**Clean Code Checklist**

1. **Naming Conventions**
2. Be descriptive

* Variables & Properties: Nouns or short phrases with adjectives
* Functions & Methods: Verbs or short phrases with adjectives
* Classes: Nouns
* Be Specific
* Use yes/no questions for Booleans
* Avoid misleading names
* Be Consistent

1. **Comments & Formatting**
2. Most comments are bad, avoid them
3. Some good comments are acceptable

* Legal comments
* Warnings
* Helpful explanations
* Todos

1. Use Vertical Formatting

* Related concepts close to each other
* Add vertical space between concepts not directly related
* Write code top to bottom. Avoid calling function before defining them
* Use Horizontal Formatting
* Avoid using long lines. Break it into multiple lines
* Use indentation to express scope.

1. **Functions**

* Limit the number of parameters. Use collections to stack large number of parameters into fewer parameters
* Functions should be small and do one thing
* Levels of abstraction inside the function body should be one level below the level implied by the function.
* Avoid mixing levels of abstraction in functions
* Avoid redundant splitting
* Stay DRY (Don’t Repeat Yourself)
* Avoid unexpected side effects.
* Control Structures and Errors
* Prefer positive checks
* Avoid deep nesting
* Consider using Guard statements.
* Consider using polymorphism & factory functions
* Extract control structures into separate functions
* Use error handling only for real errors instead of those which could be resolved just by using if statements

1. **Objects & Classes**

* Focus on making objects based on real life structures.
* Build small classes – focus on a single responsibility
* Build classes with high cohesion
* Follow the “Law of Demeter” for real objects.
* Especially SRP and OCP will help a lot with writing clean code.