Touch Typing

Adrian Kosmaczewski

2021-01-08

Probably the single most important skill that has always helped me when dealing with computers is the ability to type without looking at the keyboard.

I say it has been handy because, well, simply we spend a lot of time using keyboards, and they become the primary means of interaction with computers. Of course, these days we have lots of different input types; trackpads, mice, Wacom tablets, touchscreens… but the keyboard has kept a big deal of charm and usefulness, and touch typing has helped me master more about computers than I would have ever thought possible.

- Advantages
  - Note Taking
  - Bragging
  - Coding
  - Command Line
  - Vim
- Keyboard Fetish
- Ergonomic Keyboards
- How To Learn
- Keyboard Layouts
- Keyboard Suggestions

Advantages

I do not claim I am the best or faster typist around, mind you; I have seen much better and faster ones, and even for my own taste, I keep pushing the "backspace" key much too often.

But nevertheless, being proficient with a keyboard has provided several advantages.

Note Taking

The first useful thing about learning to touch type is simply being able to jolt down notes while listening to someone; and by listening I mean actually looking

at the person in the eyes. Yes, just like a stenographer would do, listen and type.

Bragging

At some point, somebody will see you take notes without peeking at the keyboard and you are going to get some compliments for that. Be proud about it.

Coding

When you are coding, your brain must wire itself in order to solve the problem at hand. Most software writing is relatively simple; few of us are actually sending probes to Mars or solving cancer or managing nuclear plants, so that our requirements for algorithms are quite simplistic.

But, nevertheless, coding requires attention, and if your brain spends time figuring out where the different letters are, you are going to pay less attention to the structure of your code, to your algorithms and other details. It is a simple logic; if the main thread in your brain has to pay attention to your hands, instead of just using a background thread to them, you are not going to do a great job. Keep your main thread free for analysis and coding.

Command Line

After 30 years of graphical user interfaces, I find fascinating that the most advanced visual technologies are still lagging behind the power of a good command line prompt. Call me old fashioned, but even if I enjoy the commodity of a good IDE from time to time, there is nothing like a good `Makefile` or a shell script to get things done.

In particular, I am fond of using tmux for my command line needs. Productivity is key when the time is short, so I tend to choose technologies "command line friendly" technologies whenever possible.

Vim

Related to the previous point, there is my beloved text editor, Vim. I wrote quite a few chapters about it in my book, so I do not need to say more right now; but suffice to say that touch typing has opened me the door of an absolutely delightful and incredibly powerful piece of software, probably the most stable and useful editor I have ever used in my life.

Suffice to say that Vim works best when your finger are glued to the keyboard. No mouse, nothing but keystrokes and power.

Keyboard Fetish

Oh, speaking about keyboards and power, why deny it; you will develop a keyboard fetish. It is part of the game.

You are going to realize that some keyboards are better than other, in subtle ways; some of them just require too much effort to write even the simplest of sentence. Some others are simply too soft, and will make your writing even worse, with repeated letters all over the place. Then there is the height, the width, the color, maybe, and finally, even the smell. You are going to buy them, collect them, talk about them, discuss their relative merits, people are going to hate or admire you for them, and this is just part of the game.

My personal favorites at the time of this writing are: the Das Keyboard Ultimate 4, the Apple Wireless keyboard, and in general, most Microsoft and Logitech keyboards. Historically speaking, I used to love the classic IBM PC keyboards, and I would love to own a Model M one day. I can say the same about the Apple Extended Keyboard II. I would love to own those.

Ergonomic Keyboards

I used to own a Microsoft Ergonomic keyboard back in the nineties; I loved writing with my wrists in a natural position. I find it sad that there are no good ergonomic keyboards around these days anymore. I think that the idea of an ergonomic keyboard was absolutely fantastic, and having used them, I could really see myself writing this book in one of them.

How To Learn

I learnt touch typing at school, in Argentina, in 1989. At home we had a beautiful Remington Typewriter, one that belonged to my grandmother, and I loved to write in it. However, my mother, who was a secretary, would type in her IBM Selectronic typewriters at work at incredible speed, and I wanted to know how to do that.

In 1989, when I was in 3rd year of high school, we had the option to choose between a course in drawing and another about touch typing. I chose the latter. In our school there were no computers (remember, this is the Argentina of the '80s I mentioned so many times in this blog) so we had a large room stocked with old mechanical typewriters from various brands and origins, in the last floor of the school building.

We would go there every Friday morning, during one hour, for a whole year, and the machines had metal plates on top of the keyboards, so that we would not see our hands while writing. The exercises consisted of filling sheets of paper with sequences like `abababababab cdcdcdcdcd efefefefef` and more, so that our fingers would remember the position of the different keys once and for all.

Keyboard Layouts

My personal experience shows that the American keyboard layout (commonly known as "QWERTY") is the one and only one that helps me write good code. The best programmers I know use it, the best software I wrote was with it. Thankfully, the typewriters we used at school had either the American layout, or the Latin American one (which includes the quintessential "ñ" letter.)

The reason for this choice of the American layout is that many of the most popular programming languages (among which one can easily spot Java, C, C++, Objective-C, JavaScript and others) all use some combination of symbols (such as curly and square brackets, semicolons and apostrophes) which are immediately available in the American layout.

In stark contrast, I argue that the French ("AZERTY") and the Swiss keyboard layout ("QWERTZ") are the least appropriate for writing code. I have seen students in my training classes struggle with laptops with those keyboards. Hitting "ALT GR+6" and "ALT GR+7" to get a set of curly brackets seems to me a useless waste of energy, and if you have to repeat this every 10 seconds, you are going to get very angry, very soon.

And you do not want to code angry.

Keyboard Suggestions

Here goes a list of interesting keyboards out of the ordinary; find your own favorite here:

- Ergodox EZ
- Moonlander Mark I
- Keyboardio Atreus
- Razer BlackWidow V3 Pro
- Das Keyboard
- Ultimate Hacking Keyboard
- MAX Keyboards
- Pimoroni Keybow
- Keychron
- WASD Keyboards
- Happy Hacking Keyboard
- Unicomp PC "Model M"-like Keyboard
- GH60 Programmable Keyboard
- KBDfans
- CODE Mechanical Keyboards
- Matias Ergo Pro and Tactile Pro (the latter for Mac users, particularly those feeling nostalgia about the Apple Extended keyboard)

For the record, I used my Das Keyboard 4 Ultimate to write these lines. Highly recommended.

PS: If you liked this article, you will enjoy my book "Tales of Editors & Keyboards", freely available in this website.