**Brendon Samson**

**CLEAN CODE**

**SUMMARY**

# Chapter 1: Clean Code

This chapter is about code. What makes clean and bad code. What professional programmers thinks makes up a clean code. What comprises a bad code and its effects on the overall project.

Code is a language in which we ultimately express the requirements. It is our tool to create applications that we use that could communicate with our devices. Writing code can become very messy depending on the attitude of the programmer writing it.

In coding there is a thing called a “Bad Code.” It defines a type or format of code that contains undesirable qualities that could possibly break the program instead. These negative qualities include the writing style of the programmer. There should be a standard format that should be followed by every developer creating the program to make sure that communication between them is fluent and will not cause misunderstanding and disinformation.

The opposite of “Bad Code” is “Clean Code” which every developer should strive to achieve and maintain. To achieve a clean code developers should follow the standard they have planned and continue to do so to maintain the cleanliness of the program. A program with clean code is less prone to encountering errors since the developers will be able to pinpoint exactly where to find the bug if ever it does occur.

The definition of Clean Code differs with every professional developer mentioned in the book but it is only on the words that they used. Their definition though uses different wordings its thought is still the same which is readable and simple. A clean code heavily depends on the characteristic of the developer writing it because it takes time and will to really make your code clean.

Developers are also authors and their books are their code. What all authors desire the most is for their readers to understand their writings and be able to take away something from it. If your code is considered a “Bad Code” your readers will be finding it hard to read and understand and will ultimately cause disinformation and in the worst case scenario even you, yourself, would not be able to understand your own creation in the future.

# Chapter 2: Meaningful Names

We use names to identify something correctly which equally true when it comes to writing your code. In programming there will be instances where we will be using variables, classes, objects, etc. that would require for us to name them. Each variable, classes or objects we declare and name does different jobs, so we have to make sure that their name is related to what they do to make sure we don’t mix them up with other objects. The names of your objects dictate the readability of your code.

There are different guides that you should follow to maintain a readable code. *Using Intention-Revealing Names* is one. It means that you should relate the name of your object with the task it’s supposed to do. When you use intention revealing names you are increasing the simplicity of your code makes it more readable which in turn makes it easier to understand and change whatever part of your code needs changing.

*Avoiding Disinformation* is also one of the naming rules you should follow. It means that the word you are using for the name is true for that object. If you use a word that is not true for that object it would cause disinformation although it is readable but it could cause misunderstanding because the people reading it would assume that it does the thing that you named it from but it is in fact different.

*Distinguishing Names* also helps in making your code easier to understand. There could be times where you have multiple objects the does the same thing whilst also being different. Distinguishing the name for these objects could help in the readability of your code because it would make it easier to identify for the reader.

Using Pronounceable and Searchable Names. You should be able to speak the name of your object without biting your own tongue. The name should also be searchable inside your file and less duplicate result is much more preferable.

Correctly naming your objects is essential to the readability of your code. The reader should be able to guess the job or the contents of the object just by reading the its name. The most important skill that is used when naming is your descriptive skills because the purpose of naming your objects is to understand and know what they do.