi is joining us to help explain some key command line concepts! Let's dive into more details on the command line for beginners and how to use command line programming to interact with your computer.

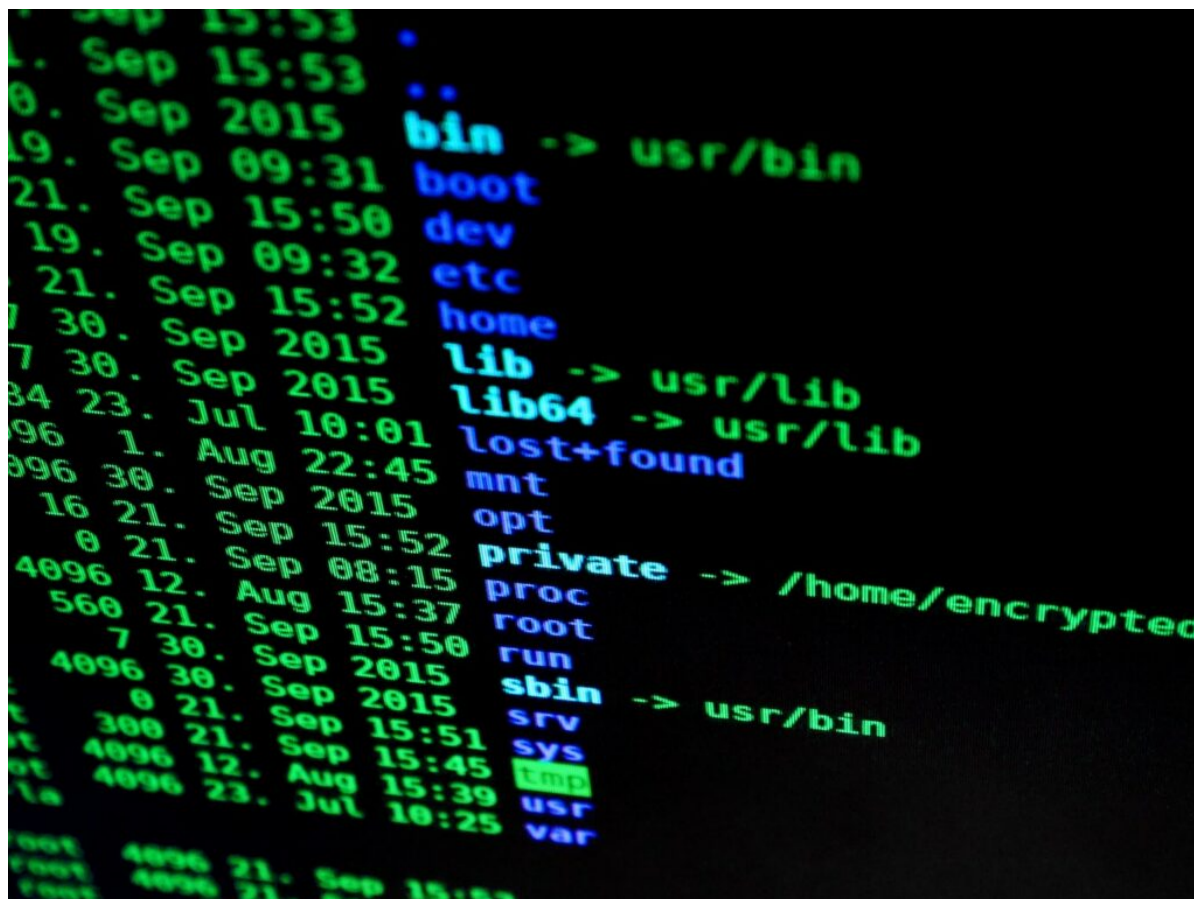
Table of Contents

- [What Is Command Line Interface/CLI?](#)
- [Shell vs. Bash vs. Terminal vs. Command Prompt](#)
- [What is the Difference Between the CLI and GUI?](#)
- [Why Should I Learn the Command Line?](#)
- [What Does the Command Line Look Like?](#)
- [How to Learn Command Line Skills](#)

Disclosure: I'm a proud affiliate for some of the resources mentioned in this article. If you buy a product through my links on this page, I may get a small commission for referring you. Thanks!

What Is Command Line Interface/CLI?

💡 The command line, or command-line interface (CLI), is a text-based interface that allows you to type commands via lines of text that are then processed by the computer.



