Clean code

Ali Farnudi, Nov 2022

Code is read
much often than it is written

Code writing starts fast... then it gets slow

Code writing starts fast...
then it gets slow

In the beginning:

A: Hey, how long does it take to add a new feature?
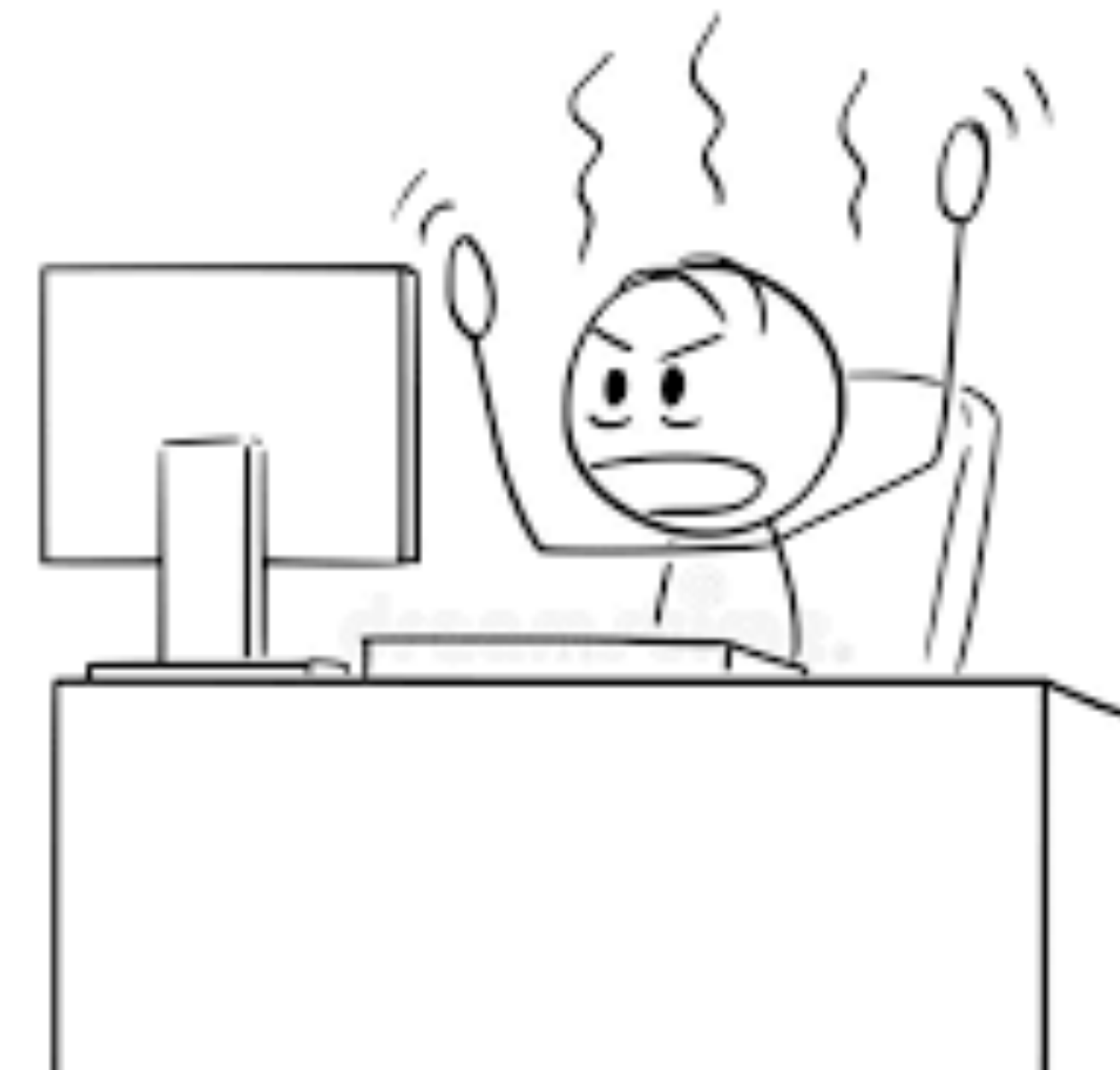
Programmer: Maybe a couple of days

After a year:

A: Hey, how long does it take to add a new feature?

Programmer: Maybe 6 months..

A: It used to be quicker.

Programmer: Yes, but you don't know how messy it has become! If I touch even one line, all hell could breaks loose.

What is a clean code?

Clean code always looks like it was written by someone who cares.

Michael C. Feathers

(Expert at revitalising legacy code)

You know you are working on clean code when each routine you read turns out to be pretty much what you expect...

Ward Cunningham
(Inventor of the wiki)

Basics of code writing

- Use self-explanatory variable, function, and class names

- use consistent capitalisation

- Use a consistent coding style (PEP8)

- Keep functions short

- Comment if you chose to write a complicated algorithm

- Comments are dead

```python
a = "membraneTest.txt"

def func1():
    pass
```

```python
config_file_name = "membraneTest.txt"

def get_colour_index():
    pass
```

PEP8
Python Enhancement Proposal

- Modules, function, and instance variable names:

    all_lower_case_separated_by_underscore

- Constants:

    ALL_CAPITAL

- Class names:

    CapWords

- Global Variable:

    Only used within one module

Go to [pep8.org](pep8.org)

Pip3 install pycodestyle

Use 'black'
https://pypi.org/project/black

Pip3 install black

Python code formatter

PEP 8 Song

When should I clean my code?

You clean code as soon as it works!

How long will it take?

Almost the same time as you wrote it!

This is the big problem

You get fast
by doing well

You are writing code that other people need to maintain and use

'Clean code'
by Robert C. Martin

[Link to summary](#)

"Always code as if the guy who ends up maintaining your code will be a **violent psychopath** who knows where you live"

Martin Golding