

## Cheat Sheet (Clean Code)

### General Rules

- Follow standard conventions
- Keep it simple
- Don't repeat yourself
- Be consistent
- Do not override safeties
- Constants should be easy to change

### Code Layout

- Use vertical formatting to separate your code and different concepts
- Variables should be declared as close to their usage as possible
- Instance variables should be declared at the top of the class
- Put statics methods on top of the package
- Similar and dependent functions should be close vertically
- Use consistent indentation

### Functions

- Functions should be relatively small and should do one thing
- Functions should have meaningful names
- Functions should have as few arguments as possible.(not more than 3 arguments)
- Avoid using output arguments and flag arguments

### Design Rules

- Try to avoid negative conditions
- Use polymorphism instead of if/else or switch statements
- Use Enums over constants
- Functions should descend only one level of abstraction
- Statements in functions should be at the same level of abstraction.

### Naming Convention

- Use meaningful names for variables, functions, classes, packages
- Separate words for variables, functions, methods and constants by underscores to improve readability
- Use descriptive names instead of abbreviations
- Replace Magic Numbers with Constants
- Always try to use the most concise but descriptive names possible

### Comments

- Always try to explain yourself in code.
- Don't be redundant.
- Don't add obvious noise.
- Don't use closing brace comments.
- Don't comment out code. Just remove.
- Use as explanation of intent.
- Use as clarification of code.

### **Exceptions Handling**

Use Exceptions instead of returning error codes

Throw exceptions with meaningful context

Write the try-catch-finally statement first, it will help you structure your code

Don't return null, don't pass null either

### **Credits**

From "Clean Code" by Robert C. Martin

### **Tests**

F.I.R.S.T : Fast, Independent, Repeatable, Self-Validating, Timely

One assert per test

Test should be easily readable

Tests should be easy to run