

Programmer commentary

Tom Rochette <tom.rochette@coreteks.org>

January 30, 2020 — [e9ce34f1](#)

1 Question

Is a programmer commentary something that can provide value?

2 Answer

A few years ago I was learning how to drive. One of the techniques they teach in order to learn the basics of driving and help you reduce the amount of stress related to driving is to comment on what you are doing: I'm looking at this panel that says the maximum speed is 50 km/h, I am watching in my mirror, I am signaling that I want to move to the right lane, I'm checking my right dead angle, etc. The use of this technique is that it makes what you are doing in your head explicit by saying it out loud. There's a similar technique called [pointing and calling](#) that is used by Japanese and Chinese railways drivers in order to keep focus and attention.

When we're doing pair programming with someone else, we will often communicate with the other person what our current intent is and what we're currently trying to write down. It may help us clarify our objectives or express clearly the various steps we'll have to go through. In the case of pair programming, one of the benefits is that someone else is there to double-check what you are doing and what you're about to do.

In the context where you are programming by yourself, you can still benefit from talking out loud about what you are doing. If you are willing, you can record yourself and even possibly use speech to text technologies to create a log of what you were doing at the time. This can be useful to recall what you were working on if you are interrupted at some point. It can also serve as a good way to get back context from a piece of code if you need to work on it again.

Writing down comments can also be useful, however there's a lot of thinking that goes into programming that is only transient and if it were to be written as comments, it would make the code more difficult to consume.

3 References

- [Pointing and calling](#)