

Coding a Better World with Uncle Bob

Uncle Bob approved tricks to keep your code clean:

Single responsibility: a function should only do one thing, and it should do it well!

One level of abstraction per function: each line of a function should be at the same of level of abstraction. When you're writing code, your brain does this isolation from the highest to the lowest level. Your job is to clean or fix the code once it works.

Meaningful names matter: construct well-written statements from the names of variables and functions. Functions should be named after a verb or a verb phrase. Classes and objects should have nouns or noun phrases as names.

Functions should be as small as possible: extract another function from the function until you cannot extract any more.

Flag arguments: do not pass booleans as arguments to functions. By doing that, you'll lose the 'Single Responsibility Principle' because of if-else statements.

Avoid side effects: side effects equal to lies in your contracts because they are not apparent to the reader at first sight. Make sure functions don't cause side effects in your code.

Less arguments are better: functions shouldn't have more than 3 arguments. Keep the number as low as possible. When a function seems like it needs more arguments than that, there's a good chance some of these ought to be wrapped into a class of their own.

Comments are not purely good: don't use comments unless they are absolutely necessary. Instead, make sure that your classes, methods, and variables describe their purpose. Remember: the best documentation is the code itself.

DRY Principle: Don't Repeat Yourself; don't duplicate code.

Quotes

"The only way to go fast, is to go well."—Uncle Bob

"You are not done when it works, you are done when it's right."—Uncle Bob

Software rules the world!