For example, you can learn how to use command line skills to quickly rename hundreds of files in a folder, completely erase a drive, navigate through files and folders, or pull up a list of every program installed on your computer — without ever using your mouse!

Depending on whether you use Windows or Mac/Linux, there will be differences in the CLI.

As Michael notes, “most distributions of Linux and Mac run bash as their choice of shell for the terminal.” Nonetheless, all of the commands are standard across Unix machines.

While Windows does have a command line interface, unless you work in IT or are doing Penetration Testing (<https://learntocodewith.me/learn/penetration-testing/>), Michael wouldn't recommend learning it as a prerequisite. The Windows commands and how its CLI works are different from the Linux and Mac CLI.

Learn what the command line is, what it's used for, and how to learn it in this post! (<https://twitter.com/share?text=Learn+what+the+command+line+is%2C+what+it%E2%80%99s+used+for%2C+and+how+to+learn+it+in+this+post%21&via=learncodewithme&related=learncodewithme&url=https://learntocodewith.me/learn/command-line/>)

IN+THIS+POST%21&VIA=LEARNCODEWITHME&RELATED=LEARNCODEWITHME&URL=HTTPS://LEARNTOCODEWITH.ME/LEARN/COMMAND-

 [Back to the table of contents](#)

What is the Difference Between Shell, Bash, Terminal, and Command Prompt?

When we talk about the command line, there are a few terms that might come up and that you should know.

The term “shell” is typically used to reference any command line interface. It's the software that interprets and executes the commands. On a Mac, the CLI application is called “Terminal,” while on Windows, it's called “Command Prompt.”

Bash is a type of shell and stands for “Bourne Again Shell.” It is usually the default shell in Linux systems, but there are other shells available, including ksh, tcsh, dash, and zsh.

 [Back to the table of contents](#)

What is the Difference Between the CLI and GUI?

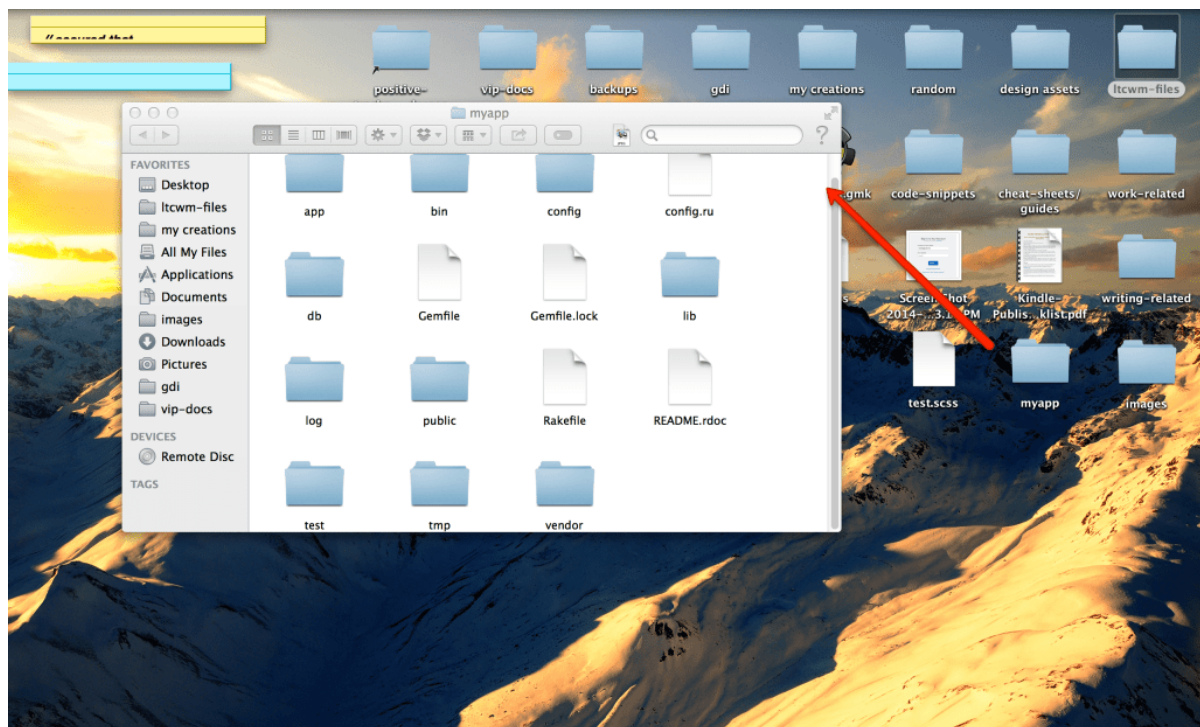
Both the CLI and the Graphical User Interface (GUI) allow users to interact with a computer system. However, there are a few differences.

