It's probably harder to read code than to write

12/24/2011

Deepak Surti, deepaksurti.com
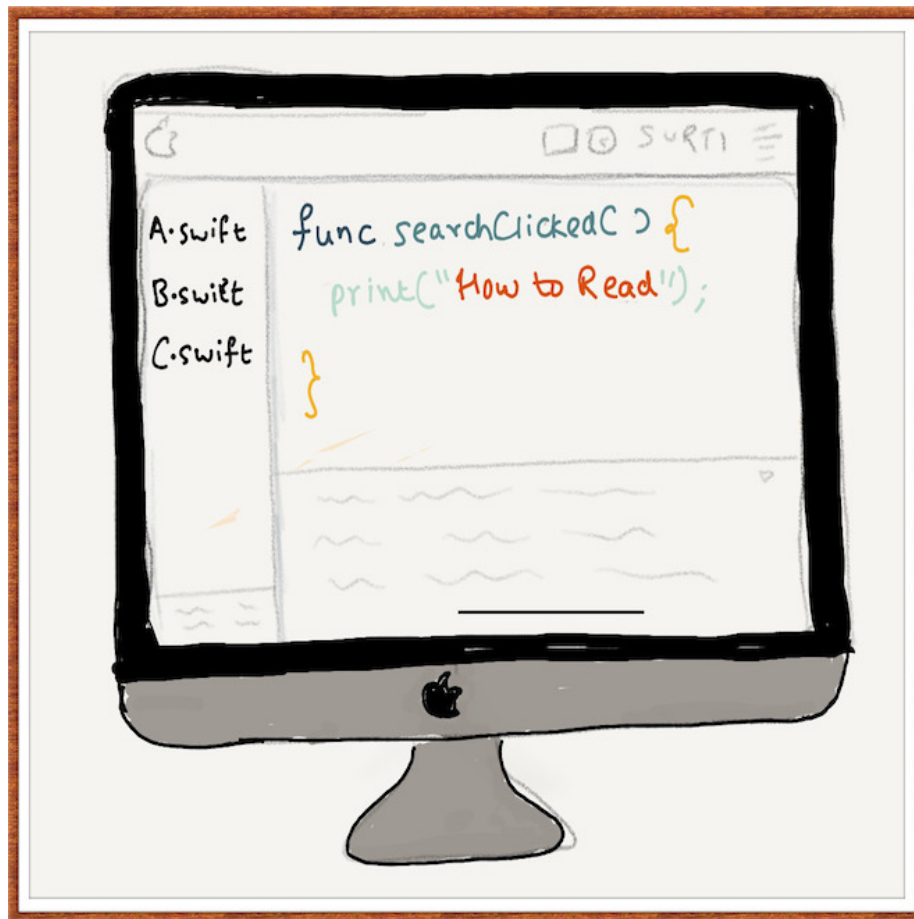
Categories: books, learning

Figure 1:

I came across this piece by Joel, where he says "It's harder to read code than to write it".

PERFECT. Atleast my programming experience matches with what he says. On the other hand, I do not interpret it as 'so just stop reading code and do

something else'. If you have to do real programming, you have to spend time 'understanding' already written code.

Now, the deeper reasons for why it could be hard to read code could be: - To read code, you need to have a purpose. That purpose is defined by any feature, typically, a bug fix or an enhancement task. But any feature does not use all code. So you will not read all code.

- Next, it is humanly impossible to read all code of any application (Have you ever)? In fact I have not even completely read all the code of any application that I wrote completely! Let alone all the code, you will not read all code related just to the feature you are working on.

- Reading code without input/output information is a mindless task. It is not productive and does not make sense. This is because input/output information gives meaning to code without having to infer it, which otherwise is hard, error prone and cumbersome.

Here I have found my detective utilities (here , here) very handy. It eliminates the futility of reading code as it helps me 'understand' code relevant to the feature I am working upon as it provides: Code trace relevant to a feature (I start, stop detective) Code information such as input, output, argument names and time spent

With the information provided by detective, I can explain the code in business terms to someone else while pointing to function names. Thats it. I have learnt the lesson the hard way.

So if Joel says "Its harder to read code than to write it.", we now have a corallary:

"Never read code without contextual information."