

GENERAL SOFTWARE DEVELOPMENT PRINCIPLES

DRY: Don't Repeat Yourself

Не Повторяйся

Не копируй блоки кода без каких-либо изменений.

Заключай их в функции или методы.

Конкретная проблема должна решаться в системе в единственном месте

YAGNI: You Ain't Gonna Need It

Тебе Это Не Понадобится

Решай проблемы, которые очевидны.

Выполняй текущие задачи.

Не пиши спекулятивный код:

при написании кода может захотеться

реализовать дополнительную функциональность,

которая может потребоваться в будущем. Это плохая идея.

CODE SMELLS

Запахи Кода

При чтении своего или чужого кода, который может быть правильным, может компилироваться и даже работает, иногда кажется, что что-то не так. Это и есть запах кода.

Некоторые примеры запаха кода:

- ▷ **слишком длинный метод.** Разворачиваешь метод и видишь 250-300 строк. Такой метод следует разделить на гораздо меньшие методы
- ▷ **слишком короткие или слишком длинные идентификаторы.** Кроме итератора `i` не следует применять идентификаторы `a`, `b`, `c`.
- ▷ **бесполезные комментарии.** Комментарии нужны, но только не такие, которые больше чем сам код и ничего не поясняют
- ▷ **божественный объект,** который делает почти все в программе или выполняет очень разные обязанности, почти не связанные одна с другой. Такой объект следует проанализировать и разделить на правильные объекты
- ▷ **излишняя зависимость.** Если класс только и занят тем, что использует методы другого класса это также свидетельствует о том, что необходимо пересмотреть роль одного из классов