As Michael explains,

“When you use programs with graphical user interfaces (GUI) certain buttons perform certain tasks on the computers. The command line interface does that too, but it allows you to do it with more precision and power. You type words and hit enter, the shell interprets those words, and works with the OS kernel and files to execute the command.”

Okay, this may sound really confusing. So let me break it down.

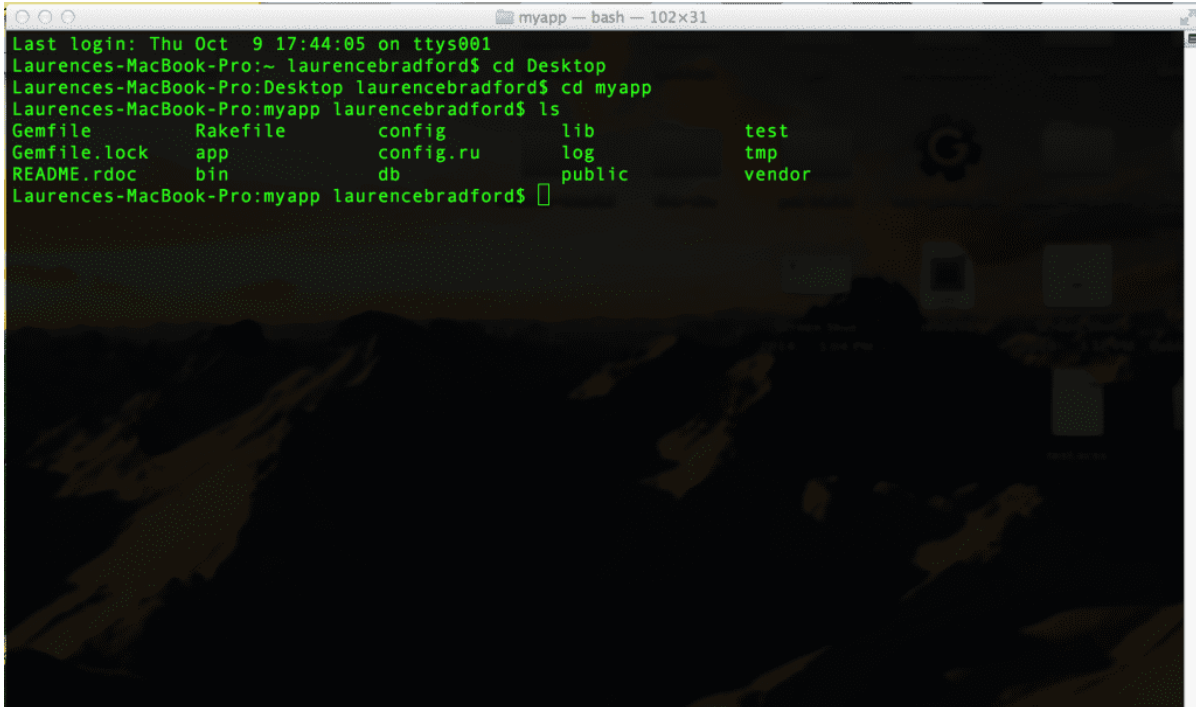
The GUI is what most people use 95% of the time to tell their computer what to do. Below is the GUI version of opening the folder “myapp” on my desktop.



(<https://learntocodewith.me/getting-started/topics/command-line/attachment/gui/>)

That looks normal, right?

The image below has been taken inside the CLI on my computer. Here, I am also opening the folder “myapp” and then listing the other files/folders within it.

A screenshot of a terminal window titled 'myapp - bash - 102x31'. The terminal shows the following commands and output:

```
Last login: Thu Oct 9 17:44:05 on ttys001
Laurences-MacBook-Pro:~ laurencebradford$ cd Desktop
Laurences-MacBook-Pro:Desktop laurencebradford$ cd myapp
Laurences-MacBook-Pro:myapp laurencebradford$ ls
Gemfile      Rakefile      config         lib            test
Gemfile.lock app            config.ru      log            tmp
README.rdoc  bin           db             public         vendor
Laurences-MacBook-Pro:myapp laurencebradford$
```

The background of the terminal window shows a dark, mountainous landscape.

