

What is it like getting a job as a software developer? Well, it's hard to compare it to anything.

Employees at companies will care very strongly about specific things, and if you're not a mind reader, or somehow magically on the same page as them then you're not a good match. What's worse is that every company cares strongly about different things. After interviewing at multiple companies inconsistency is the consistency.

Some companies care a lot about in-code documentation, every method must be documented. Some companies think in-code documentation is a waste of valuable time.

Some companies think documentation is valuable. Some companies think no documentation is job security.

Some companies care a lot about certain design patterns. Some companies do not.

Some companies think everything should be done in JavaScript. Some companies do not.

Some companies think you should make use of new language features. Some companies think you should stick with traditional features that will never change.

Some companies think unit testing is a valuable practice. Some companies think unit testing is a complete waste of time.

Some companies think it's important to have cameras on during meetings. Some companies do not care. Some companies are somehow in the middle too, in a sort of follow the crowd kind of way.

Some companies think every database needs to be MongoDB. Other companies think MongoDB is an OmgnoDB. Some companies think MongoDB has a place in certain circumstances.

Some companies think Kubernetes is the future and needs to be used to orchestrate every project. Some companies, much like my spell-checker, have no idea what Kubernetes even is.

Some companies think development should be done within Linux. Other companies think development should be done within Mac-OS. Other companies think Windows OS is the best operating system for developers.

Some companies think JavaScript should always be used with TypeScript. Well, most companies seem to think this.

Some companies think everything on Raspberry Pis should be written using Python. Others think it should all be in C. Some others think C++ is the correct choice.

Some companies think leetcode is the best measure of a developer. Some companies think arbitrary questions are. Yet others think building a small project upon request is.

Some companies program only using Microsoft products and they only deploy to servers controlled by Microsoft. Seriously some companies really like Microsoft. Other companies think Microsoft produces malware and should be avoid.

Some companies think mobile development should only be done with Kotlin and Swift. Others use Java. Some even use React Native and probably NodeJS on their backend too.

Some companies think Java is great. Others think Java is horrible.

Some companies think Java is a cutting edge technology. Some companies think Golang is.

Some companies think coding begins with abstraction. Other companies think abstraction should be implemented when it is necessary.

Some companies care a lot about implementing SOLID principles. Some companies do not care about SOLID principles.

Some companies practice thorough code reviews. Some companies don't even have code reviews.

Most companies think a software developer should be more social than they need to be.

What is the process for developing the best software? I used to think I knew. I used to think it involved robust testing, documentation, refinement, refactoring, design patterns, optimizations, time-space complexity, and many other things.

This industry represents itself, in such a way, where there are no real correct or standard answers here.

What is correct in one office at one company is incorrect at another office in another company.

Going through school, I did develop a lack of respect for the soft sciences. How their theory was founded on whimsical evidentiary support. Much of it appeared closer to opinion than it did science. Additionally, psychology seemed to have two camps for any given question, the 'yes' camp, and the 'no' camp. Perhaps there was a third camp, the 'I don't know' camp. Shoot, maybe everyone is actually in the 'I don't know' camp and they are just in denial about it.

I have felt that it was incomprehensible that some people chose to dedicate their lives to a field as flimsy, as, for example, psychology. It isn't so much about the necessity for psychologists in this world, rather it is about the depth of the field itself. I viewed it as a rather shallow field that lacks robustness.

Today, thinking about that makes me feel like a hypocrite.

Today, I look at myself. I look at this software industry.

Did we trick the world?

Did I trick myself?