**Brendon Samson**

**Title of the Book and Chapter**

Clean Code, Chapter 3: Functions

**What are the 3 things I Learned today:**

1. **Before** – Writing complex and messy functions or doing everything inside a single function.

**After** – You are not the only one who will be reading your code. You should keep functions simple. Extract any methods into their own separate function to make it understandable.

1. **Before** – Using switch every time you encounter a logic with multiple results.

**After** – Switch is a large block of code. It could also mean infinite additions to your code whenever your case has an additional value. It can only be tolerated if they appear only once, are used to create polymorphic objects, and are hidden behind an inheritance.

1. **Before** – Having multiple Arguments for a function.

**After** – Ideally, a function should only have one argument. Two and three arguments are tolerable but more than that requires justification as to why that function requires that many arguments.

**Title of the Book and Chapter**

Clean Code, Chapter 4: Comments

**What are the 3 things I Learned today:**

1. **Before** – Depending on comments to explain your code.

**After** – You should make your code explain itself, if it can, and only use comments when necessary.

1. **Before** – Using comments anywhere.

**After** – Comments could be use to translate the meaning of some obscure argument or return type value into something readable.

1. **Before** – Rewriting already written comments.

**After** – There is no need to write a comment if the meaning is already conveyed with your code as it will only be redundant.