(<https://learntocodewith.me/getting-started/topics/command-line/attachment/cli/>)

The shell is essentially where you are writing the commands in the command line. [Bash](http://www.wikiwand.com/en/Bash_(Unix_shell)) ([http://www.wikiwand.com/en/Bash_\(Unix_shell\)](http://www.wikiwand.com/en/Bash_(Unix_shell))) is a widely used Unix shell (as seen above.) For Windows machines, you typically would use [Command Prompt](http://www.wikiwand.com/en/Cmd.exe) (<http://www.wikiwand.com/en/Cmd.exe>).

According to Google, a Kernel is:

“... the central component of most computer operating systems; it is a bridge between applications and the actual data processing done at the hardware level.”

The point is, both the GUI and CLI are ways you can interact with your computer. But when it comes to the command line, once you learn the commands, it is faster and more powerful to navigate around your computer. That’s why many programmers prefer it.

In fact, **there are lots of directions your computer can only execute if you tell it to do so via the command line.** That's yet another reason why programmers must learn how to use it.

👉 [Back to the table of contents](#)

Start coding now

Stop waiting and start learning! Get my 10 tips on teaching yourself how to code.

ENTER YOUR EMAIL ADDRESS

GET THE TIPS!

Don't worry. I'll never, ever spam you!

Why Should I Learn the Command Line?

Why is learning command line skills important?

According to Michael,

"A lot of very useful programs work best on the CLI and it's easier to set up and run them on there. For instance git ([http://www.wikiwand.com/en/Git_\(software\)](http://www.wikiwand.com/en/Git_(software))), an often-used software versioning system, was designed exclusively for the command line (though someone has probably made a GUI). It's easier and faster to use and work on the code base with it."

Despite whether you have used it or not, I am sure most of you reading this have heard of [Github](https://learntocodewith.me/learn/github/) or [Bitbucket](https://bitbucket.org/) before. This is what Michael is talking about above. Lots of programmers working individually or in collaborative

(<https://learntocodewith.me>)

environments “push” their repository to [Github](https://learntocodewith.me/tutorials/github-pages/) (<https://learntocodewith.me/tutorials/github-pages/>) or Bitbucket. There are also many other features and benefits to using git.

Moreover, Michael explains:

“The CLI allows you to do manipulations with your system’s internals and with code in a much more fine-tuned way.. [It] offers greater flexibility and control than a GUI regardless of what OS is used. Many programs that you might want to use in your software that are hosted on say Github also require running some commands on the CLI in order to get them running.”

Basically, it’s really, really important that you at least have a basic understanding of the command line. But to be a full-fledged developer, you have to be comfortable working in the command line interface.

👉 [Back to the table of contents](#)

What Does the Command Line Look Like?

Anyone with a computer can easily start using the command line, as it comes preinstalled with Windows, Linux, and Mac computers. You don’t need any special command line tools, just the skills!

To open the command prompt in Windows, open the Start menu, click the Windows icon, scroll down to find the Windows System folder, and click Command Prompt.

To open the terminal on Mac, open spotlight search (command + the space bar), type in “terminal,” and select the terminal application.

How to use command line interfaces begins with understanding the components of a command. CLI coding is made up of a few elements: a prompt, a command to run, an option/flag, and an argument/parameter.

