Timothy Bruggeman

Experience is Better than Knowledge

Books are like stories, ready to be enjoyed and learned from, but books can never compare with the adventures you have on your own. Of course, just mindlessly doing a task is not always enough. On some level it will prepare you to do the task on the most mundane baseline. However, if the job at some point requires thought you can’t expect to magically know how to think through a problem if you have not had any practice. Humans work off of their practical experiences. No experiences means an inability to perform. Therefore when practicing we need to be deliberate in thinking about why we do something or not. Every time we code or look at some other code we need to think about why functions are the way they are and what they do. This is necessary to see how other people solve a problem and what the trade-offs are doing things certain ways. All of this gives experience.

The number ten thousand has been tossed around as an example to illustrate how much work needs to be put into something before anything of any worth can be created. It is just an arbitrary number, but it is large. Practice makes permanent and if you can practice you can improve and make those improvements stick. But practice, practice, practice. You need more practice to improve. Every time you do something you will get better. This is why when you look back at the early things you did you will realize how bad they were and how much you improved.

The two steps that stuck out to me were “work on projects after other people” and “learn at least a half-dozen languages”. Both of these steps are to help you see different perspectives and ways of programming. But they are very different. Working on projects after other people requires you to understand how a person did a thing even if it is not exactly how you would do it. Maybe there are only subtle differences, maybe it is fundamentally different. Either way gives a new vision. Reading code can be a bit complicated, so that is good practice as well to just find information through the logic and looking up, down, and sideways at every point on the page. Different languages are good to understand because they make you think of logic problems in entirely new ways, even more than other people’s codes. The examples given in the prompt were excellent. Every coding language picks up entirely new and different ways to do things, which is fantastic and gives me new colors for my pallet.