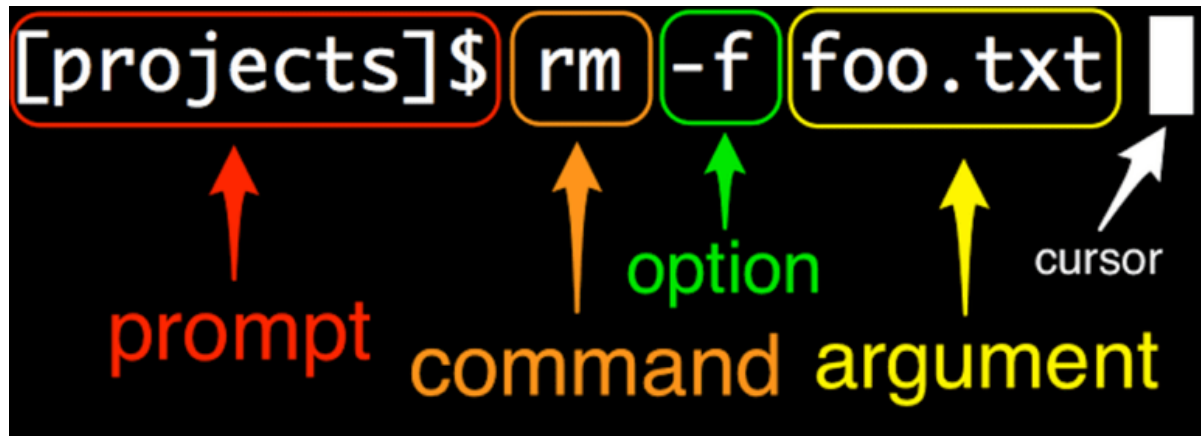


Image source: [Learn Enough Command Line Tutorial](https://www.learnenough.com/command-line-tutorial) (<https://www.learnenough.com/command-line-tutorial>)

Prompts: Every command line starts with a prompt, which is provided automatically by the terminal and usually ends with a \$ or %. The prompt depends on the details of your system and can be customized (which is a more advanced topic).

Commands: This is where you tell the computer what to do. Some examples of commands include:

- `cd`: change directory
- `open`: open files
- `mv`: move a file
- `mkdir`: create a directory
- `top`: list actively running computer processes
- `ditto`: copy contents of folder to a new folder

Options: Bits of info that customize how a command works. For example, for the Linux ls (ls) command that allows you to view a list of files and folders in a given directory, options include:

- -l: shows permissions, the user who made it, and other details
- -a: shows hidden files
- -S: sort by file size, largest first
- -C: list entries by columns

Arguments: An argument could be the string of characters you want to print or paths to files or folders. Not all programs need arguments or parameters.

 [Back to the table of contents](#)

How to Learn Command Line Skills

So, why learn command line skills for your coding journey? And if you want to learn the command line, what's the best way to go about it? In this section, we'll quickly cover some of the important Q&As about how to learn the command line.

Who Should Learn How to Use Command Line Interfaces?

Learning how to operate within the command line has practical applications for any sort of programmer, especially those working on the backend. As Michael says:

“If you’re a backend developer [it] matters more since you’ll be dealing with the server aspect of it, such as what OS is it running, configuration files, and other things needed to make sure it is running ... [also] making sure the code for your site is working.”

Nonetheless, even if you’re a [front end developer](https://learntocodewith.me/web-dev/front-end-developer-skills/) (<https://learntocodewith.me/web-dev/front-end-developer-skills/>), knowing how to work with the CLI is beneficial:

“Front end developers don’t need to know it as much as you are really just designing the pages for the website, though it certainly wouldn’t hurt to know it if you’re trying to beef up your resume ... And you’ll definitely need to know some of it if your team is using git.”

Front end devs don’t need to know the CLI as well as backend devs. But it can boost your resume.
([https://twitter.com/share?](https://twitter.com/share?text=Front+end+devs+don%27t+need+to+know+the+CLI%26nbsp%3Bas+well%26nbsp%3Bas+backend+devs.+But+it%26nbsp%3Bcan+boost%26nbsp%3Byour+resume.&via=learncodewithme&related=learncodewithme&url=https://learntocodewith.me/learn/command-line/)
[text=Front+end+devs+don%27t+need+to+know+the+CLI%26nbsp%3Bas+well%26nbsp%3Bas+backend+devs.+But+it%26nbsp%3Bcan+boost%26nbsp%3Byour+resume.&via=learncodewithme&related=learncodewithme&url=https://learntocodewith.me/learn/command-line/](https://twitter.com/share?text=Front+end+devs+don%27t+need+to+know+the+CLI%26nbsp%3Bas+well%26nbsp%3Bas+backend+devs.+But+it%26nbsp%3Bcan+boost%26nbsp%3Byour+resume.&via=learncodewithme&related=learncodewithme&url=https://learntocodewith.me/learn/command-line/))

3BYOUR+RESUME.&VIA=LEARNCODEWITHME&RELATED=LEARNCODEWITHME&URL=HTTPS://LEARNTOCODEWITH.ME/LEARN/COMMAND-

Proficiency with the command line isn’t only beneficial for developers. It can also be helpful for anyone who wants to *work* closely with developers, such as [UX designers](https://learntocodewith.me/posts/web-developers-) (<https://learntocodewith.me/posts/web-developers->

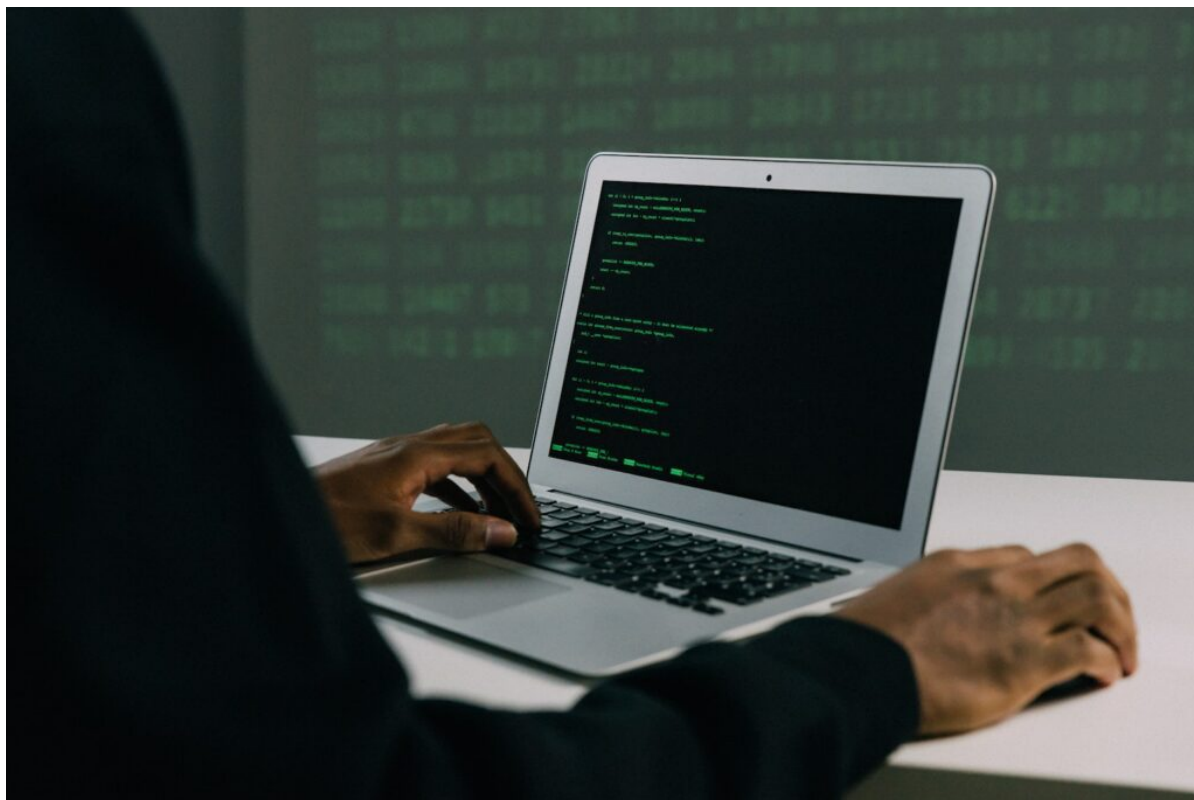
[ux-designers/](https://learntocodewith.me/work-in-tech/product-management/)) and [product managers \(https://learntocodewith.me/work-in-tech/product-management/\)](https://learntocodewith.me/work-in-tech/product-management/). Some people, like Michael, use the CLI almost exclusively for they find it preferable to the GUI.

Ultimately, learning command line basics (at least) is crucial if you ever want to be taken seriously as a developer at the professional level.

How long does it take to learn the command line?

The command line is relatively easy to learn, so it should take only a few weeks to get familiar with it. Of course, there are more advanced topics that will take longer to master, but understanding the syntax and how it works should only take a few weeks.

For example, [Codecademy's Learn the Command Line \(https://learntocodewith.me/go/codecademy-learn-command-line/\)](https://learntocodewith.me/go/codecademy-learn-command-line/) takes eight hours to complete. You can learn the basics of the command line in [Coursera's Practical Introduction to the Command Line \(https://learntocodewith.me/go/coursera-practical-introduction-to-the-command-line/\)](https://learntocodewith.me/go/coursera-practical-introduction-to-the-command-line/) in just two hours.



? When to learn the command line

If you're new to coding, you might be wondering *when* you should learn the command line. I recommend learning [HTML](https://learntocodewith.me/learn/html/) (<https://learntocodewith.me/learn/html/>), [CSS](https://learntocodewith.me/learn/css/) (<https://learntocodewith.me/learn/css/>), and [JavaScript](https://learntocodewith.me/learn/javascript/) (<https://learntocodewith.me/learn/javascript/>) first, then tackling the command line.

However, you can choose to start learning the command line while learning these other languages. It just might be easier to understand if you have at least a basic knowledge of HTML/CSS and JavaScript.

Want to get better with HTML?

Download my free HTML5 cheat sheet below.

GET IT NOW

Don't worry. I'll never, ever spam you!

Where to learn command line programming

WHAT MICHAEL RECOMMENDS

As far as recommended resources go, Michael recommends:

- **[LinuxCommand.org](http://linuxcommand.org)**
(http://linuxcommand.org/lc3_learning_the_shell.php): An excellent resource to begin learning the CLI, it builds from the basics up to shell scripting
- **[Arch Wiki's Command Line Shell](https://wiki.archlinux.org/index.php/Command-line_shell)**
(https://wiki.archlinux.org/index.php/Command-line_shell): While this information may be from the Arch Wiki it is a resource that documents many things and usually has links to other resources.
- **[Zed Shaw's Command Line Crash Course](http://cglab.ca/~morin/teaching/1405/clcc/book/cli-crash-course.html)**
(<http://cglab.ca/~morin/teaching/1405/clcc/book/cli-crash-course.html>): Covers Linux, Mac and Windows and is well worth the read if you just want to learn the basics.

WHAT I RECOMMEND

- (<https://www.learnenough.com/command-line>) **Unix Command Cheat Sheet**: (<https://learntocodewith.me/command-line/unix-command-cheat-sheet/>) First things first, check out my handy Unix command line cheat sheet.
- **[Learn Enough Command Line to Be Dangerous](https://learntocodewith.me/go/learn-enough-developer-fundamentals-01-learn-enough-command/)**
(<https://learntocodewith.me/go/learn-enough-developer-fundamentals-01-learn-enough-command/>) by Michael Hartl is an introduction to the Unix command line for complete beginners.
- **[Pluralsight's Meet the Command Line](#)**

(<https://learntocodewith.me/go/pluralsight-command-line/>): This Unix command line course starts with the basics and ends with advanced tips and tricks. *(Read our full Pluralsight review [here](#) (<https://learntocodewith.me/reviews/pluralsight/>)).*

- **[Team Treehouse's Introduction to the Terminal](#)**

(<https://learntocodewith.me/console-foundations-treehouse/>): I love Treehouse – their courses are taught by professionals and you can take them at your own pace.

- **[Codecademy's Learn the Command Line](#)**

(<https://learntocodewith.me/go/codecademy-learn-command-line/>): This course will teach you to navigate, access, and modify files and folders on your computer—all without a mouse! *(Read our full Codecademy review [here](#) (<https://learntocodewith.me/reviews/codecademy/>)).*

- **[Udacity's Linux Command Line Basics](#)**

(<https://learntocodewith.me/go/udacity-linux-command-line-basics/>): Learn the basics of the command line interface of a Linux server: the terminal and shell (GNU Bash). This course includes an introduction to files and directories in the Linux filesystem.

- **[Launch School's A Guide to Git and Github](#)**

(<https://learntocodewith.me/go/launch-school-git-github-book/>): This free beginner-friendly ebook offers a gentle intro to Git and GitHub.

- **[Skillshare's Command Line Interface / Linux Terminal for Absolute Beginners](#)** (<https://learntocodewith.me/go/skillshare-command-line-interface/>): Learn how to use the terminal on any Linux or Unix based machines.

Now it's up to you to start learning command line programming! The command line for beginners is a fantastic foundational skill to help set yourself up for future success as a developer.

About the Author

Michael Gattozzi is a student at Umass Boston and a member of the MA Army National Guard. In his free time he likes to write scripts, read technical and non technical things, and tinker with his laptop for the perfect setup. He's also a follower of the Google Dark Arts and an Arch Linux die hard fan.

You can follow him on Github on github.com/mgattozzi (<https://github.com/mgattozzi>) and on Twitter [@mgattozzi](https://twitter.com/mgattozzi) (<https://twitter.com/mgattozzi>).

Tweet



About Laurence Bradford

Hey! I'm the creator of Learn to Code With Me, a blog for beginners teaching themselves how to code. Once upon a time, I was digitally unsavvy. *But not anymore.* Find out how I changed my life by [teaching myself digital skills here](https://learntocodewith.me/about/) (<https://learntocodewith.me/about/>).

Read Next...

Popular Posts

1. [The 22 Best Coding Interview Courses and Books \(2021 List\)](https://learntocodewith.me/posts/coding-interview-resources/) (<https://learntocodewith.me/posts/coding-interview-resources/>)
2. [70 Best Programming Books for Every Coder in 2022](https://learntocodewith.me/posts/programming-books/) (<https://learntocodewith.me/posts/programming-books/>)
3. [113 of the Best Places to Learn to](#)

What Is Low Code / No Code Development?

(<https://learntocodewith.me/learn/low-code-no-code/>)

READ NEXT POST » (<https://learntocodewith.me/learn/low-code-no-code/>)

[Code For Free](https://learntocodewith.me/posts/code-for-free/)

(<https://learntocodewith.me/posts/code-for-free/>)

4. [17 Steps to Becoming a Software Engineer \(Without a CS Degree\)](https://learntocodewith.me/posts/become-a-software-engineer/)
(<https://learntocodewith.me/posts/become-a-software-engineer/>)
5. [43 Best Web Development Courses Online in 2021 \(Free & Paid\)](https://learntocodewith.me/reviews/best-web-development-courses/)
(<https://learntocodewith.me/reviews/best-web-development-courses/>)

Coding platform reviews

Coursera

(<https://learntocodewith.me/reviews/coursera/>)

Udemy

(<https://learntocodewith.me/reviews/udemy/>)


Codecademy

(<https://learntocodewith.me/reviews/codecademy/>)

Udacity

(<https://learntocodewith.me/reviews/udacity/>)

Recommendations

 Best Tech & Coding Gifts

(<https://learntocodewith.me/posts/best-tech-coding-gifts/>)

101+ Free Places to Learn to Code

(<https://learntocodewith.me/posts/101-free-places-to-learn-to-code/>)

Code for free

(<https://learntocodewith.me/posts/code-for-free/>)

Best Programming Books

(<https://learntocodewith.me/posts/best-programming-books/>)

Top 2021 Tech Skills

(<https://learntocodewith.me/posts/top-2021-tech-skills-in-demand/>)

Working in tech

Make Money While Learning

to Code
(<https://learntocodewith.me/posts/make-money-coding/>)

Building Your Online Portfolio

(<https://learntocodewith.me/posts/building-your-online-portfolio-tips/>)

Acing Your Tech Interview

(<https://learntocodewith.me/posts/acing-your-tech-interview/>)

Freelance Starter Guide

(<https://learntocodewith.me/freelance-starter-guide/>)

Special resources

Tech Resources for POC

(<https://learntocodewith.me/posts/tech-resources/>)

Coding Resources for Women

(<https://learntocodewith.me/posts/coding-resources-for-women-learn-code/>)

Learning to Code After Age 40

(<https://learntocodewith.me/posts/learning-to-code-after-40/>)

Teach Your Child to Code

(<https://learntocodewith.me/resources/teach-your-kids/>)

© 2014 - 2021

learntocodewith.me

(<https://learntocodewith.me/>)

Get digital skills, be happy.

Follow us



(<https://www.facebook.com/learntocodewithme>)



(<https://twitter.com/learncodewithme>)



(<https://www.instagram.com/laurencebradford/>)



(<https://www.youtube.com/learntocodewithme>)

Blog

(<https://learntocodewith.me/blog/>)

Podcast

(<https://learntocodewith.me/podcast/>)

About

(<https://learntocodewith.me/about/>)

FAQ

(<https://learntocodewith.me/faq/>)

Contact

(<https://learntocodewith.me/contact/>)

Permissions Policy

(<https://learntocodewith.me/permissions-policy/>)

Earnings Disclaimer

(<https://learntocodewith.me/affiliate-disclaimer/>)

Privacy Policy

(<https://learntocodewith.me/privacy-policy/>)

Learning tech skills has changed my life for the better. **And it can do the same for you.**

LEARN TO CODE, TOO ([HTTPS://LEARNTOCODEWITH.ME/](https://learntocodewith.me/))

Built by Laurence Bradford with ♥ and ☕. WordPress hosting by Kinsta (<https://learntocodewith.me/go/kinsta/>). CDN by StackPath (<https://learntocodewith.me/go/stackpath/>).